

# Promotion EPP Extension for the .tld TLD

CORE Association

## 1. Extension for Promotion Codes

The CORE Registration System provides a proprietary EPP extension for the use of promotion codes when creating or renewing domains. Promotion codes are issued by the registry and enable registrants to benefit from specially priced domain creation and renewal offers.

### 1.1 Introduction

This document describes an extension mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) described in RFC 5730. This mapping is an extension of the domain name mapping described in RFC 5731. It is specified using the Extensible Markup Language (XML) and XML Schema notation.

This extension serves the purpose of specifying promotion codes issued by the registry in order to achieve special, promotional pricing for domain create or renew operations.

### 1.2 EPP Command Mapping

This section deals with the specific command mappings for the EPP extension for promotion codes.

In the following, the respective root elements of the extensions are mentioned. If used, they must be placed or expected within the optional <extension> element at the proper location in the XML document representing the EPP command or response, as described in RFC 5730. Note that the use of the "promo:" XML namespace prefix is for documentation purposes only. Conforming to the "Namespaces in XML 1.1" standard, EPP and the registry implementation take only the associated namespace URI into account, and not the prefix itself. So actually any prefix or even the default namespace may be used in requests and must be expected in responses.

The promotion code extension is only used in relation to domain objects. It will not occur in commands that are related to host and contact objects.

#### 1.2.1 EPP Query Commands

There are four EPP commands to retrieve object information: <check> to find out whether an object is known to the server, <info> to ask for detailed information associated with an object, <poll> to discover and retrieve queued service messages for individual clients and <transfer> to get transfer status information for an object.

#### **1.2.1.1 EPP <domain:check> Command**

This extension does not add any element to the EPP <check> command.

#### **1.2.1.2 EPP <domain:info> Command**

This extension does not add any element to the EPP <info> command.

#### **1.2.1.3 EPP <poll> Command**

This extension does not add any element to the EPP <poll> command.

#### **1.2.1.4 EPP <transfer> Query Command**

This extension does not add any element to the EPP <transfer> query command.

### **1.2.2 EPP Transform Commands**

There are five EPP 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.

#### **1.2.2.1 EPP <domain:create> Command**

The create command, which allows the registration of domain objects, or, during a registry's sunrise and landrush phases, the application for domain objects, can be augmented by a <promo:create> element in the extension section of the command that identifies the promotion extension namespace. The <promo:create> element must contain a <promo:code> element containing the promotion code specified by the registrant (which he has received from the registry to participate in a promotion campaign). If the content matches a suitable promotion code in the registry, the domain creation will be performed at a special price.

An example of a <create> command using the extension can be found in the "Examples" section below.

#### **1.2.2.2 EPP <domain:delete> Command**

There are no extension elements for the domain delete command and response.

### 1.2.2.3 EPP <domain:renew> Command

The renew command, which allows the renewal of the validity period of domain objects, can be augmented by a <promo:renew> element in the extension section of the command that identifies the promotion extension namespace. The <promo:renew> element must contain a <promo:code> element containing the promotion code specified by the registrant (which he has received from the registry to participate in a promotion campaign). If the content matches a suitable promotion code in the registry, the domain renewal will be performed at a special price.

An example of a <renew> command using the extension can be found in the "Examples" section below.

### 1.2.2.4 EPP <domain:transfer> Command

There are no extension elements for the transfer command and response.

### 1.2.2.5 EPP <domain:update> Command

There are no extension elements for the update command and response.

## 1.3 Formal Syntax (Schema Definition)

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://xmlns.corenic.net/epp/promotion-1.0"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:promo="http://xmlns.corenic.net/epp/promotion-1.0"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <annotation>
    <documentation>
      Extensible Provisioning Protocol v1.0
      extension schema for specifying promotion codes in
      domain create and renew commands
    </documentation>
  </annotation>

  <!-- child elements found in EPP commands -->

  <element name="create" type="promo:createType"/>
  <element name="renew" type="promo:renewType"/>

  <!-- create extension -->

  <complexType name="createType">
    <sequence>
      <element name="code" type="promo:promotionCodeType"/>
    </sequence>
  </complexType>
```

```

<!-- renew extension -->

<complexType name="renewType">
  <sequence>
    <element name="code" type="promo:promotionCodeType"/>
  </sequence>
</complexType>

<!-- shared types -->

<simpleType name="promotionCodeType">
  <restriction base="token">
    <minLength value="1"/>
    <maxLength value="64"/>
  </restriction>
</simpleType>

</schema>

```

## 1.4 Examples

In the following examples, "C:" represents lines sent by an EPP client and "S:" represents lines returned by the EPP server.

### 1.4.1 EPP <create> Command

#### 1.4.1.1 Example <create> command specifying a promotion code:

```

C: <?xml version="1.0" encoding="UTF-8"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <create>
C:       <create xmlns="urn:ietf:params:xml:ns:domain-1.0">
C:         <name>example.tld</name>
C:         <period unit="y">1</period>
C:         <ns>
C:           <hostObj>ns1.example.net</hostObj>
C:           <hostObj>ns2.example.net</hostObj>
C:         </ns>
C:         <registrant>abc123</registrant>
C:         <contact type="admin">def456</contact>
C:         <contact type="tech">ghi789</contact>
C:         <authInfo>
C:           <pw>secret42</pw>
C:         </authInfo>
C:       </create>
C:     </create>
C:     <extension>
C:       <promo:create xmlns:promo="http://xmlns.corenic.net/epp/promotion-1.0">
C:         <promo:code>EXAMPLE-PROMO-1231</promo:code>
C:       </promo:create>
C:     </extension>
C:     <clTRID>abc-00042</clTRID>
C:   </command>
C: </epp>

```

## 1.4.2 EPP <renew> Command

### 1.4.2.1 Example <renew> command specifying a promotion code:

```
C:<?xml version="1.0" encoding="UTF-8"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <renew>
C:      <renew xmlns="urn:ietf:params:xml:ns:domain-1.0">
C:        <name>example.tld</name>
C:        <curExpDate>2015-04-03</curExpDate>
C:        <period unit="y">1</period>
C:      </renew>
C:    </renew>
C:  <extension>
C:    <promo:renew xmlns:promo="http://xmlns.corenic.net/epp/promotion-1.0">
C:      <promo:code>EXAMPLE-PROMO-1231</promo:code>
C:    </promo:renew>
C:  </extension>
C:  <c1TRID>abc-00042</c1TRID>
C: </command>
C:</epp>
```