J. Mitchell
AusRegistry Pty Ltd
September 2010

Domain Name Variant Extension Mapping for the Extensible Provisioning Protocol (EPP)

Abstract

This document describes an Extensible Provisioning Protocol (EPP) extension mapping for the provisioning and management of domain name variants stored in a shared central repository. Specified in XML, this mapping extends the EPP domain name mapping to provide additional features required for the provisioning of domain name variants adhering to the Internationalized Domain Names in Applications (IDNA) standards

Status of This Document

This document specifies an extension to the EPP protocol first implemented in AusRegistry's Domain Name Registry EPP service. Please refer to AusRegistry for more information on the status of this document. Distribution of this document and use of the protocol extensions defined within is unrestricted and unlimited.

Copyright Notice

Table of Contents

TOC

1. Introduction

Internationalized Domain Names have introduced the ability to specify domain names in scripts beyond basic latin, allowing many languages and scripts to be coded as domain names. This however has led to forms of abuse, where malicious users exploit imperfections in the coding of languages in computers to register domain names with the intention of deceiving legitimate users.

Registries, language experts and linguists have collaborated to define variant domain names as names that should be blocked from registration independent of the original name. However, due to technological and contextual issues, some blocked variants should be activated and published in the DNS to provide satisfactory end user experience. This document considers only activated variants; the listing of possible variants and blocked variants of a domain name is out of scope of this extension mapping.

This document describes an extension mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) described in RFC 5730 **[RFC5730]**. This mapping, an extension of the domain name mapping described in RFC 5731 **[RFC5731]**, is specified using the Extensible Markup Language (XML) 1.0 **[W3C.REC-xml-20040204]** and XML Schema notation (**[W3C.REC-xmlschema-1-20041028] [W3C.REC-xmlschema-2-20041028]**).

The EPP core protocol specification **[RFC5730]** provides a complete description of EPP command and response structures. A thorough understanding of the base protocol specification is necessary to understand the mapping described in this document. Familiarity with the IDNA protocol described in **[RFC5890]**, **[RFC5891]**, **[RFC5892]**, **[RFC5893]**, and **[RFC5894]** is required to understand the concepts described in this document.

TOC

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 **[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 mandatory feature of this protocol.

TOC

2. Object Attributes

This extension adds additional elements to the EPP domain name mapping **[RFC5731]**. Only those new elements are described here.

TOC

2.1. Variant Domain Name

Internationalized Domain Names in Applications introduces a mechanism to convert a domain name expressed in Unicode to an ASCII-compatible encoding (ACE) form that is a valid DNS label compatible with existing applications and infrastructure. This mechanism gives rise to two representations of the domain name containing:

- one or more A-labels with zero or more non-internationalized labels
- one or more U-labels with zero or more non-internationalized labels

This mapping requires the client supply both the A-label and U-label representations of the variant domain name to remove any ambiguity between server and client as to the name being activated.

The <variant> element in this document maps to the A-label representation of the domain name and MUST NOT contain any U-labels. The associated "userForm" attribute maps to the U-label representation of the domain name and MUST NOT contain any A-labels.

TOC

3. 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 implementing IDNA domain provisioning processes via EPP.

TOC

3.1. EPP Query Commands

EPP provides three commands to retrieve object information: <check> to determine if an object is known to the server, <info> to retrieve detailed information associated with an object, and <transfer> to retrieve object transfer status information.

TOC

3.1.1. EPP <check> Command

This extension does not add any elements to the EPP <check> command or <check> response described in the EPP domain mapping **[RFC5731]**.

3.1.2. EPP <info> Command

TOC

This extension adds elements to the EPP <info> command as well as the EPP <info> response, both described in the EPP domain mapping [1]. The response to this command MAY vary depending on the identity of the querying client, use of authorisation information, and server policy towards unauthorised clients.

In addition to the EPP command elements described in the EPP domain mapping [1], the <info> command MUST contain an EPP <extension> element, which MUST contain an <info> element that identifies the variant namespace. The <info> element contains the following:

- An optional "variants" attribute that indicates whether the response to the info command should list all variants of the domain. Allowable values for this attribute are "all", or "none", with the default value being "all". If "none" is specified, then no variants are listed in the response. If "all" is specified, or the attribute is not present, then all variants are listed in the response.

Example <info> command:

```
C:<?xml version="1.0" encoding="UTF-8"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <info>
C:      <info xmlns="urn:ietf:params:xml:ns:domain-1.0">
C:        <name>xn--3xan.example</name>
C:      </info>
C:    </info>
C:    <extension>
C:      <info xmlns="urn:X-ar:params:xml:ns:variant-1.0"
C:        variants="all"/>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C: </epp>
```

When an <info> command has been successfully processed, the EPP <resData> element in the <info> response MUST contain child elements as described in the EPP domain mapping **[RFC5731]**. In addition, the <info> response MUST contain an EPP <extension> element, which MUST contain an <infData> element that identifies the variant namespace. The <infData> element contains the following child elements:

- One or more <variant> elements containing the fully qualified name of any variants of a domain name, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name.

Example <info> response:

```
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg lang="en">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <infData xmlns="urn:ietf:params:xml:ns:domain-1.0">
S:        <name>xn--3xan.example</name>
S:        <roid>EXAMPLE1-REP</roid>
S:        <status s="ok" />
S:        <registrant>jd1234</registrant>
S:        <contact type="admin">sh8013</contact>
S:        <contact type="tech">sh8013</contact>
```

```
S:        <ns>
S:          <hostObj>ns1.example.com</hostObj>
S:          <hostObj>ns1.example.net</hostObj>
S:        </ns>
S:        <clID>ClientX</clID>
S:        <crID>ClientY</crID>
S:        <crDate>1999-04-03T22:00:00.0Z</crDate>
S:        <upID>ClientX</domain:upID>
S:        <upDate>1999-12-03T09:00:00.0Z</upDate>
S:        <exDate>2005-04-03T22:00:00.0Z</exDate>
S:        <trDate>2000-04-08T09:00:00.0Z</trDate>
S:        <authInfo>
S:          <pw>2fooBAR</pw>
S:        </authInfo>
S:      </infData>
S:    </resData>
S:    <extension>
S:      <infData xmlns="urn:X-ar:params:xml:ns:variant-1.0">
S:        <variant userForm="&#969;&#963;.example">
S:          xn--4xal.example
S:        </variant>
S:      </infData>
S:    </extension>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54321-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

TOC

3.1.3. EPP <transfer> Command

This extension does not add any elements to the EPP <transfer> command or <transfer> response described in the EPP domain mapping **[RFC5731]**.

TOC

3.2. EPP Transform Commands

EPP provides five commands to transform 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 object sponsorship changes, and <update> to change information associated with an object.

TOC

3.2.1. EPP <create> Command

This extension does not add any elements to the EPP <create> command, however it does add extensions to the EPP <create> response. Both are described in the EPP domain mapping **[RFC5731]**.

When a <create> command has been successfully processed, the EPP <resData> element in the <create> response MUST contain child elements as described in the EPP domain mapping **[RFC5731]**. In addition, the <create> response MUST contain an EPP <extension> element, which MUST contain an <creData> element identifying the variant namespace. The <creData> contains the following child elements:

- One or more <variant> elements containing the fully qualified name of any variants of a domain name, in DNS form. Each <variant> element MUST have a userForm attribute indicating the native presentation form of the variant name.

Example <create> response:

```
S:<?xml version="1.0" encoding="UTF-8"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg lang="en">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <creData xmlns="urn:ietf:params:xml:ns:domain-1.0">
S:        <name>xn--3xan.example</name>
S:        <crDate>1999-04-03T22:00:00.0Z</crDate>
S:        <exDate>2001-04-03T22:00:00.0Z</exDate>
S:      </creData>
S:    </resData>
S:    <extension>
S:      <creData xmlns="urn:X-ar:params:xml:ns:variant-1.0">
S:        <variant userForm="&#969;&#963;.example">
S:          xn--4xal.example
S:        </variant>
S:      </creData>
S:    </extension>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54321-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

TOC

3.2.2. EPP <delete> Command

This extension does not add any elements to the EPP <delete> command or <delete> response described in the EPP domain mapping **[RFC5731]**.

TOC

3.2.3. EPP <renew> Command

This extension does not add any elements to the EPP <renew> command or <renew> response described in the EPP domain mapping **[RFC5731]**.

TOC

3.2.4. EPP <transfer> Command

This extension does not add any elements to the EPP <transfer> command or <transfer> response described in the EPP domain mapping **[RFC5731]**.

TOC

3.2.5. EPP Command

This extension adds elements to the EPP <update> command described in the EPP domain mapping **[RFC5731]**. In addition to the command elements in the domain mapping, the <update> command MUST contain an EPP <extension> element, which MUST contain an <update> element that identifies the variant namespace. The <update> element contains the following elements:

- An OPTIONAL <add> element that contains the domain variants to be activated
- An OPTIONAL <rem> element that contains the domain variants to be deactivated.

At least one <add> and <rem> element MUST be provided. The <add> and <rem> elements MUST contain the following elements

- One or more <variant> elements, each containing the fully qualified name of a domain variant, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name

Example <update> Command, Activate a Variant:

```
C:<?xml version="1.0" encoding="UTF-8"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <update>
C:      <update xmlns="urn:ietf:params:xml:ns:domain-1.0">
C:        <name>xn--3xan.example</name>
C:      </update>
C:    </update>
C:    <extension>
C:      <update xmlns="urn:X-ar:params:xml:ns:variant-1.0">
C:        <add>
C:          <variant userForm="&#969;&#963;.example">
C:            xn--4xal.example
C:          </variant>
C:        </add>
C:      </update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> Command, Remove a Variant:

```
C:<?xml version="1.0" encoding="UTF-8"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <update>
C:      <update xmlns="urn:ietf:params:xml:ns:domain-1.0">
C:        <name>xn--3xan.example</name>
C:      </update>
C:    </update>
C:    <extension>
C:      <update xmlns="urn:X-ar:params:xml:ns:variant-1.0">
C:        <rem>
C:          <variant userForm="&#969;&#963;.example">
C:            xn--4xal.example
C:          </variant>
C:        </rem>
C:      </update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

TOC

4. 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:X-ar:params:xml:ns:variant-1.1"
  xmlns:variant="urn:X-ar:params:xml:ns:variant-1.1"
  xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
  xmlns:eppcom="urn:ietf:params:xml:ns:eppcom-1.0"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

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


  <annotation>
    <documentation>
      Internationalised Domain Name Extensions to the Extensible
      Provisioning Protocol v1.0 schema.
    </documentation>
  </annotation>

  <!--
    Custom command extensions
  -->
  <element name="info" type="variant:infoType" />
  <element name="update" type="variant:updateType" />

  <!--
    Child elements of the <info> command extension.
  -->
  <complexType name="infoType">
    <attribute name="variants" type="variant:infoAttrType"
      default="all" />
  </complexType>

  <simpleType name="infoAttrType">
    <restriction base="token">
      <enumeration value="all" />
      <enumeration value="none" />
    </restriction>
  </simpleType>

  <!--
    Child elements of the <update> command extension.
  -->
  <complexType name="updateType">
    <sequence>
      <element name="add" type="variant:addRemType" minOccurs="0" />
      <element name="rem" type="variant:addRemType" minOccurs="0" />
    </sequence>
  </complexType>

  <complexType name="addRemType">
    <sequence>
      <element name="variant" type="variant:variantType"
        maxOccurs="unbounded" />
    </sequence>
  </complexType>

  <complexType name="variantType">
    <simpleContent>
      <extension base="eppcom:labelType">
        <attribute name="userForm" type="eppcom:labelType"
          use="required" />
      </extension>
    </simpleContent>
  </complexType>

  <!--
    Response extension elements.
  -->
  <element name="infData" type="variant:resDataType" />
  <element name="creData" type="variant:resDataType" />

  <complexType name="resDataType">
    <sequence>
      <element name="variant" type="variant:variantType"
        maxOccurs="unbounded" />
    </sequence>
  </complexType>

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

TOC

5. 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 **[RFC3629]**. Conformant XML processors recognize both UTF-8 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 in environments where parser encoding support incompatibility exists.

As an extension of the EPP domain mapping **[RFC5731]**, the internationalization requirements in the EPP domain mapping **[RFC5731]** are followed by this extension. This extension does not override any of the EPP domain mapping **[RFC5731]** internationalization features.

TOC

6. Security Considerations

The mapping extensions described in this document do not provide any security services beyond those described by EPP **[RFC5730]**, the EPP domain name mapping **[RFC5731]**, and protocol layers used by EPP. The security considerations described in these other specifications apply to this specification as well.

As with other domain object transforms, the EPP transform operations described in this document MUST be restricted to the sponsoring client as authenticated using the mechanisms described in Sections 2.9.1.1 and 7 of **[RFC5730]**. Any attempt to perform a transform operation on a domain object by any client other than the sponsoring client MUST be rejected with an appropriate EPP authorization error.

TOC

7. References

TOC

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).

[RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)," STD 69, RFC 5730, August 2009 (TXT).

[RFC5731] Hollenbeck, S., "Extensible Provisioning Protocol (EPP) Domain Name Mapping," STD 69, RFC 573 August 2009 (TXT).

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Documen Framework," RFC 5890, August 2010 (TXT).

[RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol," RFC 5891, August 2010 (TXT).

[RFC5892] Faltstrom, P., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)," RFC 5892, August 2010 (TXT).

[RFC5893] Alvestrand, H. and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)," RFC 5893, August 2010 (TXT).

[RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation and Rationale," RFC 5894, August 2010 (TXT).

[W3C.REC-xml-20040204] Bray, T., Maler, E., Paoli, J., Yergeau, F., and C. Sperberg-McQueen, "Extensible Markup Language (XML) 1.0 (Third Edition)," World Wide Web Consortium FirstEdition REC-xml-20040204, February 200 (HTML).

[W3C.REC-xmlschema-1-20041028] Beech, D., Thompson, H., Mendelsohn, N., and M. Maloney, "XML Schema Part 1: Structures Second Edition," World Wide Web Consortium Recommendation REC-xmlschema-1-20041028, October 2004 (HTML).

[W3C.REC-xmlschema-2-20041028] Malhotra, A. and P. Biron, "XML Schema Part 2: Datatypes Second Edition," World Wide Web Consortium Recommendation REC-xmlschema-2-20041028, October 2004 (HTML).

TOC

7.2. Informative References

[RFC2781] Hoffman, P. and F. Yergeau, "UTF-16, an encoding of ISO 10646," RFC 2781, February 2000 (TXT).

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646," STD 63, RFC 3629, November 2003 (TXT).

TOC

Author's Address

James Mitchell
AusRegistry Pty Ltd
Email: support@ausregistry.com.au