Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: April 14, 2016

L. Zhou
N. Kong
CNNIC
C. Qi
Teleinfo
X. Lee
CNNIC
J. Gould
VeriSign, Inc.
October 12, 2015

Extensible Provisioning Protocol (EPP) Reseller Mapping
draft-zhou-eppext-reseller-mapping-02

Abstract

This document describes an Extensible Provisioning Protocol (EPP) mapping for provisioning and management of reseller object stored in a shared central repository. Specified in Extensible Markup Language (XML), this extended mapping is applied to provide additional features required for the provisioning of resellers.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 14, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction 3
2. Conventions Used in This Document 3
3. Object Attributes 3
3.1. Reseller Identifier 4
3.2. Contact and Client Identifiers 4
3.3. Reseller State 4
3.4. Parent Identifier 4
3.5. URL 4
3.6. Disclosure of Data Elements and Attributes 5
4. EPP Command Mapping 5
4.1. EPP Query Commands 5
4.1.1. EPP <check> Command 5
4.1.2. EPP <info> Command 7
4.1.3. EPP <transfer> Command 13
4.2. EPP Transform Commands 13
4.2.1. EPP <create> Command 13
4.2.2. EPP <delete> Command 16
4.2.3. EPP <renew> Command 18
4.2.4. EPP <transfer> Command 18
4.2.5. EPP <update> Command 18
5. Formal Syntax 21
6. Internationalization Considerations 27
7. IANA Considerations 28
7.1. XML Namespace 28
7.2. EPP Extension Registry 28
8. Security Considerations 29
9. Normative References 29

Appendix A. Change Log 30
Authors' Addresses 30

1. Introduction

Domain resellers are the individuals or companies that act as agents for domain name registrars. A domain name registrar is a direct customer of the domain name registry, is represented as the sponsoring client to the server in [RFC5730], and may have several resellers to help them sell domain names to end users.

This document describes an extension mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) [RFC5730]. This EPP mapping specifies the reseller object mapping.

This document is specified using the XML 1.0 as described in [W3C.REC-xml-20040204] 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", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

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 to develop a conforming implementation.

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

3. Object Attributes

An EPP reseller 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. Reseller Identifier

Reseller identifier provides the ID of the reseller of a sponsoring registrar. Its corresponding element is <reseller:id> defined in this document.

3.2. Contact and Client Identifiers

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

3.3. Reseller State

A reseller object MUST always have at least one associated state value. Valid values include "ok", "readonly" and "terminated".

State Value Descriptions:

- o ok: the normal status value for the reseller object.

- o readonly: transform commands submitted with the reseller identifier in the reseller extension would not be allowed.

- o terminated: query and transform commands submitted with the reseller identifier in the reseller extension would not be allowed.

3.4. Parent Identifier

There can be more than one layer of resellers. The parent identifier, as defined with the <reseller:parentId> element, represents the parent reseller identifier in a child reseller. 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 SHOULD be prohibited. If reseller A has B as parent identifier, reseller B must not have reseller A as parent identifier.

3.5. URL

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

3.6. Disclosure of Data Elements and Attributes

This document supports the same disclosure features described in Section 2.9 of with the use of the <reseller:disclose> element. [RFC5733].

The <reseller:disclose> element MUST contain at least one of the following child elements:

<reseller:name type="int"/>

<reseller:name type="loc"/>

<reseller:addr type="int"/>

<reseller:addr type="loc"/>

<reseller:voice/>

<reseller:fax/>

<reseller:email/>

<reseller:url/>

<reseller:contact/>

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 reseller information via EPP.

4.1. EPP Query Commands

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

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 a <reseller:check> element that identifies the reseller namespace. The <reseller:check> element contains the following child elements:

- One or more <reseller:id> elements that contain the server-unique identifier of the reseller 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:      <reseller:check
C:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
C:        <reseller:id>res1523</reseller:id>
C:        <reseller:id>re1523</reseller:id>
C:        <reseller:id>1523res</reseller:id>
C:      </reseller: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 <reseller:chkData> element that identifies the reseller namespace. The <reseller:chkData> element contains one or more <reseller:cd> elements that contain the following child elements:

- A <reseller: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.

- An OPTIONAL <reseller: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 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>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <reseller:chkData
S:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
S:        <reseller:cd>
S:          <reseller:id avail="1">res1523</reseller:id>
S:        </reseller:cd>
S:        <reseller:cd>
S:          <reseller:id avail="0">re1523</reseller:id>
S:          <reseller:reason>In use</reseller:reason>
S:        </reseller:cd>
S:        <reseller:cd>
S:          <reseller:id avail="1">1523res</reseller:id>
S:        </reseller:cd>
S:      </reseller: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 a reseller object. In addition to the standard EPP command elements, the <info> command MUST contain a <reseller:info> element that identifies the reseller namespace. The <reseller:info> element contains the following child elements:

o A <reseller:id> element that contains the server-unique identifier of the reseller 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:      <reseller:info
C:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
C:        <reseller:id>res1523</reseller:id>
C:      </reseller: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 <reseller:infData> element that identifies the reseller namespace. The <reseller:infData> element contains the following child elements:

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

- o A <reseller:roid> element that contains the Repository Object IDentifier assigned to the reseller object when the object was created.

- o A <reseller:state> element that contains the operational state of the reseller, as defined in Section 3.3.

- o An OPTIONAL <reseller:parentId> element that contains the identifier of the parent object, as defined in Section 3.4.

- o One or two <reseller: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 UTF-8 that can be represented in the 7-bit US-ASCII character set. If a localized form (type="loc") is provided, element content MAY be represented in unrestricted UTF-8. The <reseller:postalInfo> element contains the following child elements:

  - * A <reseller:name> element that contains the name of the reseller, which SHOULD be the name of the organization.

* A <reseller:addr> element that contains address information associated with the reseller. A <reseller:addr> element contains the following child elements:

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

  + A <reseller:city> element that contains the reseller's city.

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

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

  + A <reseller:cc> element that contains the reseller's country code.

- An OPTIONAL <reseller:voice> element that contains the reseller's voice telephone number.

- An OPTIONAL <reseller:fax> element that contains the reseller's facsimile telephone number.

- A <reseller:email> element that contains the reseller's email address.

- A <reseller:url> element that contains the URL to the website of the reseller.

- Zero or more OPTIONAL <reseller:contact> elements that contain identifiers for the contact objects to be associated with the reseller object. Contact object identifiers MUST be known to the server before the contact object can be associated with the reseller object. An attribute "type" associated with <reseller:contact> is used to represent contact types. The type values include admin, tech and billing.

- A <reseller:clID> element that contains the identifier of the sponsoring client, who is the domain name registrar.

- A <reseller:crID> element that contains the identifier of the client that created the reseller object.

- A <reseller:crDate> element that contains the date and time of reseller-object creation.

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

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

- o An OPTIONAL <reseller:disclose> element that identifies elements that require exceptional server-operator handling to allow or restrict disclosure to third parties. See Section 3.6 for a description of the child elements contained within the <reseller:disclose> element.

Example <info> response for the sponsoring client:

```
S:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <reseller:infData
S:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
S:        <reseller:id>res1523</reseller:id>
S:        <reseller:roid>res1523-REP</reseller:roid>
S:        <reseller:state>ok</reseller:state>
S:        <reseller:parentId>1523res</reseller:parentId>
S:        <reseller:postalInfo type="int">
S:          <reseller:name>Example Reseller Inc.</reseller:name>
S:          <reseller:addr>
S:            <reseller:street>123 Example Dr.</reseller:street>
S:            <reseller:street>Suite 100</reseller:street>
S:            <reseller:city>Dulles</reseller:city>
S:            <reseller:sp>VA</reseller:sp>
S:            <reseller:pc>20166-6503</reseller:pc>
S:            <reseller:cc>US</reseller:cc>
S:          </reseller:addr>
S:        </reseller:postalInfo>
S:        <reseller:voice x="1234">+1.7035555555</reseller:voice>
S:        <reseller:fax>+1.7035555556</reseller:fax>
S:        <reseller:email>contact@reseller.example</reseller:email>
S:        <reseller:url>http://reseller.example</reseller:url>
S:        <reseller:contact type="admin">sh8013</reseller:contact>
S:        <reseller:contact type="billing">sh8013</reseller:contact>
S:        <reseller:clID>ClientY</reseller:clID>
S:        <reseller:crID>ClientX</reseller:crID>
S:        <reseller:crDate>1999-04-03T22:00:00.0Z</reseller:crDate>
S:        <reseller:upID>ClientX</reseller:upID>
S:        <reseller:upDate>1999-12-03T09:00:00.0Z</reseller:upDate>
S:        <reseller:disclose flag="0">
S:          <reseller:voice/>
S:          <reseller:email/>
S:        </reseller:disclose>
S:      </reseller:infData>
S:    </resData>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54322-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

Example <info> for the non-sponsoring client, according to the disclosure policy:

```
S:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <reseller:infData
S:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
S:        <reseller:id>res1523</reseller:id>
S:        <reseller:roid>res1523-REP</reseller:roid>
S:        <reseller:state>ok</reseller:state>
S:        <reseller:parentId>1523res</reseller:parentId>
S:        <reseller:postalInfo type="int">
S:          <reseller:name>Example Reseller Inc.</reseller:name>
S:          <reseller:addr>
S:            <reseller:street>123 Example Dr.</reseller:street>
S:            <reseller:street>Suite 100</reseller:street>
S:            <reseller:city>Dulles</reseller:city>
S:            <reseller:sp>VA</reseller:sp>
S:            <reseller:pc>20166-6503</reseller:pc>
S:            <reseller:cc>US</reseller:cc>
S:          </reseller:addr>
S:        </reseller:postalInfo>
S:        <reseller:fax>+1.7035555556</reseller:fax>
S:        <reseller:url>http://reseller.example</reseller:url>
S:        <reseller:clID>ClientY</reseller:clID>
S:        <reseller:crID>ClientX</reseller:crID>
S:        <reseller:crDate>1999-04-03T22:00:00.0Z</reseller:crDate>
S:        <reseller:upID>ClientX</reseller:upID>
S:        <reseller:upDate>1999-12-03T09:00:00.0Z</reseller:upDate>
S:      </reseller: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> Command

The transfer semantics does not apply to reseller object. No EPP <transfer> command is defined in this document.

4.2. EPP Transform Commands

EPP provides four commands to transform reseller-object information: <create> to create an instance of a reseller object, <delete> to delete an instance of a reseller object, <transfer> to manage reseller-object sponsorship changes, and <update> to change information associated with a reseller 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, all clients involved in the transaction MUST be notified using a service message that the action has been completed and that 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 a reseller object. In addition to the standard EPP command elements, the <create> command MUST contain a <reseller:create> element that identifies the reseller namespace. The <reseller:create> element contains the following child elements:

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

- A <reseller:state> element that contains the operational status of the reseller, as defined in Section 3.3.

- An OPTIONAL <reseller:parentId> element that contains the identifier of the parent object, as defined in Section 3.4.

- o One or two <reseller: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 UTF-8 that can be represented in the 7-bit US-ASCII character set. If a localized form (type="loc") is provided, element content MAY be represented in unrestricted UTF-8. The <reseller:postalInfo> element contains the following child elements:

  - * A <reseller:name> element that contains the name of the reseller, which SHOULD be the name of the organization..

  - * A <reseller:addr> element that contains address information associated with the reseller. A <reseller:addr> element contains the following child elements:

    - + One, two, or three OPTIONAL <reseller:street> elements that contain the reseller's street address.

    - + A <reseller:city> element that contains the reseller's city.

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

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

    - + A <reseller:cc> element that contains the reseller's country code.

- o An OPTIONAL <reseller:voice> element that contains the reseller's voice telephone number.

- o An OPTIONAL <reseller:fax> element that contains the reseller's facsimile telephone number.

- o A <reseller:email> element that contains the reseller's email address.

- o A <reseller:url> element that contains the URL to the website of the reseller.

- o Zero or more OPTIONAL <reseller:contact> elements that contain identifiers for the human or organizational social information objects associated with the reseller object.

o An OPTIONAL <reseller:disclose> element that identifies elements that require exceptional server-operator handling to allow or restrict disclosure to third parties. See Section 3.6 for a description of the child elements contained within the <reseller:disclose> element.

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:      <reseller:create
C:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
C:        <reseller:id>res1523</reseller:id>
C:        <reseller:state>ok</reseller:state>
C:        <reseller:parentId>1523res</reseller:parentId>
C:        <reseller:postalInfo type="int">
C:          <reseller:name>Example Reseller Inc.</reseller:name>
C:          <reseller:addr>
C:            <reseller:street>123 Example Dr.</reseller:street>
C:            <reseller:street>Suite 100</reseller:street>
C:            <reseller:city>Dulles</reseller:city>
C:            <reseller:sp>VA</reseller:sp>
C:            <reseller:pc>20166-6503</reseller:pc>
C:            <reseller:cc>US</reseller:cc>
C:          </reseller:addr>
C:        </reseller:postalInfo>
C:        <reseller:voice x="1234">+1.7035555555</reseller:voice>
C:        <reseller:fax>+1.7035555556</reseller:fax>
C:        <reseller:email>contact@reseller.example</reseller:email>
C:        <reseller:url>http://reseller.example</reseller:url>
C:        <reseller:contact type="admin">sh8013</reseller:contact>
C:        <reseller:contact type="billing">sh8013</reseller:contact>
C:        <reseller:disclose flag="0">
C:          <reseller:voice/>
C:          <reseller:email/>
C:        </reseller:disclose>
C:      </reseller: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 <reseller:creData> element

that identifies the reseller namespace. The <reseller:creData> element contains the following child elements:

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

- o A <reseller:crDate> element that contains the date and time of reseller-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>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <reseller:creData
S:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
S:        <reseller:id>res1523</reseller:id>
S:        <reseller:crDate>1999-04-03T22:00:00.0Z</reseller:crDate>
S:      </reseller: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 a reseller object. In addition to the standard EPP command elements, the <delete> command MUST contain a <reseller:delete> element that identifies the reseller namespace. The <reseller:delete> element MUST contain the following child element:

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

A reseller object SHOULD NOT be deleted if it is associated with other known objects. An associated reseller SHOULD NOT be deleted until associations with other known objects have been broken. A server SHOULD 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:      <reseller:delete
C:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
C:        <reseller:id>res1523</reseller:id>
C:      </reseller: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>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 reseller objects, so there is no mapping defined for the EPP <renew> command.

4.2.4. EPP <transfer> Command

Transfer semantics do not apply to reseller 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 a reseller object. In addition to the standard EPP command elements, the <update> command MUST contain a <reseller:update> element that identifies the reseller namespace. The <reseller:update> element contains the following child elements:

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

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

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

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

At least one <reseller:add>, <reseller:rem> or <reseller:rem> 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 <reseller:add> and <reseller:rem> elements contain the following child element:

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

A <reseller:chg> element contains the following OPTIONAL child elements. At least one child element MUST be present:

- o A <reseller:state> element that contains the operational status of the reseller.

- o A <reseller:parentId> element that contains the identifier of the parent object.

- o One or two <reseller: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 UTF-8 that can be represented in the 7-bit US-ASCII character set. If a localized form (type="loc") is provided, element content MAY be represented in unrestricted UTF-8. The <reseller:postalInfo> element contains the following child elements:

  - * A <reseller:name> element that contains the name of the reseller, which SHOULD be the name of the organization.

  - * A <reseller:addr> element that contains address information associated with the reseller. A <reseller:addr> element contains the following child elements:

    - + One, two, or three OPTIONAL <reseller:street> elements that contain the reseller's street address.

    - + A <reseller:city> element that contains the reseller's city.

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

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

    - + A <reseller:cc> element that contains the reseller's country code.

- o An <reseller:voice> element that contains the reseller's voice telephone number.

- o An <reseller:fax> element that contains the reseller's facsimile telephone number.

- o A <reseller:email> element that contains the reseller's email address.

- o A <reseller:url> element that contains the URL to the website of the reseller.

o An <reseller:disclose> element that identifies elements that require exceptional server-operator handling to allow or restrict disclosure to third parties. See Section 2.9 in [RFC5733] for a description of the child elements contained within the <reseller:disclose> element.

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:      <reseller:update
C:       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0">
C:        <reseller:id>res1523</reseller:id>
C:        <reseller:add>
C:          <reseller:contact type="tech">sh8013</reseller:contact>
C:        </reseller:add>
C:        <reseller:chg>
C:          <reseller:state>readonly</reseller:state>
C:          <reseller:postalInfo type="int">
C:            <reseller:org/>
C:            <reseller:addr>
C:              <reseller:street>124 Example Dr.</reseller:street>
C:              <reseller:street>Suite 200</reseller:street>
C:              <reseller:city>Dulles</reseller:city>
C:              <reseller:sp>VA</reseller:sp>
C:              <reseller:pc>20166-6503</reseller:pc>
C:              <reseller:cc>US</reseller:cc>
C:            </reseller:addr>
C:          </reseller:postalInfo>
C:          <reseller:voice>+1.7034444444</reseller:voice>
C:          <reseller:fax/>
C:          <reseller:disclose flag="1">
C:            <reseller:voice/>
C:            <reseller:email/>
C:          </reseller:disclose>
C:        </reseller:chg>
C:      </reseller: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>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.

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:reseller-1.0"
       xmlns:reseller="urn:ietf:params:xml:ns:reseller-1.0"
       xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
       xmlns:eppcom="urn:ietf:params:xml:ns:eppcom-1.0"
       xmlns:contact="urn:ietf:params:xml:ns:contact-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"/>
 <import namespace="urn:ietf:params:xml:ns:contact-1.0"/>

 <annotation>
```

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

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

      <!--
      Utility types.
      -->
      <simpleType name="stateType">
        <restriction base="token">
          <enumeration value="ok"/>
          <enumeration value="readonly"/>
          <enumeration value="terminated"/>
        </restriction>
      </simpleType>

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

      <complexType name="contactType">
       <simpleContent>
         <extension base="eppcom:clIDType">
           <attribute name="type" type="reseller:contactAttrType"/>
         </extension>
       </simpleContent>
      </complexType>

      <complexType name="postalInfoType">
        <sequence>
          <element name="name" type="contact:postalLineType"/>
          <element name="addr" type="reseller:addrType"/>
        </sequence>
        <attribute name="type" type="contact:postalInfoEnumType"
```

```
         use="required"/>
      </complexType>

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

      <complexType name="discloseType">
        <sequence>
          <element name="name" type="contact:intLocType"
           minOccurs="0" maxOccurs="2"/>
          <element name="org" type="contact:intLocType"
           minOccurs="0" maxOccurs="2"/>
          <element name="addr" type="contact:intLocType"
           minOccurs="0" maxOccurs="2"/>
          <element name="voice" minOccurs="0"/>
          <element name="fax" minOccurs="0"/>
          <element name="email" minOccurs="0"/>
          <element name="url" minOccurs="0"/>
          <element name="contact" minOccurs="0"/>
        </sequence>
        <attribute name="flag" type="boolean" use="required"/>
      </complexType>

      <!--
      Child elements of the <create> command.
      -->
          <complexType name="createType">
         <sequence>
           <element name="id" type="eppcom:clIDType"/>
           <element name="state" type="reseller:stateType"
            minOccurs="0"/>
           <element name="parentId" type="eppcom:clIDType"
            minOccurs="0"/>
           <element name="postalInfo" type="reseller:postalInfoType"
            maxOccurs="2"/>
           <element name="voice" type="contact:e164Type"
            minOccurs="0"/>
           <element name="fax" type="contact:e164Type"
```

```
            minOccurs="0"/>
          <element name="email" type="eppcom:minTokenType"/>
          <element name="url" type="anyURI"
           minOccurs="0"/>
          <element name="contact" type="reseller:contactType"
           minOccurs="0" maxOccurs="3"/>
          <element name="disclose" type="reseller:discloseType"
           minOccurs="0"/>
        </sequence>
      </complexType>

      <!--
      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>

      <!--
      Child elements of the <info> commands.
      -->
      <complexType name="infoType">
        <sequence>
          <element name="id" type="eppcom:clIDType"/>
        </sequence>
      </complexType>

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

```
            minOccurs="0"/>
           <element name="chg" type="reseller:chgType"
            minOccurs="0"/>
         </sequence>
       </complexType>

      <!--
      Data elements that can be added or removed.
      -->
       <complexType name="addRemType">
         <sequence>
           <element name="contact" type="reseller:contactType"
            minOccurs="0"/>
         </sequence>
       </complexType>

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

       <complexType name="chgPostalInfoType">
         <sequence>
           <element name="name" type="contact:postalLineType"
            minOccurs="0"/>
           <element name="org" type="contact:optPostalLineType"
            minOccurs="0"/>
           <element name="addr" type="reseller:addrType"
            minOccurs="0"/>
```

```
        </sequence>
        <attribute name="type" type="contact:postalInfoEnumType"
         use="required"/>
      </complexType>

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

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

       <complexType name="checkType">
         <sequence>
           <element name="id" type="reseller: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>

      <!--
      <create> response elements.
      -->
       <complexType name="creDataType">
         <sequence>
           <element name="id" type="eppcom:clIDType"/>
           <element name="crDate" type="dateTime"/>
         </sequence>
       </complexType>
```

```
<!--
<info> response elements.
-->
 <complexType name="infDataType">
   <sequence>
     <element name="id" type="eppcom:clIDType"/>
     <element name="roid" type="eppcom:roidType"/>
               <element name="state" type="reseller:stateType"/>
     <element name="parentId" type="eppcom:clIDType"
      minOccurs="0"/>
     <element name="postalInfo" type="reseller:postalInfoType"
      maxOccurs="2"/>
     <element name="voice" type="contact:e164Type"
      minOccurs="0"/>
     <element name="fax" type="contact:e164Type"
      minOccurs="0"/>
     <element name="email" type="eppcom:minTokenType"/>
     <element name="url" type="anyURI"
      minOccurs="0"/>
     <element name="contact" type="reseller:contactType"
      minOccurs="0" maxOccurs="3"/>
     <element name="clID" type="eppcom:clIDType"/>
     <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"/>
     <element name="disclose" type="reseller:discloseType"
      minOccurs="0"/>
   </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 and its more compact representations including UTF-8. Conformant XML processors recognize both UTF-8 and UTF-16. 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 reseller 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 is requested to assignment the following URI.

Registration request for the reseller namespace:

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

- Registrant Contact: See the "Author's Address" section of this document.

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

7.2. EPP Extension Registry

The EPP extension described in this document should be registered by the IANA in the EPP Extension Registry described in [RFC7451]. The details of the registration are as follows:

Name of Extension: Domain Reseller Object Extension

Document status: Standards Track

Reference: (insert reference to RFC version of this document)

Registrant Name and Email Address: See the "Author's Address" section of this document.

TLDs: any

IPR Disclosure: none

Status: active

Notes: none

8. Security Considerations

Authorization information described in [RFC5733] is not supported in this document. If the querying client is not the sponsoring registrar of the reseller, not all the object information is accessible. The disclose element defined in [RFC5733] is used to allow or restrict disclosure of object elements to third parties. Other mechanism, such as defining a registry customized authorization information list according to their local policies and regulations, is also possible.

9. 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, <http://www.rfc-editor.org/info/rfc2119>.

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

[RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <http://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, <http://www.rfc-editor.org/info/rfc5733>.

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

[W3C.REC-xml-20040204]
Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E., and F. Yergeau, ""Extensible Markup Language (XML) 1.0 (Third Edition)", World Wide Web Consortium FirstEdition REC-xml-20040204", February 2004, <http://www.w3.org/TR/2004/REC-xml-20040204>.

[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>.

Appendix A. Change Log

Initial -00: Individual document submitted.

-01:

* Updated abstract text.
* Added sentences to avoid loop of parent identifiers in section 3.4.
* Revised typos in section 3.6.
* Added explanation of contact type attribute in section 4.1.2.
* Updated <info> responses.
* Deleted description of <transfer> command in section 4.1 and 4.2.
* Deleted whoisInfo disclose type in XML schema.
* Deleted maxOccurs of addRemType.
* Deleted extra "OPTIONAL" in section 4.2.5.
* Updated typos in <update> response.

-02:

* Changed author information.
* Updated url definition.
* Updated XML schema.

Authors' Addresses

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

Phone: +86 10 5881 2677
Email: zhoulinlin@cnnic.cn

Ning Kong
CNNIC
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Phone: +86 10 5881 3147
Email: nkong@cnnic.cn

Chao Qi
Teleinfo

Email: qichao911@163.com

Xiaodong Lee
CNNIC
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Phone: +86 10 5881 3020
Email: xl@cnnic.cn

James Gould
VeriSign, Inc.
12061 Bluemont Way
Reston, VA 20190
US

Email: jgould@verisign.com