

Groupe de travail Réseau
Request for Comments : 5731
STD : 69
RFC rendue obsolète : 4931
Catégorie : Norme

S. Hollenbeck, VeriSign, Inc.
août 2009

Traduction Claude Brière de L'Isle

Protocole d'approvisionnement extensible (EPP) Transposition de nom de domaine

Résumé

Le présent document décrit une transposition de protocole d'approvisionnement extensible (EPP, *Extensible Provisioning Protocol*) pour l'approvisionnement et la gestion des noms de domaine de l'Internet mémorisés dans un répertoire central partagé. Spécifiée en XML, la transposition définit la syntaxe et la sémantique des commandes XML appliquées aux noms de domaines. Le présent document rend obsolète la RFC 4931.

Statut de ce mémoire

Le présent document spécifie un protocole Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et des suggestions pour son amélioration. Prière de se reporter à l'édition actuelle du STD 1 "Normes des protocoles officiels de l'Internet" pour connaître l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de copyright

Copyright (c) 2009 IETF Trust et les personnes identifiées comme auteurs du document. Tous droits réservés.

Le présent document est soumis au BCP 78 et aux dispositions légales de l'IETF Trust sur les documents de l'IETF (<http://trustee.ietf.org/license-info>) en vigueur à la date de publication du présent document. Prière de relire attentivement ces documents, car ils décrivent vos droits et obligations à l'égard du présent document.

Table des Matières

1. Introduction.....	2
1.1 Relations des objets de domaine et des objets d'hôte.....	2
1.2 Conventions de notation.....	3
2. Attributs d'objet	3
2.1 Noms de domaine et d'hôte	3
2.2 Identifiants de contact et de client.....	3
2.3 Valeurs d'état.....	3
2.4 Dates et heures.....	5
2.5 Périodes de validité.....	5
2.6 Informations d'autorisation.....	5
2.7 Autres attributs d'enregistrement de ressource du DNS.....	5
3. Transposition de commande EPP.....	5
3.1 Commande d'interrogation EPP.....	5
3.2 Commandes de transformation EPP.....	11
3.3 Révision hors ligne d'actions demandées.....	17
4. Syntaxe formelle.....	19
5. Considérations d'internationalisation.....	25
6. Considérations relatives à l'IANA.....	26
7. Considérations pour la sécurité.....	26
8. Remerciements.....	26
9. Références.....	27
9.1 Références normatives.....	27
9.2 Références pour information.....	27
Adresse de l'auteur.....	28

1. Introduction

Le présent document décrit une transposition de nom de domaine Internet pour la version 1.0 du protocole d'approvisionnement extensible (EPP, *Extensible Provisioning Protocol*). Cette transposition est spécifiée en utilisant le langage de balisage extensible (XML, *Extensible Markup Language*) 1.0 tel que décrit dans [xml-20040204] et la notation de schéma XML telle que décrite dans [xmldata-20041028] et [xmldata-20041028]. Le présent document rend obsolète la [RFC4931].

La [RFC5730] donne une description complète des structures de commande et de réponse d'EPP. Une bonne compréhension de la spécification du protocole de base est nécessaire pour comprendre la transposition décrite dans le présent document.

XML est sensible à la casse. Sauf mention contraire, les spécifications et exemples XML fournis dans le présent document DOIVENT être interprétés dans la casse de caractères présentée pour développer une mise en œuvre conforme.

1.1 Relations des objets de domaine et des objets d'hôte

La transposition EPP pour les objets d'hôte est décrite dans la [RFC5732]. Le présent document suppose que les objets de nom de domaine ont une relation supérieure avec les objets de nom d'hôte subordonnés. Par exemple, le nom de domaine "example.com" a une relation supérieure avec le nom d'hôte "ns1.example.com". Les actions EPP (telles que les transferts d'objet) qui ne préservent pas cette relation DOIVENT être explicitement interdites.

Un objet de nom d'hôte peut être créé dans un répertoire pour lequel aucun objet de nom de domaine supérieur n'existe. Par exemple, un nom d'hôte "ns1.example.com" peut être créé dans le répertoire ".example" afin que les domaines DNS dans ".example" puissent être délégués à l'hôte. De tels hôtes sont décrits comme des hôtes "externes" dans la présente spécification car le nom de l'hôte n'appartient pas à l'espace de noms du répertoire dans lequel l'hôte est utilisé pour les besoins de la délégation.

Qu'un hôte soit externe ou interne se rapporte au répertoire dans lequel est utilisé l'hôte pour les besoins de délégation. Qu'un hôte interne soit ou non subordonné se rapporte à un domaine au sein du répertoire. Par exemple, l'hôte ns1.example1.com est un hôte subordonné d'un domaine example1.com, mais il n'est pas un hôte subordonné du domaine example2.com. ns1.example1.com peut être utilisé comme serveur de noms pour example2.com. Dans ce cas, ns1.example1.com DOIT être traité comme un hôte interne, sous réserve des règles qui gouvernent les opérations sur les hôtes subordonnés au sein du même répertoire.

Les hôtes serveurs de noms pour la délégation de domaine peuvent être spécifiés soit comme références à des objets hôtes existants soit comme attributs de domaine qui décrivent une machine hôte. Un opérateur de serveur DOIT utiliser de façon cohérente une forme de spécification de nom de serveur. Un opérateur de serveur qui annonce la prise en charge des objets d'hôte dans un salut EPP NE DOIT PAS permettre que des attributs de domaine décrivent une machine hôte de serveur de noms. Un opérateur de serveur qui n'annonce pas la prise en charge des objets hôtes DOIT permettre que les attributs de domaine décrivent la machine hôte de serveur de noms. Lorsque des attributs de domaine sont utilisés pour décrire une machine hôte de serveur de noms, des adresses IP DEVRAIENT être exigées seulement comme nécessaire pour générer des enregistrements DNS glues.

Les serveurs de noms sont spécifiés au sein d'un élément <domain:ns>. Cet élément DOIT contenir un ou plusieurs éléments <domain:hostObj> ou un ou plusieurs éléments <domain:hostAttr>. Un élément <domain:hostObj> contient le nom pleinement qualifié d'un objet hôte de serveur de noms connu. Un élément <domain:hostAttr> contient les éléments fils suivants :

- Un élément <domain:hostName> qui contient le nom pleinement qualifié d'un hôte.
- Zéro, un ou plusieurs éléments FACULTATIFS <domain:hostAddr> qui contiennent les adresses IP à associer à l'hôte. Chaque élément PEUT contenir un attribut "ip" pour identifier le format d'adresse IP. La valeur d'attribut "v4" est utilisée pour noter le format d'adresse IP v4. La valeur d'attribut "v6" est utilisée pour noter le format d'adresse IP v6. Si l'attribut "ip" n'est pas spécifié, "v4" est la valeur d'attribut par défaut. Les exigences de syntaxe d'adresse IP sont décrites au paragraphe 2.5 de la transposition d'hôte EPP [RFC5732].

Exemple d'élément de serveur de nom d'hôte-objet pour le domaine example.com :

```
<domain:ns>  
<domain:hostObj>ns1.example.net</domain:hostObj>  
<domain:hostObj>ns2.example.net</domain:hostObj>
```

```
</domain:ns>
```

Exemple d'élément de serveur de nom d'hôte-attribut pour le domaine example.com :

```
<domain:ns>
  <domain:hostAttr>
    <domain:hostName>ns1.example.net</domain:hostName>
    <domain:hostAddr
      ip="v4">192.0.2.2</domain:hostAddr>
    <domain:hostAddr
      ip="v6">1080:0:0:0:8:800:200C:417A</domain:hostAddr>
  </domain:hostAttr>
  <domain:hostAttr>
    <domain:hostName>ns2.example.net</domain:hostName>
  </domain:hostAttr>
</domain:ns>
```

1.2 Conventions de notation

Les mots clés "DOIT", "NE DOIT PAS", "EXIGÉ", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" dans ce document sont à interpréter comme décrit dans la [RFC2119].

Dans les exemples, "C:" représente les lignes envoyées par un client de protocole et "S:" représente les lignes retournées par un serveur du protocole. Les sauts à la ligne et les espaces ne sont fournis dans les exemples que pour illustrer les relations entre les éléments et ne sont pas des caractéristiques EXIGÉES de ce protocole. Un client de protocole qui est autorisé à gérer un objet existant est décrit comme un client "parrain" dans le présent document.

2. Attributs d'objet

Un objet de domaine EPP a des attributs et des valeurs associées qui peuvent être vues et modifiées par le client parrain ou le serveur. Cette section décrit en détail chaque type d'attribut. La syntaxe formelle des valeurs d'attribut décrite ici se trouve dans la section "Syntaxe formelle" du présent document et dans les références normatives appropriées.

2.1 Noms de domaine et d'hôte

La syntaxe des noms de domaines et d'hôtes décrite dans le présent document DOIT se conformer à la [RFC0952] et à la [RFC1123]. Au moment de la présente rédaction, la [RFC3490] décrit une norme d'utilisation de certaines étiquettes de nom ASCII pour représenter des étiquettes de noms non ASCII. Ces exigences de conformité pourraient changer par suite des progrès des travaux de développement de normes pour les noms de domaine internationalisés. Un serveur PEUT restreindre les noms de domaine permis à un domaine de niveau supérieur particulier, à un domaine de second niveau, ou à d'autres domaines pour lesquels le serveur est d'autorité. Le point en queue exigé lorsque ces noms sont mémorisés dans une zone du DNS est implicite et NE DOIT PAS être fourni lors des échanges de noms d'hôte et de domaines.

2.2 Identifiants de contact et de client

Tous les contacts EPP sont identifiés par un identifiant unique pour le serveur.

Les identifiants de contact sont des chaînes de caractères d'une longueur minimum et maximum, et d'un format spécifiés. Les identifiants de contact utilisent la syntaxe d'identifiant de client "clIDType" décrite dans la [RFC5730].

2.3 Valeurs d'état

Un objet de domaine DOIT toujours avoir au moins une valeur d'état associée. Les valeurs d'état ne peuvent être réglées que par le client qui parraine un objet de domaine et par le serveur sur lequel réside l'objet. Un client peut changer l'état d'un objet de domaine en utilisant la commande EPP <update>. Chaque valeur d'état PEUT être accompagnée d'une chaîne de texte lisible par l'homme qui décrit la raison de l'état appliqué à l'objet.

Un client NE DOIT PAS altérer les valeurs d'état réglées par le serveur. Un serveur PEUT altérer ou outrepasser les valeurs d'état réglées par un client, sous réserve des politiques de serveur locales. L'état d'un objet PEUT changer par suite d'une commande de transformation initiée par un client ou par une action effectuée par un opérateur de serveur.

Les valeurs d'état qui peuvent être ajoutée ou retirées par un client sont munies du préfixe "client". Les valeurs d'état correspondantes qui peuvent être ajoutées ou retirées par un serveur sont munies du préfixe "server". Les valeurs d'état qui ne commencent pas par "client" ou "server" sont gérées par le serveur.

Descriptions des valeurs d'état :

- clientDeleteProhibited, serverDeleteProhibited
Les demandes de suppression de l'objet DOIVENT être rejetées.
- clientHold, serverHold
Les informations de délégation du DNS NE DOIVENT PAS être publiées pour l'objet.
- clientRenewProhibited, serverRenewProhibited
Les demandes de renouvellement de l'objet DOIVENT être rejetées.
- clientTransferProhibited, serverTransferProhibited
Les demandes de transfert de l'objet DOIVENT être rejetées.
- clientUpdateProhibited, serverUpdateProhibited
Les demande de mise à jour de l'objet (autres que de supprimer cet état) DOIVENT être rejetées.
- inactive
Les informations de délégation n'ont pas été associées à cet objet. C'est l'état par défaut lorsque un objet de domaine est créé pour la première fois et qu'il n'y a pas d'objet hôte associé pour la délégation du DNS. Cet état peut aussi être établi par le serveur lorsque toutes les associations hôte-objet sont supprimées.
- ok
C'est la valeur d'état normale pour un objet qui n'a pas d'opération ou interdiction en cours. Cette valeur est établie et supprimée par le serveur lorsque d'autres valeurs d'état sont ajoutées ou supprimées.
- pendingCreate, pendingDelete, pendingRenew, pendingTransfer, pendingUpdate
Une commande de transformation a été traitée pour l'objet, mais l'action n'a pas été achevée par le serveur. Les opérateurs de serveurs peuvent retarder l'achèvement de l'action pour diverses raisons, comme de permettre une révision par l'homme ou l'action d'un tiers. Une commande de transformation qui est traitée, mais dont l'action demandée est en cours, est notée avec le code de réponse 1001.

Lorsque l'action demandée a été achevée, la valeur d'état pendingCreate, pendingDelete, pendingRenew, pendingTransfer, ou pendingUpdate DOIT être supprimée. Tous les clients impliqués dans la transaction DOIVENT en être notifiés en utilisant un message de service qui dit que l'action a été achevée et que l'état de l'objet a changé.

L'état "ok" NE DOIT PAS être combiné avec un autre état.

L'état "pendingDelete" NE DOIT PAS être combiné avec l'état "clientDeleteProhibited" ou "serverDeleteProhibited".

L'état "pendingRenew" NE DOIT PAS être combiné avec l'état "clientRenewProhibited" ou "serverRenewProhibited".

L'état "pendingTransfer" NE DOIT PAS être combiné avec l'état "clientTransferProhibited" ou "serverTransferProhibited".

L'état "pendingUpdate" NE DOIT PAS être combiné avec l'état "clientUpdateProhibited" ou "serverUpdateProhibited".

Les valeurs d'état pendingCreate, pendingDelete, pendingRenew, pendingTransfer, et pendingUpdate NE DOIVENT PAS être combinées les unes avec les autres.

Les autres combinaisons d'état non expressément interdites PEUVENT être utilisées.

2.4 Dates et heures

Les valeurs d'attribut de date et heure DOIVENT être représentées en temps universel coordonné (UTC, *Universal Coordinated Time*) en utilisant le calendrier grégorien. La forme étendue de date-heure utilisant les caractères majuscules "T" et "Z" définis dans [xmlschema-2-20041028] DOIT être utilisée pour représenter les valeurs de date-heure, car le schéma XML ne prend pas en charge les formes tronquées de date-heure ou les caractères "T" et "Z" minuscules.

2.5 Périodes de validité

Un objet de nom de domaine PEUT avoir une période de validité spécifiée. Si la politique du serveur accepte les périodes de validité de domaine/objet, la période de validité est définie lorsque un objet de domaine est créé, et elle PEUT être étendue par les commandes EPP <renew> ou <transfer>. S'agissant de politique de serveur, la présente spécification ne définit pas d'actions à effectuer à l'expiration de la période de validité d'un objet de domaine.

Les périodes de validité sont mesurées en années ou mois avec les unités appropriées spécifiées en utilisant l'attribut "unit". Les valeurs valides pour l'attribut "unit" sont "y" pour les années et "m" pour les mois. La valeur de période minimum admise est un (1). La valeur maximum admise est quatre vingt dix neuf (99) en décimal. Un serveur PEUT prendre en charge une valeur maximum inférieure.

2.6 Informations d'autorisation

Les informations d'autorisation sont associées à des objets de domaine pour faciliter les opérations de transfert. Les informations d'autorisation sont allouées lorsque un objet de domaine est créé, et elles peuvent être mises à jour ultérieurement. La présente spécification décrit des informations d'autorisation fondées sur le mot de passe, bien que d'autres mécanismes soient possibles.

2.7 Autres attributs d'enregistrement de ressource du DNS

Bien que le DNS permette que de nombreux types d'enregistrement de ressource soient associés à un domaine, la présente transposition ne spécifie explicitement que les éléments qui décrivent les enregistrements de ressource utilisés pour la délégation et la résolution de domaine. Des facilités pour provisionner d'autres types d'enregistrement de ressource en rapport avec les domaines peuvent être développé en étendant cette transposition.

La méthode de provisionnement décrite dans cette transposition sépare les éléments de données discrets par type de données. Cette méthode de définition de données permet aux processeurs de schéma XML d'effectuer les tâches de base de validation de syntaxe, réduisant l'ambiguïté et la quantité de travail d'analyse et de vérification de syntaxe nécessaire pour les processeurs du protocole. Les méthodes de provisionnement et d'extension qui agrègent les données en chaînes opaques sont possibles, mais de telles méthodes ne devraient pas être utilisées parce qu'elles imposent des exigences supplémentaires d'analyse, d'interprétation, et de validation aux processeurs du protocole.

3. Transposition de commande EPP

Une description détaillée de la syntaxe et de la sémantique d'EPP se trouve dans la [RFC5730]. Les transpositions de commandes décrites ici sont spécifiquement à utiliser pour le provisionnement et la gestion des noms de domaine de l'Internet via EPP.

3.1 Commande d'interrogation EPP

EPP fournit trois commandes pour restituer les informations de domaine : <check> pour déterminer si un objet de domaine peut être provisionné au sein d'un répertoire, <info> pour restituer les informations détaillées associées à un objet de domaine, et <transfer> pour restituer les informations d'état de transfert de domaine/objet.

3.1.1 Commande EPP <check>

La commande EPP <check> est utilisée pour déterminer si un objet peut être provisionné au sein d'un répertoire. Elle fournit

une indication qui permet à un client d'anticiper le succès ou l'échec de l'approvisionnement d'un objet en utilisant la commande <create>, car les exigences d'approvisionnement d'objet sont en fin de compte une affaire de politique de serveur.

En plus des éléments standard de commande EPP, la commande <check> DOIT contenir un élément <domain:check> qui identifie l'espace de noms de domaine. L'élément <domain:check> contient les éléments fils suivants :

- Un ou plusieurs éléments <domain:name> qui contiennent les noms pleinement qualifiés des objets de domaine à interroger.

Exemple de commande <check> :

```
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: <domain:check
C:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:   <domain:name>example.com</domain:name>
C:   <domain:name>example.net</domain:name>
C:   <domain:name>example.org</domain:name>
C: </domain:check>
C: </check>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <check> a été traitée avec succès, l'élément EPP <resData> DOIT contenir un élément <domain:chkData> fils qui identifie l'espace de noms du domaine. L'élément <domain:chkData> contient un ou plusieurs éléments <domain:cd> qui contiennent les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine interrogé. Cet élément DOIT contenir un attribut "avail" dont la valeur indique la disponibilité de l'objet (si il peut ou non être provisionné) au moment où la commande <check> a été achevée. Une valeur de "1" ou "vrai" signifie que l'objet peut être provisionné. Une valeur de "0" ou "faux" signifie que l'objet ne peut pas être provisionné.
- Un élément FACULTATIF <domain:reason> qui PEUT être fourni lorsque un objet ne peut pas être provisionné. Si il est présent, cet élément contient un texte spécifique du serveur pour aider à expliquer pourquoi l'objet ne peut pas être provisionné. Ce texte DOIT être représenté dans le langage de réponse négocié précédemment avec le client ; un attribut FACULTATIF "lang" PEUT être présent pour identifier le langage si la valeur négociée est quelque chose d'autre que la valeur par défaut de "en" (anglais).

Exemple de réponse <check> :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <resData>
S: <domain:chkData
S:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:   <domain:cd>
S:     <domain:name avail="1">example.com</domain:name>
S:   </domain:cd>
S:   <domain:cd>
S:     <domain:name avail="0">example.net</domain:name>
S:   <domain:reason>In use</domain:reason>
S: </domain:cd>
S: <domain:cd>
S:   <domain:name avail="1">example.org</domain:name>
```

```
S: </domain:cd>
S: </domain:chkData>
S: </resData>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54322-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une commande <check> ne peut être traitée pour une raison quelconque.

3.1.2 Commande EPP <info>

La commande EPP <info> est utilisée pour restituer les informations associées à un objet de domaine. La réponse à cette commande PEUT varier selon l'identité du client qui interroge, l'utilisation des informations d'autorisation, et la politique du serveur à l'égard des clients non autorisés. Si le client qui interroge est le client parrain, toutes les informations disponibles DOIVENT être retournées. Si le client qui interroge n'est pas le client parrain mais si le client fournit des informations d'autorisation valides, toutes les informations disponibles DOIVENT être retournées. Si le client qui interroge n'est pas le client parrain et si il ne fournit pas d'informations d'autorisation valides, la politique du serveur détermine quels éléments FACULTATIFS seront retournés.

En plus des éléments de commande EPP, la commande <info> DOIT contenir un élément <domain:info> qui identifie l'espace de noms du domaine. L'élément <domain:info> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine à interroger. Un attribut FACULTATIF "hosts" est disponible pour contrôler le retour des informations qui décrivent les hôtes en rapport avec l'objet de domaine. Une valeur de "all" (par défaut, qui PEUT être absente) retourne les informations qui décrivent les hôtes aussi bien subordonnés que délégués. Une valeur de "del" retourne les informations qui décrivent seulement les hôtes délégués. Une valeur de "sub" retourne les informations qui décrivent seulement les hôtes subordonnés. Une valeur de "none" ne retourne pas d'informations décrivant les hôtes délégués ou subordonnés.
- Un élément FACULTATIF <domain:authInfo> qui contient les informations d'autorisation associées à l'objet de domaine ou les informations d'autorisation associées au contact enregistrant ou aux contacts associés de l'objet de domaine. Un attribut FACULTATIF "roid" DOIT être utilisé pour identifier l'objet enregistrant ou le contact si et seulement si les authInfo données sont associées à un objet enregistrant ou un contact, et non à l'objet de domaine lui-même. Si cet élément n'est pas fourni ou si les informations d'autorisation sont invalides, la politique du serveur détermine si la commande est rejetée ou si des informations de réponse seront retournées au client.

Exemple de commande <info> sans informations d'autorisation :

```
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: <domain:info
C: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C: <domain:name hosts="all">example.com</domain:name>
C: </domain:info>
C: </info>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Exemple de commande <info> avec informations d'autorisation :

```
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: <domain:info
```

```
C:  xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:  <domain:name hosts="all">example.com</domain:name>
C:  <domain:authInfo>
C:  <domain:pw>2fooBAR</domain:pw>
C:  </domain:authInfo>
C:  </domain:info>
C:  </info>
C:  <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <info> a été traitée avec succès, l'élément EPP <resData> DOIT contenir un élément fils <domain:infData> qui identifie l'espace de noms du domaine. Les éléments qui ne sont pas FACULTATIFS DOIVENT être retournés ; les éléments FACULTATIFS sont retournés sur la base de l'autorisation du client et de la politique du serveur. L'élément <domain:infData> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine.
- Un élément <domain:roid> qui contient l'identifiant d'objet de répertoire alloué à l'objet de domaine à la création de l'objet.
- Zéro, un ou plusieurs éléments FACULTATIFS <domain:status> qui contiennent les descripteurs de l'état actuel associés au domaine.
- Si il est accepté par le serveur, un élément FACULTATIF <domain:registrar> et un ou plusieurs éléments FACULTATIFS <domain:contact> qui contiennent des identifiants des objets d'information sur les personnes ou organisations sociales associées à l'objet de domaine.
- Un élément FACULTATIF <domain:ns> qui contient le nom pleinement qualifié des hôtes délégués ou les attributs d'hôtes (serveurs de noms) associé à l'objet de domaine. Voir au paragraphe 1.1 la description des éléments utilisés pour spécifier les objets hôtes ou les attributs d'hôte.
- Zéro, un ou plusieurs éléments FACULTATIFS <domain:host> qui contiennent le nom pleinement qualifié des objets hôtes subordonnés qui existent sous cet objet de domaine supérieur.
- Un élément <domain:clID> qui contient l'identifiant du client parrain.
- Un élément FACULTATIF <domain:crID> qui contient l'identifiant du client qui a créé l'objet de domaine.
- Un élément FACULTATIF <domain:crDate> qui contient la date et heure de création de l'objet de domaine.
- Un élément FACULTATIF <domain:exDate> qui contient la date et heure qui identifie la fin de la période d'enregistrement de l'objet de domaine.
- Un élément FACULTATIF <domain:upID> qui contient l'identifiant du client qui est le dernier à avoir mis à jour l'objet de domaine. Cet élément NE DOIT PAS être présent si le domaine n'a jamais été modifié.
- Un élément FACULTATIF <domain:upDate> qui contient la date et heure de la plus récente modification de domaine/objet. Cet élément NE DOIT PAS être présent si l'objet de domaine n'a jamais été modifié.
- Un élément FACULTATIF <domain:trDate> qui contient la date et heure du plus récent transfert réussi de domaine/objet. Cet élément NE DOIT PAS être fourni si l'objet de domaine n'a jamais été transféré.
- Un élément FACULTATIF <domain:authInfo> qui contient les informations d'autorisation associées à l'objet de domaine. Cet élément DOIT être retourné seulement si le client qui interroge est le client parrain actuel ou si le client a fourni des informations d'autorisation valides avec la commande.

Exemple de réponse <info> pour un client autorisé :

```
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>Commande achevée avec succès</msg>
S:  </result>
S:  <resData>
S:  <domain:infData
S:  xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:  <domain:name>example.com</domain:name>
S:  <domain:roid>EXAMPLE1-REP</domain:roid>
S:  <domain:status s="ok"/>
S:  <domain:registrar>jd1234</domain:registrar>
S:  <domain:contact type="admin">sh8013</domain:contact>
S:  <domain:contact type="tech">sh8013</domain:contact>
S:  <domain:ns>
```



```
S: <domain:hostObj>ns1.example.com</domain:hostObj>
S: <domain:hostObj>ns1.example.net</domain:hostObj>
S: </domain:ns>
S: <domain:host>ns1.example.com</domain:host>
S: <domain:host>ns2.example.com</domain:host>
S: <domain:clID>ClientX</domain:clID>
S: <domain:crID>ClientY</domain:crID>
S: <domain:crDate>1999-04-03T22:00:00.0Z</domain:crDate>
S: <domain:upID>ClientX</domain:upID>
S: <domain:upDate>1999-12-03T09:00:00.0Z</domain:upDate>
S: <domain:exDate>2005-04-03T22:00:00.0Z</domain:exDate>
S: <domain:trDate>2000-04-08T09:00:00.0Z</domain:trDate>
S: <domain:authInfo>
S: <domain:pw>2fooBAR</domain:pw>
S: </domain:authInfo>
S: </domain:infData>
S: </resData>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54322-XYZ</svTRID>
S: </trID>
S: </response>
S: </epp>
```

Un serveur qui a une politique différente de retour d'informations PEUT fournir moins d'informations dans une réponse.

Exemple de réponse <info> pour un client non autorisé :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <resData>
S: <domain:infData
S: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S: <domain:name>example.com</domain:name>
S: <domain:roid>EXAMPLE1-REP</domain:roid>
S: <domain:clID>ClientX</domain:clID>
S: </domain:infData>
S: </resData>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54322-XYZ</svTRID>
S: </trID>
S: </response>
S: </epp>
```

Une réponse d'erreur DOIT être retournée si une commande <info> ne peut être traitée pour une raison quelconque.

3.1.3 Commande d'interrogation EPP <transfer>

La commande EPP <transfer> fournit une opération d'interrogation qui permet à un client de déterminer en temps réel l'état des demandes de transfert en cours et achevées. En plus des éléments standard de commande EPP, la commande <transfer> DOIT contenir un attribut "op" avec la valeur "query", et un élément <domain:transfer> qui identifie l'espace de noms du domaine. L'élément <domain:transfer> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine à interroger.
- Un élément FACULTATIF <domain:authInfo> qui contient les informations d'autorisation associées à l'objet de domaine ou

les informations d'autorisation associées aux contacts d'enregistrement ou associés de l'objet de domaine. Un attribut FACULTATIF "roid" DOIT être utilisé pour identifier l'objet enregistrant ou de contact si et seulement si les authInfo données sont associées à un objet enregistrant ou de contact, et non à l'objet de domaine lui-même. Si cet élément n'est pas fourni ou si les informations d'autorisation sont invalides, la politique du serveur détermine si la commande est rejetée ou si les informations en réponse seront retournées au client.

Exemple de commande d'interrogation <transfer> :

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C: <command>
C: <transfer op="query">
C: <domain:transfer
C:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:   <domain:name>example.com</domain:name>
C:   <domain:authInfo>
C:     <domain:pw roid="JD1234-REP">2fooBAR</domain:pw>
C:   </domain:authInfo>
C: </domain:transfer>
C: </transfer>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande d'interrogation <transfer> a été traitée avec succès, l'élément EPP <resData> DOIT contenir un élément fils <domain:trnData> qui identifie l'espace de noms du domaine. L'élément <domain:trnData> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine.
- Un élément <domain:trStatus> qui contient l'état de la plus récente demande de transfert.
- Un élément <domain:reID> qui contient l'identifiant du client qui a demandé le transfert de l'objet.
- Un élément <domain:reDate> qui contient la date et l'heure de demande du transfert.
- Un élément <domain:acID> qui contient l'identifiant du client qui DEVRAIT agir sur une demande de transfert EN COURS. Pour tous les autres types d'état, la valeur identifie le client qui a fait l'action indiquée.
- Un élément <domain:acDate> qui contient la date et l'heure d'une réponse exigée ou achevée. Pour une demande EN COURS, la valeur identifie la date et l'heure à laquelle une réponse est exigée avant qu'une action de réponse automatique soit faite par le serveur. Pour tous les autres types d'état, la valeur identifie la date et l'heure à laquelle la demande a été achevée.
- Un élément FACULTATIF <domain:exDate> qui contient la fin de la période de validité de l'objet de domaine si la commande <transfer> a causé ou cause un changement de période de validité.

Exemple de réponse d'interrogation <transfer> :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <resData>
S: <domain:trnData
S:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:   <domain:name>example.com</domain:name>
S:   <domain:trStatus>pending</domain:trStatus>
S:   <domain:reID>ClientX</domain:reID>
S:   <domain:reDate>2000-06-06T22:00:00.0Z</domain:reDate>
S:   <domain:acID>ClientY</domain:acID>
S:   <domain:acDate>2000-06-11T22:00:00.0Z</domain:acDate>
S:   <domain:exDate>2002-09-08T22:00:00.0Z</domain:exDate>
S: </domain:trnData>
S: </resData>
```

```
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54322-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une interrogation <transfer> ne peut être traitée pour une raison quelconque.

3.2 Commandes de transformation EPP

EPP fournit cinq commandes pour transformer les objets de domaine : <create> pour créer une instance d'objet de domaine, <delete> pour supprimer une instance d'objet de domaine, <renew> pour étendre la période de validité d'un objet de domaine, <transfer> pour gérer les changements de parrainage d'un objet de domaine, et <update> pour changer les informations associées à un objet de domaine.

Les commandes de transformation sont normalement traitées et achevées en temps réel. Les opérateurs de serveurs PEUVENT recevoir et traiter les commandes de transformation mais différer d'achever l'action demandée si une révision humaine ou de tiers est nécessaire avant que l'action demandée puisse être achevée. Dans de telles situations le serveur DOIT retourner un code de réponse 1001 au client pour noter que la commande a été reçue et traitée mais que l'action demandée est en cours. Le serveur DOIT aussi gérer l'état de l'objet qui fait l'objet de la commande pour refléter l'initiation et l'achèvement de l'action demandée. Une fois l'action achevée, tous les clients impliqués dans la transaction DOIVENT être notifiés en utilisant un message de service de ce que l'action a été achevée et de ce que l'état de l'objet a changé. D'autres méthodes de notification PEUVENT être utilisées en plus du message de service requis.

Les opérateurs de serveurs DEVRAIENT confirmer qu'un client est autorisé à effectuer une commande de transformation sur un objet donné. Toute tentative de transformer un objet par un client non autorisé DOIT être rejetée, et le serveur DOIT retourner un code de réponse 2201 au client pour noter que le client n'a pas les privilèges pour exécuter la commande demandée.

3.2.1 Commande EPP <create>

La commande EPP <create> fournit une opération de transformation qui permet à un client de créer un objet de domaine. En plus des éléments de commande EPP standard, la commande <create> DOIT contenir un élément <domain:create> qui identifie l'espace de noms du domaine. L'élément <domain:create> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine à créer.
- Un élément FACULTATIF <domain:period> qui contient la période initiale d'enregistrement de l'objet de domaine. Un serveur PEUT définir une période d'enregistrement initial par défaut si elle n'est pas spécifiée par le client.
- Un élément FACULTATIF <domain:ns> qui contient les noms pleinement qualifiés des objets d'hôte délégués ou les attributs d'hôte (serveurs de noms) associés à l'objet de domaine pour fournir des services de résolution pour le domaine ; voir au paragraphe 1.1 la description des éléments utilisés pour spécifier les objets hôtes ou les attributs d'hôtes. Un objet hôte DOIT être connu du serveur avant que l'objet hôte puisse être associé à un objet de domaine.
- Un élément FACULTATIF <domain:registrant> ,qui contient l'identifiant de l'objet d'informations sociales humaines ou organisationnelles (contact) à associer à l'objet de domaine comme enregistrant l'objet. Cet identifiant d'objet DOIT être connu du serveur avant que l'objet de contact puisse être associé à l'objet de domaine. La transposition de commande EPP pour les objets de contact est décrite dans la [RFC5733].
- Zéro, un ou plusieurs éléments FACULTATIFS <domain:contact> qui contiennent les identifiants des autres objets de contact à associer à l'objet de domaine. Les identifiants d'objet de contact DOIVENT être connus du serveur avant que l'objet de contact puisse être associé à l'objet de domaine.
- Un élément <domain:authInfo> qui contient les informations d'autorisation à associer à l'objet de domaine. Cette transposition inclut un mécanisme d'authentification fondé sur le mot de passe, mais le schéma permet que de nouveaux mécanismes soient définis dans de nouveaux schémas.

Exemple de commande <create> :

```
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: <domain:create
C:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:   <domain:name>example.com</domain:name>
C:   <domain:period unit="y">2</domain:period>
C:   <domain:ns>
C:     <domain:hostObj>ns1.example.net</domain:hostObj>
C:     <domain:hostObj>ns2.example.net</domain:hostObj>
C:   </domain:ns>
C:   <domain:registrant>jd1234</domain:registrant>
C:   <domain:contact type="admin">sh8013</domain:contact>
C:   <domain:contact type="tech">sh8013</domain:contact>
C:   <domain:authInfo>
C:     <domain:pw>2fooBAR</domain:pw>
C:   </domain:authInfo>
C: </domain:create>
C: </create>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>

```

Lorsque une commande `<create>` a été traitée avec succès, l'élément EPP `<resData>` DOIT contenir un élément fils `<domain:creData>` qui identifie l'espace de noms du domaine. L'élément `<domain:creData>` contient les éléments fils suivants :

- Un élément `<domain:name>` qui contient le nom pleinement qualifié de l'objet de domaine.
- Un élément `<domain:crDate>` qui contient la date et heure de la création de l'objet de domaine.
- Un élément FACULTATIF `<domain:exDate>` contenant la date et heure qui identifie la fin de la période d'enregistrement de l'objet de domaine.

Exemple de réponse `<create>` :

```

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>Commande achevée avec succès</msg>
S:   </result>
S:   <resData>
S:     <domain:creData
S:       xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:       <domain:name>example.com</domain:name>
S:       <domain:crDate>1999-04-03T22:00:00.0Z</domain:crDate>
S:       <domain:exDate>2001-04-03T22:00:00.0Z</domain:exDate>
S:     </domain:creData>
S:   </resData>
S:   <trID>
S:     <clTRID>ABC-12345</clTRID>
S:     <svTRID>54321-XYZ</svTRID>
S:   </trID>
S: </response>
S:</epp>

```

Une réponse d'erreur DOIT être retournée si une commande `<create>` ne peut être traitée pour une raison quelconque.

3.2.2 Commande EPP `<delete>`

La commande EPP `<delete>` fournit une opération de transformation qui permet à un client de supprimer un objet de domaine. En plus des éléments de commande EPP standard, la commande `<delete>` DOIT contenir un élément `<domain:delete>` qui identifie l'espace de noms du domaine. L'élément `<domain:delete>` contient les éléments fils suivants :

- Un élément `<domain:name>` qui contient le nom pleinement qualifié de l'objet de domaine à supprimer.

Un objet de domaine NE DEVRAIT PAS être supprimé si des objets hôtes subordonnés sont associés à l'objet de domaine. Par exemple, si le domaine "example.com" existe et si l'objet hôte "ns1.example.com" existe aussi, alors le domaine "example.com" NE DEVRAIT PAS être supprimé tant que l'hôte "ns1.example.com" n'a pas été soit supprimé, soit renommé pour exister dans un domaine supérieur différent. Un serveur DEVRAIT notifier aux clients que des relations d'objets existent en envoyant un code de réponse d'erreur 2305 lorsque une commande <delete> est tentée et échoue à cause de l'existence de relations d'objets. Les objets hôtes délégués et subordonnés associés à un objet de domaine peuvent être déterminés en utilisant la commande <info> pour l'objet de domaine.

Exemple de commande <delete> :

```
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: <domain:delete
C:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:   <domain:name>example.com</domain:name>
C: </domain:delete>
C: </delete>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <delete> a été traité avec succès, un serveur DOIT répondre par une réponse EPP sans élément <resData>.

Exemple de réponse <delete> :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54321-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une commande <delete> ne peut être traitée pour une raison quelconque.

3.2.3 Commande EPP <renew>

La commande EPP <renew> fournit une opération de transformation qui permet à un client d'étendre la période de validité d'un objet de domaine. En plus des éléments de commande EPP standard, la commande <renew> DOIT contenir un élément <domain:renew> qui identifie l'espace de noms du domaine. L'élément <domain:renew> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine dont la période de validité est à étendre.
- Un élément <domain:curExpDate> qui contient la date à laquelle se termine la période de validité en cours. Cette valeur assure que des commandes <renew> répétées ne résultent pas en plusieurs renouvellements réussis imprévus.
- Un élément FACULTATIF <domain:period> qui contient le nombre d'unités à ajouter à la période d'enregistrement de l'objet de domaine. Le nombre d'unités disponible PEUT être soumis à des limites imposées par le serveur.

Exemple de commande <renew> :

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
```

```
C: <command>
C: <renew>
C: <domain:renew
C:   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:   <domain:name>example.com</domain:name>
C:   <domain:curExpDate>2000-04-03</domain:curExpDate>
C:   <domain:period unit="y">5</domain:period>
C: </domain:renew>
C: </renew>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <renew> a été traitée avec succès, l'élément EPP <resData> DOIT contenir un élément fils <domain:renData> qui identifie l'espace de noms du domaine. L'élément <domain:renData> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine.
- Un élément FACULTATIF <domain:exDate> qui contient la date et l'heure identifiant la fin de la période d'enregistrement de l'objet de domaine.

Exemple de réponse <renew> :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <resData>
S:   <domain:renData
S:     xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:     <domain:name>example.com</domain:name>
S:     <domain:exDate>2005-04-03T22:00:00.0Z</domain:exDate>
S:   </domain:renData>
S: </resData>
S: <trID>
S:   <clTRID>ABC-12345</clTRID>
S:   <svTRID>54322-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une commande <renew> ne peut être traitée pour une raison quelconque.

3.2.4 Commande EPP <transfer>

La commande EPP <transfer> fournit une opération de transformation qui permet à un client de gérer les demandes de transfert du parrainage d'un objet de domaine. En plus des éléments de commande EPP standard, la commande <transfer> DOIT contenir un élément <domain:transfer> qui identifie l'espace de noms du domaine. L'élément <domain:transfer> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine pour lequel une demande de transfert est à créer, approuver, rejeter, ou annuler.
- Un élément FACULTATIF <domain:period> qui contient le nombre d'unités à ajouter à la période d'enregistrement de l'objet de domaine à l'achèvement du processus de transfert. Cet élément ne peut être utilisé que lorsque un transfert est demandé, et il DOIT être ignoré autrement. Le nombre d'unités disponible PEUT être soumis à des limites imposées par le serveur.
- Un élément <domain:authInfo> qui contient les informations d'autorisation associées à l'objet de domaine ou les informations d'autorisation associées au registrant ou aux contacts associés de l'objet de domaine. Un attribut FACULTATIF "roid" DOIT être utilisé pour identifier l'objet enregistrant ou de contact si et seulement si les authInfo

données sont associées à un objet enregistrant ou de contact, et non à l'objet de domaine lui-même.

Chaque commande EPP <transfer> DOIT contenir un attribut "op" qui identifie l'opération de transfert à effectuer. Les valeurs, définitions, et autorisations valides pour toutes les valeurs d'attribut sont définies dans la [RFC5730].

Le transfert d'un objet de domaine DOIT implicitement transférer tous les objets hôtes qui sont subordonnés à l'objet de domaine. Par exemple, si l'objet de domaine "example.com" est transféré et si l'objet hôte "ns1.example.com" existe, l'objet hôte DOIT être transféré au titre du processus de transfert de "example.com". Les objets hôtes qui font l'objet d'un transfert lors d'un transfert d'objet de domaine sont énumérés dans la réponse à une commande EPP <info> effectuée sur l'objet de domaine.

Exemple de commande de demande <transfer> :

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C: <command>
C:   <transfer op="request">
C:     <domain:transfer
C:       xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:       <domain:name>example.com</domain:name>
C:       <domain:period unit="y">1</domain:period>
C:       <domain:authInfo>
C:         <domain:pw roid="JD1234-REP">2fooBAR</domain:pw>
C:       </domain:authInfo>
C:     </domain:transfer>
C:   </transfer>
C:   <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <transfer> a été traitée avec succès, l'élément EPP <resData> DOIT contenir un élément fils <domain:trnData> qui identifie l'espace de noms du domaine. L'élément <domain:trnData> contient les mêmes éléments fils que définis pour une réponse d'interrogation de transfert.

Exemple de réponse <transfer> :

```
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="1001">
S:     <msg>Commande achevée avec succès ; action en cours</msg>
S:   </result>
S:   <resData>
S:     <domain:trnData
S:       xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:       <domain:name>example.com</domain:name>
S:       <domain:trStatus>pending</domain:trStatus>
S:       <domain:reID>ClientX</domain:reID>
S:       <domain:reDate>2000-06-08T22:00:00.0Z</domain:reDate>
S:       <domain:acID>ClientY</domain:acID>
S:       <domain:acDate>2000-06-13T22:00:00.0Z</domain:acDate>
S:       <domain:exDate>2002-09-08T22:00:00.0Z</domain:exDate>
S:     </domain:trnData>
S:   </resData>
S:   <trID>
S:     <clTRID>ABC-12345</clTRID>
S:     <svTRID>54322-XYZ</svTRID>
S:   </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une commande <transfer> ne peut pas être traitée pour une raison quelconque.

3.2.5 Commande EPP <update>

La commande EPP <update> fournit une opération de transformation qui permet à un client de modifier les attributs d'un objet de domaine. En plus des éléments de commande EPP standard, la commande <update> DOIT contenir un élément <domain:update> qui identifie l'espace de noms du domaine. L'élément <domain:update> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine à mettre à jour.
- Un élément FACULTATIF <domain:add> qui contient les valeurs d'attribut à ajouter à l'objet.
- Un élément FACULTATIF <domain:rem> qui contient les valeurs d'attribut à retirer à l'objet.
- Un élément FACULTATIF <domain:chg> qui contient les valeurs d'attribut d'objet à changer.

Au moins un élément <domain:add>, <domain:rem>, ou <domain:chg> DOIT être fourni si la commande n'est pas étendue. Tous ces éléments PEUVENT être omis si une extension <update> est présente. Les éléments <domain:add> et <domain:rem> contiennent les éléments fils suivants :

- Un élément FACULTATIF <domain:ns> qui contient les noms pleinement qualifiés des objets hôtes délégués ou attributs d'hôte (serveurs de noms) associés à l'objet de domaine pour fournir des services de résolution pour le domaine ; voir au paragraphe 1.1 la description des éléments utilisés pour spécifier les objets hôtes ou les attributs d'hôte. Un objet hôte DOIT être connu du serveur avant que l'objet hôte puisse être associé à un objet de domaine. Si des attributs d'hôte sont utilisés pour spécifier des serveurs de noms, on notera que les éléments d'adresse IP ne sont pas nécessaires pour identifier un serveur de noms qui est retiré. Les éléments d'adresse IP peuvent être absents ou ignorés en toute sécurité dans cette situation.
- Zéro, un ou plusieurs éléments <domain:contact> qui contiennent les identifiants des objets de contact à associer à l'objet de domaine ou à en retirer. Les identifiants d'objet de contact DOIVENT être connus du serveur avant que l'objet de contact puisse être associé à l'objet de domaine.
- Zéro, un ou plusieurs éléments <domain:status> qui contiennent les valeurs d'état à appliquer ou à retirer à l'objet. Lorsque on spécifie une valeur à retirer, seule la valeur de l'attribut est significative ; le texte de l'élément n'est pas obligé de correspondre à une valeur à retirer.

Un élément <domain:chg> contient les éléments fils suivants :

- Un élément <domain:registrant> qui contient l'identifiant de l'objet d'information (contact) humain ou d'organisation sociale à associer à l'objet de domaine comme enregistreur de l'objet. Cet identifiant d'objet DOIT être connu du serveur avant que l'objet de contact puisse être associé à l'objet de domaine. Un élément vide peut être utilisé pour supprimer les informations d'enregistreur.
- Un élément <domain:authInfo> qui contient les informations d'autorisation associées à l'objet de domaine. Cette transposition inclut un mécanisme d'authentification fondé sur le mot de passe, mais le schéma permet que de nouveaux mécanismes soient définis dans de nouveaux schémas. Un élément <domain:null> peut être utilisé au sein de l'élément <domain:authInfo> pour supprimer des informations d'autorisation.

Exemple de commande <update> :

```
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: <domain:update
C: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C: <domain:name>example.com</domain:name>
C: <domain:add>
C: <domain:ns>
C: <domain:hostObj>ns2.example.com</domain:hostObj>
C: </domain:ns>
C: <domain:contact type="tech">mak21</domain:contact>
C: <domain:status s="clientHold"
C: lang="fr">Trop perçu.</domain:status>
C: </domain:add>
C: <domain:rem>
C: <domain:ns>
C: <domain:hostObj>ns1.example.com</domain:hostObj>
```



```
C: </domain:ns>
C: <domain:contact type="tech">sh8013</domain:contact>
C: <domain:status s="clientUpdateProhibited"/>
C: </domain:rem>
C: <domain:chg>
C: <domain:registrant>sh8013</domain:registrant>
C: <domain:authInfo>
C: <domain:pw>2BARfoo</domain:pw>
C: </domain:authInfo>
C: </domain:chg>
C: </domain:update>
C: </update>
C: <clTRID>ABC-12345</clTRID>
C: </command>
C:</epp>
```

Lorsque une commande <update> a été traitée avec succès, un serveur DOIT répondre par une réponse EPP sans élément <resData>.

Exemple de réponse <update> :

```
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>Commande achevée avec succès</msg>
S: </result>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54321-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

Une réponse d'erreur DOIT être retournée si une commande <update> ne peut être traitée pour une raison quelconque.

3.3 Révision hors ligne d'actions demandées

Les commandes sont traitées par un serveur dans l'ordre de leur réception du client. Bien qu'une réponse immédiate confirmant la réception et le traitement de la commande soit produite par le serveur, un opérateur de serveur PEUT effectuer une révision hors ligne des commandes de transformation demandées avant d'achever l'action demandée. Dans de telles situations, la réponse du serveur DOIT clairement noter que la commande de transformation a été reçue et traitée mais que l'action demandée est en cours. L'état de l'objet correspondant DOIT clairement refléter le traitement de l'action en cours. Le serveur DOIT notifier au client quand le traitement hors ligne de l'action est achevé.

On inclut ici des exemples qui décrivent une commande <create> qui exige une révision hors ligne. Noter le code de résultat et le message retournés en réponse à la commande <create>.

```
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="1001">
S: <msg>Commande achevée avec succès ; action en cours</msg>
S: </result>
S: <resData>
S: <domain:creData
S: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S: <domain:name>example.com</domain:name>
S: <domain:crDate>1999-04-03T22:00:00.0Z</domain:crDate>
```

```
S: <domain:exDate>2001-04-03T22:00:00.0Z</domain:exDate>
S: </domain:creData>
S: </resData>
S: <trID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54321-XYZ</svTRID>
S: </trID>
S: </response>
S:</epp>
```

L'état de l'objet de domaine après le retour de cette réponse DOIT inclure "pendingCreate". L'opérateur de serveur reçoit la demande hors ligne, et informe le client du résultat de la revue soit en mettant en file d'attente un message de service pour restitution via la commande <poll>, soit en utilisant un mécanisme hors bande pour informer le client de la demande.

Le message de service DOIT contenir un texte qui décrit la notification dans l'élément fils <msg> de l'élément <msgQ> de la réponse. De plus, l'élément EPP <resData> DOIT contenir un élément fils <domain:panData> qui identifie l'espace de noms du domaine. L'élément <domain:panData> contient les éléments fils suivants :

- Un élément <domain:name> qui contient le nom pleinement qualifié de l'objet de domaine. L'élément <domain:name> contient un attribut EXIGÉ "paResult". Une valeur booléenne positive indique que la demande a été approuvée et achevée. Une valeur booléenne négative indique que la demande a été refusée et que l'action demandée n'a pas été effectuée.
- Un élément <domain:paTRID> qui contient l'identifiant de transaction du client et l'identifiant de transaction du serveur retournés avec l'original de la réponse au traitement de la commande. L'identifiant de transaction du client est FACULTATIF et ne sera retourné que si le client a fourni un identifiant dans la commande <create> d'origine.
- Un élément <domain:paDate> qui contient la date et l'heure de l'achèvement de l'action demandée.

Exemple de message de service "revue achevée" :

```
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="1301">
S: <msg>Commande achevée avec succès ; accuser réception pour sortir de la file d'attente</msg>
S: </result>
S: <msgQ count="5" id="12345">
S: <qDate>1999-04-04T22:01:00.0Z</qDate>
S: <msg>Action en cours achevée avec succès.</msg>
S: </msgQ>
S: <resData>
S: <domain:panData
S: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S: <domain:name paResult="1">example.com</domain:name>
S: <domain:paTRID>
S: <clTRID>ABC-12345</clTRID>
S: <svTRID>54321-XYZ</svTRID>
S: </domain:paTRID>
S: <domain:paDate>1999-04-04T22:00:00.0Z</domain:paDate>
S: </domain:panData>
S: </resData>
S: <trID>
S: <clTRID>BCD-23456</clTRID>
S: <svTRID>65432-WXY</svTRID>
S: </trID>
S: </response>
S:</epp>
```

4. Syntaxe formelle

Une transposition d'objet EPP est spécifiée en notation de schéma XML. La syntaxe formelle présentée ici est une représentation de la transposition d'objet qui convient pour les instances de validation automatique de XML EPP. Les étiquettes

DÉBUT et FIN ne font pas partie du schéma ; elles sont utilisées pour noter le début et la fin du schéma pour les besoins d'enregistrement d'URI.

Copyright (c) 2009 IETF Trust et les personnes identifiées comme auteurs du code. Tous droits réservés.

La redistribution et l'utilisation en forme source et binaire, avec ou sans modification, sont permises sous réserve du respect des conditions suivantes :

- o Les redistributions de code source doivent conserver la notice de droits de reproduction, cette liste de conditions et le déclinatoire de responsabilité qui suit.
- o Les redistributions en forme binaire doivent reproduire la notice de droits de reproduction ci-dessus, cette liste de conditions et le déclinatoire de responsabilité qui suit dans la documentation et/ou autres matériaux fournis avec la distribution.
- o Ni le nom de la Internet Society, de l'IETF ou de l'IETF Trust, ni les noms des contributeurs, ne peuvent être utilisés pour endosser ou promouvoir des produits dérivés du présent logiciel sans permission préalable écrite spécifique.

Le présent logiciel est fourni "tel quel" par les détenteurs des droits de reproduction et les contributeurs et toutes garanties expresses ou implicites, y compris, mais sans s'y limiter, les garanties implicites de commercialisation et de convenance pour un objet particulier sont déclinées. En aucun cas le détenteur des droits de reproduction ou les contributeurs ne pourront cependant être tenus pour responsables d'aucun dommage direct, indirect, incident, spécial, exemplaire, ou conséquent (incluant, sans s'y limiter, la fourniture de biens ou services, la perte d'usage, de données, ou de profits, ou une interruption d'affaires) causés et selon aucune théorie de responsabilité, que ce soit par contrat, responsabilité stricte, ou tort (incluant la négligence ou autrement) résultant de quelque façon que ce soit de l'utilisation du présent logiciel, même si il est informé de la possibilité de tels dommages.

DÉBUT

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<schema targetNamespace="urn:ietf:params:xml:ns:domain-1.0"
  xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
  xmlns:host="urn:ietf:params:xml:ns:host-1.0"
  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">
```

```
<!-- Importation des types d'éléments communs. -->
```

```
<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:host-1.0"/>
```

```
<annotation>
```

```
<documentation>
```

```
Schéma d'approvisionnement de domaine du protocole d'approvisionnement extensible v1.0
```

```
</documentation>
```

```
</annotation>
```

```
<!-- Éléments fils trouvés dans les commandes EPP. -->
```

```
<element name="check" type="domain:mNameType"/>
```

```
<element name="create" type="domain:createType"/>
```

```
<element name="delete" type="domain:sNameType"/>
```

```
<element name="info" type="domain:infoType"/>
```

```
<element name="renew" type="domain:renewType"/>
```

```
<element name="transfer" type="domain:transferType"/>
```

```
<element name="update" type="domain:updateType"/>
```

```
<!-- Éléments fils de la commande <create>. -->
```

```
<complexType name="createType">
```

```
<sequence>
```

```
<element name="name" type="eppcom:labelType"/>
```

```
<element name="period" type="domain:periodType"
```

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

```
<complexType name="periodType">
  <simpleContent>
    <extension base="domain:pLimitType">
      <attribute name="unit" type="domain:pUnitType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>
```

```
<simpleType name="pLimitType">
  <restriction base="unsignedShort">
    <minInclusive value="1"/>
    <maxInclusive value="99"/>
  </restriction>
</simpleType>
```

```
<simpleType name="pUnitType">
  <restriction base="token">
    <enumeration value="y"/>
    <enumeration value="m"/>
  </restriction>
</simpleType>
```

```
<complexType name="nsType">
  <choice>
    <element name="hostObj" type="eppcom:labelType"
      maxOccurs="unbounded"/>
    <element name="hostAttr" type="domain:hostAttrType"
      maxOccurs="unbounded"/>
  </choice>
</complexType>
```

<!-- Les noms de serveurs sont soit des objets, soit des attributs d'hôtes. -->

```
<complexType name="hostAttrType">
  <sequence>
    <element name="hostName" type="eppcom:labelType"/>
    <element name="hostAddr" type="host:addrType"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>
```

<!-- Pour des attributs, les adresses sont FACULTATIVES et suivent la structure définie dans la transposition d'hôte. -->

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

```
</simpleContent>
</complexType>

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

<complexType name="authInfoType">
  <choice>
    <element name="pw" type="eppcom:pwAuthInfoType"/>
    <element name="ext" type="eppcom:extAuthInfoType"/>
  </choice>
</complexType>

<!-- Éléments fils des commandes qui exigent un seul nom. -->
<complexType name="sNameType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
  </sequence>
</complexType>

<!-- Éléments fils des commandes qui acceptent plusieurs noms. -->
<complexType name="mNameType">
  <sequence>
    <element name="name" type="eppcom:labelType"
      maxOccurs="unbounded"/>
  </sequence>
</complexType>

<!-- Éléments fils de la commande <info>. -->
<complexType name="infoType">
  <sequence>
    <element name="name" type="domain:infoNameType"/>
    <element name="authInfo" type="domain:authInfoType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="infoNameType">
  <simpleContent>
    <extension base="eppcom:labelType">
      <attribute name="hosts" type="domain:hostsType"
        default="all"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="hostsType">
  <restriction base="token">
    <enumeration value="all"/>
    <enumeration value="del"/>
    <enumeration value="none"/>
    <enumeration value="sub"/>
  </restriction>
</simpleType>
```

```
<!-- Éléments fils de la commande <renew>. -->
<complexType name="renewType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="curExpDate" type="date"/>
    <element name="period" type="domain:periodType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Éléments fils de la commande <transfer>. -->
<complexType name="transferType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="period" type="domain:periodType"
      minOccurs="0"/>
    <element name="authInfo" type="domain:authInfoType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Éléments fils de la commande <update>. -->
<complexType name="updateType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="add" type="domain:addRemType"
      minOccurs="0"/>
    <element name="rem" type="domain:addRemType"
      minOccurs="0"/>
    <element name="chg" type="domain:chgType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Éléments de données qui peuvent être ajoutés ou supprimés. -->
<complexType name="addRemType">
  <sequence>
    <element name="ns" type="domain:nsType"
      minOccurs="0"/>
    <element name="contact" type="domain:contactType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="status" type="domain:statusType"
      minOccurs="0" maxOccurs="11"/>
  </sequence>
</complexType>

<!-- Éléments de données qui peuvent être changés. -->
<complexType name="chgType">
  <sequence>
    <element name="registrant" type="domain:clIDChgType"
      minOccurs="0"/>
    <element name="authInfo" type="domain:authInfoChgType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Permet à la valeur d'enregistreur d'être nullifiée en changeant la restriction minLength à "0". -->
<simpleType name="clIDChgType">
  <restriction base="token">
    <minLength value="0"/>
  </restriction>
</simpleType>
```

```
<maxLength value="16"/>
</restriction>
</simpleType>

<!-- Permet à la valeur de authInfo d'être nullifiée en incluant un élément vide dans le choix. -->
<complexType name="authInfoChgType">
  <choice>
    <element name="pw" type="eppcom:pwAuthInfoType"/>
    <element name="ext" type="eppcom:extAuthInfoType"/>
    <element name="null"/>
  </choice>
</complexType>

<!-- Éléments fils de réponse. -->
<element name="chkData" type="domain:chkDataType"/>
<element name="creData" type="domain:creDataType"/>
<element name="infData" type="domain:infDataType"/>
<element name="panData" type="domain:panDataType"/>
<element name="renData" type="domain:renDataType"/>
<element name="trnData" type="domain:trnDataType"/>

<!-- Éléments de réponse <check>. -->
<complexType name="chkDataType">
  <sequence>
    <element name="cd" type="domain:checkType"
      maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="checkType">
  <sequence>
    <element name="name" type="domain:checkNameType"/>
    <element name="reason" type="eppcom:reasonType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="checkNameType">
  <simpleContent>
    <extension base="eppcom:labelType">
      <attribute name="avail" type="boolean"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

<!-- Éléments de réponse <create>. -->
<complexType name="creDataType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="crDate" type="dateTime"/>
    <element name="exDate" type="dateTime"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Éléments de réponse <info>. -->
<complexType name="infDataType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
```

```
<element name="roid" type="eppcom:roidType"/>
<element name="status" type="domain:statusType"
  minOccurs="0" maxOccurs="11"/>
<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"/>
<element name="host" type="eppcom:labelType"
  minOccurs="0" maxOccurs="unbounded"/>
<element name="clID" type="eppcom:clIDType"/>
<element name="crID" type="eppcom:clIDType"
  minOccurs="0"/>
<element name="crDate" type="dateTime"
  minOccurs="0"/>
<element name="upID" type="eppcom:clIDType"
  minOccurs="0"/>
<element name="upDate" type="dateTime"
  minOccurs="0"/>
<element name="exDate" type="dateTime"
  minOccurs="0"/>
<element name="trDate" type="dateTime"
  minOccurs="0"/>
<element name="authInfo" type="domain:authInfoType"
  minOccurs="0"/>
</sequence>
</complexType>
```

<!-- L'état est une combinaison d'attributs et d'un message FACULTATIF lisible par l'homme qui peut être exprimé dans un langage autre que l'anglais. -->

```
<complexType name="statusType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="s" type="domain:statusValueType"
        use="required"/>
      <attribute name="lang" type="language"
        default="en"/>
    </extension>
  </simpleContent>
</complexType>
```

```
<simpleType name="statusValueType">
  <restriction base="token">
    <enumeration value="clientDeleteProhibited"/>
    <enumeration value="clientHold"/>
    <enumeration value="clientRenewProhibited"/>
    <enumeration value="clientTransferProhibited"/>
    <enumeration value="clientUpdateProhibited"/>
    <enumeration value="inactive"/>
    <enumeration value="ok"/>
    <enumeration value="pendingCreate"/>
    <enumeration value="pendingDelete"/>
    <enumeration value="pendingRenew"/>
    <enumeration value="pendingTransfer"/>
    <enumeration value="pendingUpdate"/>
    <enumeration value="serverDeleteProhibited"/>
    <enumeration value="serverHold"/>
    <enumeration value="serverRenewProhibited"/>
    <enumeration value="serverTransferProhibited"/>
  </restriction>
</simpleType>
```



```
<enumeration value="serverUpdateProhibited"/>
</restriction>
</simpleType>

<!-- Éléments de réponse de notification d'action en cours. -->
<complexType name="panDataType">
  <sequence>
    <element name="name" type="domain:paNameType"/>
    <element name="paTRID" type="epp:trIDType"/>
    <element name="paDate" type="dateTime"/>
  </sequence>
</complexType>

<complexType name="paNameType">
  <simpleContent>
    <extension base="eppcom:labelType">
      <attribute name="paResult" type="boolean"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

<!-- Éléments de réponse <renew>. -->
<complexType name="renDataType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="exDate" type="dateTime"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Éléments de réponse <transfer>. -->
<complexType name="trnDataType">
  <sequence>
    <element name="name" type="eppcom:labelType"/>
    <element name="trStatus" type="eppcom:trStatusType"/>
    <element name="reID" type="eppcom:clIDType"/>
    <element name="reDate" type="dateTime"/>
    <element name="acID" type="eppcom:clIDType"/>
    <element name="acDate" type="dateTime"/>
    <element name="exDate" type="dateTime"
      minOccurs="0"/>
  </sequence>
</complexType>

<!-- Fin de schéma. -->
</schema>
FIN
```

5. Considérations d'internationalisation

EPP est représenté en XML, qui fournit la prise en charge native des informations de codage en utilisant le jeu de caractères Unicode et ses représentations plus compactes incluant UTF-8. Les processeurs conformes à XML reconnaissent UTF-8 et UTF-16. Bien que XML comporte des dispositions pour identifier et utiliser d'autres codages de caractères par l'utilisation d'un attribut "encoding" dans une déclaration <?xml?>, l'utilisation de UTF-8 est RECOMMANDÉE dans les environnements où existent des incompatibilités de prise en charge d'analyseur de codage.

Toutes les valeurs de date et heure présentées via EPP DOIVENT être exprimées en temps universel coordonné en utilisant le

calendrier grégorien. Le schéma XML permet l'utilisation d'identifiants de zone horaire pour indiquer les décalages au méridien zéro, mais cette option NE DOIT PAS être utilisée avec EPP. La forme étendue de date-heure utilisant les caractères majuscules "T" et "Z" définie dans [xmlschema-2-20041028] DOIT être utilisée pour représenter les valeurs de date-heure, car le schéma XML ne prend pas en charge les formes tronquées de date-heure ou les caractères "T" et "Z" en minuscules.

Le présent document exige la syntaxe de domaine et de nom d'hôte spécifiée dans la [RFC0952] mise à jour par la [RFC1123]. Au moment de la rédaction du présent document, la [RFC3490] décrit une utilisation standard de certaines étiquettes de noms ASCII pour représenter des étiquettes de noms non ASCII. Ces exigences de conformité pourraient changer par suite des travaux en cours pour développer des normes pour les noms de domaines internationalisés.

6. Considérations relatives à l'IANA

Le présent document utilise des URN pour décrire les espaces de noms XML et les schémas XML conformes à un mécanisme de registre décrit dans la [RFC3688]. Deux allocations d'URI ont été enregistrées par l'IANA.

Demande d'enregistrement pour l'espace de nom de domaine :

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

Contact : voir la section "Adresse de l'auteur" du présent document.

XML : aucun. Les URI d'espace de noms ne représentent pas une spécification XML.

Demande d'enregistrement pour le schéma XML de domaine :

URI : urn:ietf:params:xml:schema:domain-1.0

Contact : voir la section "Adresse de l'auteur" du présent document.

XML : voir la section "Syntaxe formelle" du présent document.

7. Considérations pour la sécurité

Les informations d'autorisation décrites au paragraphe 2.6 sont EXIGÉES pour créer un objet de domaine. Ces informations sont utilisées dans certaines opérations d'interrogation et de transfert comme des moyens supplémentaires pour déterminer l'autorisation du client d'effectuer la commande. L'échec à protéger les informations d'autorisation contre une divulgation accidentelle peut résulter en des opérations de transfert non autorisées et à des divulgations non autorisées d'informations. Le client et le serveur DOIVENT tous deux s'assurer que les informations d'autorisation sont mémorisées et échangées avec des mécanismes de chiffrement de qualité pour assurer les services de confidentialité.

La transposition d'objet décrite dans le présent document ne fournit aucun autre service de sécurité ni n'introduit aucune considérations supplémentaires à celles décrites dans la [RFC5730] ou celles causées par les couches de protocole utilisées par EPP.

8. Remerciements

La RFC 3731 a été produite par le groupe de travail PROVREG, qui a suggéré des améliorations et fourni de nombreux commentaires précieux. L'auteur souhaite remercier de leurs efforts les présidents du groupe de travail Edward Lewis et Jaap Akkerhuis pour leurs contributions au traitement et à la rédaction. La RFC 4931 et le présent document sont des soumissions individuelles, fondées sur le travail effectué dans la RFC 3731.

Des suggestions spécifiques incorporées dans le présent document ont été fournies par Joe Abley, Chris Bason, Eric Brunner-Williams, Jordyn Buchanan, Dave Crocker, Ayesha Damaraju, Anthony Eden, Sheer El-Showk, Klaus Malorny, Dan Manley, Michael Mealling, Patrick Mevzek, Asbjorn Steira, Bruce Tonkin, et Rick Wesson.

9. Références

9.1 Références normatives

- [RFC0952] K. Harrenstien, M. Stahl, E. Feinler, "Spécification du tableau des hôtes de l'Internet du DOD", octobre 1985
- [RFC1123] R. Braden, éditeur, "Exigences pour les hôtes Internet – [Application et prise en charge](#)", STD 3, octobre 1989
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997
- [RFC3688] M. Mealling, "[Registre XML de l'IETF](#)", BCP 81, janvier 2004
- [RFC5730] S. Hollenbeck, "[Protocole d'approvisionnement extensible \(EPP\)](#)", STD0069, août 2009. (*Remplace la RFC4930*)
- [RFC5732] S. Hollenbeck, "Protocole d'approvisionnement extensible ([EPP](#)) : [Transposition d'hôte](#)", STD0069, août 2009. (*Remplace la RFC4932*)
- [RFC5733] S. Hollenbeck, "Protocole d'approvisionnement extensible ([EPP](#)) : [Transposition de contact](#)", STD0069, août 2009. (*Remplace la RFC4933*)
- [xml-20040204] Sperberg-McQueen, C., Maler, E., Yergeau, F., Paoli, J., and T. Bray, "Extensible Markup Language (XML) 1.0 (Third Edition)", World Wide Web Consortium FirstEdition REC-xml-20040204, février 2004, <<http://www.w3.org/TR/2004/REC-xml-20040204>>.
- [xmlschema-1-20041028] Maloney, M., Thompson, H., Mendelsohn, N., and D. Beech, "XML Schema Part 1: Structures Second Edition", World Wide Web Consortium Recommendation REC-xmlschema-1-20041028, octobre 2004, <<http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>>.
- [xmlschema-2-20041028] Malhotra, A. and P. Biron, "XML Schema Part 2: Datatypes Second Edition", World Wide Web Consortium Recommendation REC-xmlschema-2-20041028, octobre 2004, <<http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>>.

9.2 Références pour information

- [RFC2781] P. Hoffman et F. Yergeau, "UTF-16, un codage de la norme ISO 10646", février 2000
- [RFC3490] P. Faltstrom et autres, "Internationalisation des noms de domaine dans les applications (IDNA)", mars 2003. (*Remplacée par les RFC5890 et 5891, P.S.*)
- [RFC4931] S. Hollenbeck, "Transposition des noms de domaine avec le protocole d'approvisionnement extensible (EPP)", mai 2007. (Remplace [RFC3731](#)) (*Remplacée par la RFC5731*, STD 69)

Appendice A. Changements depuis la RFC 4931

1. On a changé "Le présent document rend obsolète la RFC 3731" en "Le présent document rend obsolète la RFC 4931".
2. On a remplacé la référence à la RFC 3731 par la référence à la 4931.
3. On a remplacé la référence à la RFC 4930 par la référence à la 5730.
4. On a remplacé la référence à la RFC 4932 par la référence à la 5732.
5. On a remplacé la référence à la RFC 4933 par la référence à la 5733.
6. On a mis à jour la description de l'état inactif au paragraphe 2.3.

7. On a corrigé l'exemple des noms d'hôtes aux paragraphes 1.1 et 3.2.1.
8. On a changé "mais de telles méthodes NE DEVRAIENT PAS être utilisées" en "mais de telles méthodes ne devraient pas être utilisées" au paragraphe 2.7.
9. Au paragraphe 3.2, on a ajouté "D'autres méthodes de notification PEUVENT être utilisées en plus du message de service exigé".
10. Au paragraphe 3.2, on a ajouté le texte du code de réponse 2201.
11. Ajout du texte de licence BSD à la section de schéma XML.

Adresse de l'auteur

Scott Hollenbeck
VeriSign, Inc.
21345 Ridgetop Circle
Dulles, VA 20166-6503
USA
mél : shollenbeck@verisign.com