

Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: May 16, 2016

G. Brown  
CentralNic Group plc  
J. Frakes  
November 13, 2015

**Command Reversal Extension for the Extensible Provisioning Protocol  
(EPP)  
draft-brown-epp-reverse-00**

Abstract

This document describes an Extensible Provisioning Protocol (EPP) extension mapping for reversing previous EPP commands.

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 May 16, 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.

Table of Contents

[1.](#) Introduction . . . . . [2](#)  
[1.1.](#) Conventions Used in This Document . . . . . [3](#)  
[2.](#) Extension Elements . . . . . [3](#)  
[3.](#) Server Handling of Reverse Commands . . . . . [4](#)  
[4.](#) EPP <poll> command . . . . . [4](#)  
[5.](#) Formal Syntax . . . . . [5](#)  
[6.](#) Security Considerations . . . . . [7](#)  
[7.](#) IANA Considerations . . . . . [8](#)  
[7.1.](#) XML Namespace . . . . . [8](#)  
[7.2.](#) EPP Extension Registry . . . . . [8](#)  
[8.](#) Acknowledgements . . . . . [8](#)  
[9.](#) Normative References . . . . . [8](#)  
 Authors' Addresses . . . . . [9](#)

**[1.](#) Introduction**

The Extensible Provisioning Protocol (EPP) provides a way for clients to create and update objects in a central repository. Usually, the commands that a client sends to a server will have been initiated upon request of a human being. As a result, occasionally a command is sent which contains an error.

EPP clients have some options to remedy such mistakes: for example, they can send a <delete> command to delete an object created in error (and may receive a refund if they do so within some grace period), cancel a previous <transfer> request, or send an <update> command to amend the properties of an object.

However, there are some circumstances where it is not possible to correct an error by using an existing command. For example, clients may inadvertently send multiple <renew> commands (because their implementation queries the server for a domain's expiry date, thereby defeating the idempotency measures built into EPP) or specify an incorrect period (e.g. a two-year renewal which should have only been for one year).

As another example, a client may perform an <update> command on an object, but keep no record of the previous state of the object, preventing them from correcting the error.

None of the mistakes in the examples above can be fixed using the existing EPP command repertoire. The extension described in this document attempts to provide an additional remedy for such cases, by providing a way for a client to request that a previous command be reversed. In order to reverse a command, the client need only record the <svTRID> returned by the server in its response to the command.



### **1.1. 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 [RFC 2119](#) [[RFC2119](#)].

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

"reverse" is used as an abbreviation for "urn:ietf:params:xml:ns:reverse-0.1". The XML namespace prefix "reverse" 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.

(Note to RFC Editor: remove the following paragraph before publication as an RFC.)

The XML namespace prefix above contains a version number, specifically "0.1". This version number will increment with successive versions of this document, and will reach 1.0 if and when this document is published as an RFC. This permits clients to distinguish which version of the extension a server has implemented.

## **2. Extension Elements**

This specification provides a new EPP command, called <reverse>. As the set of EPP command verbs cannot be updated without updating the core EPP specifications, this command is implemented as an extension.

When a client wants to reverse a previous command, it sends an EPP command frame containing only an <extension> element. The <extension> element contains a <reverse> element, which in turn contains the details of the command the client wishes to reverse. The <reverse> element has the following child elements:

An OPTIONAL <reason> element which contains a human-readable explanation of why the client is submitting the request. The server MAY require inclusion of a <reason> element depending on its own policy.

A <trID> element which uniquely identifies the command that the client wishes to reverse. The <trID> element is derived from the element of the same name in [[RFC5730](#)].



An OPTIONAL <clTRID> element that uniquely identifies this command to the server.

Example <reverse> request frame:

```
C: <?xml version="1.0" encoding="UTF-8"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <extension>
C:     <reverse:reverse
C:       xmlns:reverse="urn:ietf:params:xml:ns:reverse-0.1">
C:       <reverse:reason>Reversing an accidental
C:       double renewal.</reverse:reason>
C:       <reverse:trID>
C:         <reverse:clTRID>ABC-10001</reverse:clTRID>
C:         <reverse:svTRID>54321-XYZ</reverse:svTRID>
C:       </reverse:trID>
C:       <reverse:clTRID>ABC-12345</reverse:clTRID>
C:     </reverse:reverse>
C:   </extension>
C: </epp>
```

### **3. Server Handling of Reverse Commands**

Which commands a server will accept <reverse> commands for is a matter of server policy (which server operators should provide to client operators). If a server accepts a <reverse> command, it MUST respond with a 1000 or 1001 result code. If the server returns a 1001 response, the client MUST be notified of outcome of the offline process via the EPP message queue. If the server rejects the command, it MUST respond with a 2400 result code.

Servers MUST NOT allow a client to reverse a command that it did not originally submit. Servers SHOULD implement policies that flag <reverse> commands for offline processing where changes might have security implications (such as those which would remove client-assigned status codes, alter authorisation information or add or remove secDNS [[RFC5910](#)] records, etc).

Servers MUST reject a command which attempts to reverse a <reverse> command.

### **4. EPP <poll> command**

Servers which return 1001 response codes to <reverse> commands MUST notify clients of the outcome of the out-of-band process via the EPP message queue. The format of the <poll> message is as follows.



The <resData> element of the <poll> response contains a <panData> element which contains a <paTRID> element. This element contains an OPTIONAL <clTRID> and a <svTRID> which together identify the <reverse> command submitted by the client. The <paTRID> element has a MANDATORY "paResult" element which is a boolean, and which indicates the outcome of the offline review.

The <msg> element contains a human-readable message describing the outcome of the review.

Example of a <poll> message:

```
C: <?xml version="1.0" encoding="utf-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <response>
C:     <result code="1301">
C:       <msg>Command completed successfully; ack to dequeue</msg>
C:     </result>
C:     <msgQ count="5" id="12345">
C:       <qDate>2016-04-04T22:01:00.0Z</qDate>
C:       <msg>Pending action completed successfully.</msg>
C:     </msgQ>
C:     <resData>
C:       <reverse:panData
C:         xmlns:reverse="urn:ietf:params:xml:ns:reverse-0.1">
C:         <reverse:paTRID paResult="1">
C:           <reverse:clTRID>ABC-12345</reverse:clTRID>
C:           <reverse:svTRID>54321-XYZ</reverse:svTRID>
C:         </reverse:paTRID>
C:         <reverse:paDate>2016-04-04T22:00:00.0Z</reverse:paDate>
C:       </reverse:panData>
C:     </resData>
C:     <trID>
C:       <clTRID>BCD-23456</clTRID>
C:       <svTRID>65432-WXY</svTRID>
C:     </trID>
C:   </response>
C: </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.

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





Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- o Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- o Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- o Neither the name of Internet Society, IETF or IETF Trust, nor the names of specific contributors, may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BEGIN

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
  xmlns:reverse="urn:ietf:params:xml:ns:reverse-0.1"
  targetNamespace="urn:ietf:params:xml:ns:reverse-0.1"
  elementFormDefault="qualified">
  <annotation>
    <documentation>
      Extensible Provisioning Protocol v1.0 extension
      schema for command reverseal.
    </documentation>
  </annotation>

  <import
    namespace="urn:ietf:params:xml:ns:epp-1.0"
    schemaLocation="epp-1.0.xsd"/>

  <!-- command element -->
```



```
<element name="reverse" type="reverse:reverseType"/>

<complexType name="reverseType">
  <sequence>
    <element name="reason" type="epp:msgType"
      minOccurs="0" />
    <element name="trID" type="reverse:trIDType"/>
    <element name="clTRID" type="epp:trIDStringType"
      minOccurs="0" />
  </sequence>
</complexType>

<complexType name="trIDType">
  <sequence>
    <element name="clTRID" type="epp:trIDStringType"
      minOccurs="0" />
    <element name="svTRID" type="epp:trIDStringType" />
  </sequence>
</complexType>

<!-- poll response element -->
<element name="panData" type="reverse:panDataType" />

<complexType name="panDataType">
  <sequence>
    <element name="paTRID" type="reverse:paTRIDType" />
    <element name="paDate" type="dateTime" />
  </sequence>
</complexType>

<complexType name="paTRIDType">
  <complexContent>
    <extension base="reverse:trIDType">
      <attribute name="paResult" type="boolean" use="required" />
    </extension>
  </complexContent>
</complexType>

</schema>
END
```

## 6. Security Considerations

The mapping extensions described in this document do not provide any security services beyond those described by EPP [[RFC5730](#)], and protocol layers used by EPP. The security considerations described in these other specifications apply to this specification as well.



## **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](#)]. The following URI assignment is requested of IANA:

URI: urn:ietf:params:xml:ns:reverse-0.1

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: EPP Reverse 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. Acknowledgements**

The authors wish to thank Keith Gaughan for early feedback.

## **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>>.
- [RFC5910] Gould, J. and S. Hollenbeck, "Domain Name System (DNS) Security Extensions Mapping for the Extensible Provisioning Protocol (EPP)", [RFC 5910](#), DOI 10.17487/RFC5910, May 2010, <<http://www.rfc-editor.org/info/rfc5910>>.
- [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>>.

#### Authors' Addresses

Gavin Brown  
CentralNic Group plc  
35-39 Moorgate  
London, England EC2R 6AR  
GB

Phone: +44 20 33 88 0600  
Email: [gavin.brown@centralnic.com](mailto:gavin.brown@centralnic.com)  
URI: <https://www.centralnic.com>

Jothan Frakes

Email: [jothan@jothan.com](mailto:jothan@jothan.com)  
URI: <http://jothan.com>



