

RFC 4931 : Extensible Provisioning Protocol (EPP) Domain Name Mapping

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 17 Juin 2007

Date de publication du RFC : Mai 2007

<http://www.bortzmeyer.org/4931.html>

Le protocole d'avitaillement EPP ne spécifie pas comment représenter les objets qu'on crée, détruit, modifie, etc. Cette tâche est déléguée à des RFC auxiliaires comme le nôtre (remplacé depuis par le RFC 5731¹), consacré aux noms de domaine.

EPP permet à un **client** de créer, détruire et modifier des objets de types différents. En pratique, EPP n'est utilisé que dans l'industrie des noms de domaine mais, en théorie, il pourrait être utilisé pour n'importe quel type d'objets.

Le type n'est donc pas spécifié dans le protocole EPP de base, normalisé dans le RFC 4930, mais dans des RFC supplémentaires. Par exemple, celui qui fait l'objet de cet article spécifie le type (EPP dit le "*mapping*") pour les noms de domaine.

Ce type est spécifié (section 4 du RFC) dans le langage W3C XML Schema.

On note que ce schéma, se voulant adaptable à de nombreux registres différents, ne colle parfaitement à la politique d'aucun. Par exemple, `<infData>` est spécifié ainsi :

1. Pour voir le RFC de numéro NNN, <http://www.ietf.org/rfc/rfcNNN.txt>, par exemple <http://www.ietf.org/rfc/rfc5731.txt>

```

<sequence>
  <element name="registrant" type="eppcom:clIDType"
    minOccurs="0"/>
  <element name="contact" type="domain:contactType"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="ns" type="domain:nsType"
    minOccurs="0"/>

```

ce qui veut dire que des « règles métier » très courantes comme « un titulaire et un seul » ou bien « au moins deux serveurs de noms » ne sont pas respectés avec le schéma EPP seul.

Ce RFC remplace son prédécesseur, le RFC 3731 mais ce ne sont que des modifications de détail. Lui-même a ensuite été remplacé par le RFC 5731.

Voici un exemple d'une réponse à une requête `<epp:info>` pour le domaine `example.com` :

```

<response>
  <result code="1000">
    <msg>Command completed successfully</msg>
  </result>
  <resData>
    <domain:infData
      xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
      <domain:name>example.com</domain:name>
      <domain:roid>EXAMPLE1-REP</domain:roid>
      <domain:status s="ok"/>
      <domain:registrant>jd1234</domain:registrant>
      <domain:contact type="admin">sh8013</domain:contact>
      <domain:contact type="tech">sh8013</domain:contact>
      <domain:ns>
        <domain:hostObj>ns1.example.com</domain:hostObj>
        <domain:hostObj>ns1.example.net</domain:hostObj>
      </domain:ns>
      <domain:host>ns1.example.com</domain:host>
      <domain:host>ns2.example.com</domain:host>
      <domain:clID>ClientX</domain:clID>
      <domain:crID>ClientY</domain:crID>
      <domain:crDate>1999-04-03T22:00:00.0Z</domain:crDate>
      <domain:upID>ClientX</domain:upID>
      <domain:upDate>1999-12-03T09:00:00.0Z</domain:upDate>
      <domain:exDate>2005-04-03T22:00:00.0Z</domain:exDate>
      <domain:trDate>2000-04-08T09:00:00.0Z</domain:trDate>
      <domain:authInfo>
        <domain:pw>2fooBAR</domain:pw>
      </domain:authInfo>
    </domain:infData>
  </resData>
  ...

```

On y voit les serveurs de noms du domaine, `ns1.example.com` et `ns1.example.net`, son titulaire, identifié par un code, ici `jd1234`, sa date de création, le 3 avril 1999, etc.

De même, pour créer un domaine, on appelle `<epp:create>` avec des éléments de l'espace de nom `urn:ietf:params:xml:ns:domain-1.0`, par exemple :

```
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.com</domain:name>
        <domain:period unit="y">2</domain:period>
        <domain:ns>
          <domain:hostObj>ns1.example.com</domain:hostObj>
          <domain:hostObj>ns1.example.net</domain:hostObj>
        </domain:ns>
        <domain:registrant>jd1234</domain:registrant>
      </domain:create>
    </create>
  </command>
</epp>
...
```