

Groupe de travail Réseau
Request for Comments : 2483
Catégorie : Expérimentale
Traduction Claude Brière de L'Isle

M. Mealling, Network Solutions, Inc.
R. Daniel, Jr., Los Alamos National Laboratory
janvier 1999

Services de résolution d'URI nécessaires pour la résolution d'URN

Statut de ce mémoire

Le présent mémoire définit un protocole expérimental pour la communauté de l'Internet. Il ne définit en aucune façon une norme de l'Internet. Des discussions et suggestions sont demandées pour son amélioration. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de copyright

Copyright (C) The Internet Society (1999). Tous droits réservés.

Résumé

La restitution de la ressource identifiée par un identifiant de ressource uniforme (URI, *Uniform Resource Identifier*) [RFC1630] est seulement une des opérations qui peuvent être effectuées sur un URI. On peut aussi demander et obtenir une liste des autres identifiants qui sont des alias pour l'URI original ou une description bibliographique de la ressource notée par l'URI, par exemple. Ceci s'applique aux noms de ressource uniforme (URN, *Uniform Resource Name*) et aux localisateurs de ressource uniforme (URL, *Uniform Resource Locator*). Les caractéristiques de ressource uniforme (URC, *Uniform Resource Characteristic*) ne sont discutées dans le présent document que comme descriptions de ressources plutôt que d'identifiants.

Un service qui fournit dans le réseau l'accès à une ressource peut fournir une ou plusieurs de ces options, mais il n'a pas besoin de les fournir toutes. Le présent mémoire spécifie un ensemble initial des opérations qui peuvent être utilisées pour décrire les interactions fournies par un certain service d'accès. Il suggère aussi des lignes directrices qu'on devrait suivre lorsque ces opérations sont codées dans un protocole.

Table des Matières

1.	Introduction.....
2.	Spécification générale.....
3.	Opérations de codage.....
4.	Ensemble incomplet.....
5.	Type de support Internet text/uri-list.....
6.	Considérations pour la sécurité.....
7.	Références.....
8.	Adresse des auteurs.....
9.	Déclaration complète de droits de reproduction.....

1. Introduction

Lors de la formulation de propositions récentes [RFC2168] concernant les URN [RFC2141], il est devenu évident que demander aux serveurs de gérer toutes les fonctions désirées ou demander aux clients de traiter les diverses informations retournées par un serveur était irréaliste et une barrière à leur adoption. Il est nécessaire qu'il y ait un moyen pour qu'un client soit capable d'identifier un serveur qui se spécialise dans le complexe et un autre qui se spécialise dans le simple (mais rapide). Aussi, dans des conversations ultérieures il est devenu évident que, dans la plupart des cas, certaines des opérations étaient inappropriées ou difficiles pour certains identifiants.

Le problème

Dans le processus d'acquisition de connaissances sur une ressource de l'Internet, il y a diverses fonctions possibles qui peuvent être importantes et/ou utiles, comme la découverte de localisateurs, de noms, de descriptions, et l'accès à la ressource elle-même. Un service donné peut ne prendre en charge qu'un sous-ensemble d'entre elles ; donc, il est important de décrire un tel service d'accès par les types de fonctions prises en charge et les ressources dont il a connaissance. Par exemple, dans le cadre d'un service de découverte de résolveur (RDS, *Resolver Discovery Service*) décrit dans la [RFC2276], le RDS lui-même peut fournir des URL [RFC1736], [RFC1738], alors que les résolveurs peuvent fournir des descriptions, des URL, ou même les ressources elles-mêmes. Le concept de RDS tel que proposé dans la [RFC2168], peut être plus généreux et fournir tout ce qui précède.

Ce problème exige un ensemble bien compris d'identifiants qui spécifient ces opérations. Mais un ensemble exhaustif serait à la fois impossible et pas très utile. Donc, le présent document va énumérer plusieurs opérations, ainsi que dresser la liste des exigences pour spécifier de nouvelles opérations.

L'objet du présent document est de définir une liste des fonctions et de leurs noms abrégés et de les utiliser pour définir l'interface à un service d'accès. Les versions précédentes de ce document se référaient à des services où les arguments étaient des types spécifiques d'URI tels que les URN ou les URL. Ces services étaient appelés, par exemple, "N2L" et "L2L". Leur utilisation a été changée en faveur de la forme plus générale d'URI.

Critères de conception

Pour satisfaire à ces exigences, un critère de conception très simple a été utilisé. Le besoin d'identifier l'opération avec un jeton de telle sorte que son opérande, son algorithme, et ses erreurs, soient connus, se révèle suffisant pour satisfaire à ces exigences.

2. Spécification générale

Pour fournir un cadre à la fois aux spécifications du présent document et aux travaux futurs que d'autres écriront, les lignes directrices qui suivent sont suggérées pour les documents qui cherchent à spécifier de nouvelles opérations. Toute spécification d'un membre de cet ensemble d'opérations devrait traiter ces questions par rapports à ses opérandes, son algorithme, son résultat, et ses erreurs.

Du fait du petit nombre de fonctions énumérées, un mécanisme d'enregistrement a été jugé prématuré. Si cette liste devait croître, un mécanisme d'enregistrement serait probablement nécessaire.

Du fait aussi de la nature expérimentale de ce document et des systèmes qui utilisent ces spécifications, l'utilisation des mots DOIT et DEVRA est limitée. Lorsque ils sont utilisés, ils reflètent un cas où la présente spécification pourrait causer un dommage aux systèmes existants, non expérimentaux, tels que HTTP et les URN. Donc, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" dans ce document sont à interpréter comme décrit dans la RFC2119.

2.1 Opérandes

Les opérandes doivent contenir les éléments d'information suivants :

- * nom de l'opération,
- * mnémonique insensible à la casse de l'opération,
- * nombre des opérandes,
- * type de chaque opérande,
- * format de chaque opérande.

2.2 Algorithme

L'algorithme exact de l'opération doit soit être spécifié complètement, soit il doit être considéré comme opaque et défini par le serveur ou l'application.

2.3 Résultat

Le résultat doit spécifier une des situations suivantes :

- * il n'y a pas de résultat,
- * le résultat est indéfini,
- * le résultat lui-même et son contenu,
- * le fait que le résultat est un objet, le type et le format de l'objet,
- * toutes erreurs non spécifiques du protocole.

2.4 Conditions d'erreur

Toutes les erreurs qui sont considérées applicables à travers tous les environnements de mise en œuvre et d'application doivent être incluses. Les erreurs qui dépendent du système qui porte le service ne sont pas incluses. Donc, nombre des erreurs attendues comme la disponibilité du service ou la syntaxe des opérations ne sont pas incluses dans ce document car elles dépendent de la mise en œuvre.

2.5 Considérations pour la sécurité

Toutes considérations relatives à la sécurité du service fourni doivent être spécifiées. Cela N'INCLUT PAS les considérations sur le protocole utilisé pour convoier le service ou celles qui accompagnent normalement les résultats du service. Par exemple, un service qui retourne un seul URL aurait besoin de discuter de la situation où un malveillant insère un URI incorrect dans le résolveur mais PAS le cas où quelqu'un envoie des informations personnelles à travers l'Internet à la ressource identifiée par l'URL correct.

3 Opérations de codage

Pour être utiles, ces opérations doivent être utilisées au sein d'un système ou protocole. Dans de nombreux cas, ces systèmes et protocoles vont mettre des restrictions sur les opérations admises et sur comment celles qui ont un sens sont syntaxiquement représentées. Il est suffisant pour ces protocoles de définir les nouvelles opérations au sein de leurs propres documents de spécification de protocole mais il faut veiller à bien faire connaître ce fait.

Aussi, un certain système ou protocole va avoir ses propres spécifications de résultat qui peuvent restreindre les formats du résultat d'une certaine opération. De plus, un certain protocole peut avoir une meilleure solution pour un résultat que celles données ici. Par exemple, le résultat d'une opération qui convertit un URI en plus d'un URL peut être codé d'une manière spécifique du protocole qui porte les informations sur la proximité de chaque ressource sur le réseau.

Donc, les exigences sur le codage de ces opérations au sein d'un système donné sont les suivantes :

- * quel sous-ensemble des opérations est admis,
- * comment est codé l'opérateur,
- * comment sont codées les opérandes,
- * comment sont retournés les codes d'erreur.

Le type de support MIME text/uri-list est spécifié à la Section 5. Ce type de support est simplement une suggestion pour les systèmes expérimentaux qui ont besoin d'une mise en œuvre simple. Il n'est inclus ici que comme simple exemple pour être complet (quelque soit sa simplicité).

4. Ensemble incomplet

4.1 I2L (d'URI en URL)

- * Nom : d'URI en URL
- * Mnémonique : I2L
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque.
- * Résultat : un URL et un seul.
- * Conditions d'erreur :
 - o URI mal formé.
 - o URI syntaxiquement valide mais n'existe sous aucune forme.
 - o L'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o L'URI a existé dans le passé mais on ne sait actuellement rien sur lui.
 - o L'accès est refusé.
- * Considérations pour la sécurité :
 - o Redirection malveillante : Un des dangers fondamentaux qui se rapporte à tout service comme celui-là est qu'une entrée malveillante dans la base de données d'un résolveur cause la résolution en un mauvais URL de l'URI par les clients. L'intention est peut-être de faire que le client restitue une ressource contenant des matériaux frauduleux ou dommageables.
 - o Déni de service : En supprimant l'URL en lequel transpose l'URI, un intrus malveillant peut supprimer la capacité du client à restituer la ressource.

Cette opération est utilisée pour transposer un seul URI en un seul URL. Elle est utilisée par des clients légers qui n'ont pas la capacité de choisir sur une liste d'URL ou de comprendre un URC. L'algorithme pour cette transposition dépend du schéma d'URI.

4.2 I2Ls (d'un URI à des URL)

- * Nom : d'un URI à des URL
- * Mnémonique : I2LS
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : une liste de zéro, un ou plusieurs URL.
- * Erreurs :
 - o URI mal formé,
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o L'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o L'URI a existé dans le passé mais on ne sait rien de lui actuellement.
 - o Accès refusé.
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir en I2L)
 - o Déni de service (voir en I2L)

Cette opération est utilisée pour transposer un seul URI en zéro, un ou plusieurs URL. Elle est utilisée par un client qui peut prendre un URL dans une liste sur la base de critères qui sont importants pour le client. Le client ne devrait pas faire de suppositions sur l'ordre des URL retournés. Quel que soit le type de support, le résultat devrait être une liste des URL qui peuvent être utilisés pour obtenir une instance de la ressource identifiée par l'URI. Tous les URI devront être codés conformément aux spécifications des URL [RFC1738] et des URN [RFC2141].

4.3 I2R (d'URI en ressource)

- * Nom : d'URI en ressource
- * Mnémonique : I2R
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : une instance de la ressource désignée par l'URI.
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o L'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o L'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o Accès refusé.
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Déni de service (voir à I2L)

Cette opération est utilisée pour retourner une seule instance de la ressource désignée par l'URI. Le format du résultat dépend de la ressource elle-même.

4 I2Rs (d'URI en ressources)

- * Nom : d'URI en ressources
- * Mnémonique : I2Rs
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : Zéro, une ou plusieurs instances de la ressource désignée par l'URI.
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o l'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o l'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Déni de service (voir à I2L)

Cette opération est utilisée pour retourner plusieurs instances d'une ressource, par exemple, des versions GIF et JPEG d'une image. Le jugement sur le fait que les ressources sont "les mêmes" appartient à l'autorité de désignation qui a produit l'URI.

Le résultat devra être un message MIME multipartiel/alternatif [RFC2045] avec les versions de remplacement de la ressource dans une partie de corps séparée. Si il n'y a qu'une seule version de la ressource identifiée par l'URN, elle PEUT être retournée sans l'enveloppe multipartiel/alternative.

4.5 I2C (d'URI en URC)

- * Nom : d'URI en URC
- * Mnémonique : I2C
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : un URC
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o l'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o l'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Déni de service (voir à I2L)

Les caractéristiques de ressource uniforme sont des descriptions de ressources. Cette demande permet au client d'obtenir une description de la ressource identifiée par un URI, par opposition à la ressource elle-même ou simplement des URL de la ressource. La description peut être une citation bibliographique, une signature numérique, ou un historique des révisions. Le présent mémoire ne spécifie pas le contenu d'une réponse à une demande d'URC. Ce contenu est supposé varier d'un serveur à l'autre.

4.6 I2CS (d'un URI à des URC)

- * Nom : d'un URI à des URC
- * Mnémonique : I2CS
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : Zéro, une ou plusieurs URC
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o l'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o l'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Déni de service (voir à I2L)

Les URC peuvent venir sous différents formats et types. Cette opération retourne zéro, une ou plusieurs URC qui sont appropriées pour cet URI.

4.7 I2N (d'URI à URN)

- * Nom : d'URI à URN
- * Mnémonique : I2N
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URN.
- * Format de chaque opérande : le premier opérande est codé comme un URI.

- * Algorithme : opaque
- * Résultat : un URN et un seul
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o l'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o l'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Dénî de service (voir à I2L)

Bien que les URN soient supposés identifier une ressource et une seule, cela ne signifie pas qu'une ressource puisse n'avoir qu'un seul URN. Par exemple, considérons une ressource qu'une organisation souhaite appeler "foo" ; une autre organisation, en accord avec la première, veut appeler la ressource "bar". Les deux organisations peuvent s'accorder pour que les deux noms "désignent" la même ressource et pour que les URN "foo" et "bar" soient équivalents.

Le résultat est un URN, connu du serveur, qui identifie la même ressource comme URN d'entrée.

Un soin extrême devrait être porté à ce service car il joue avec l'idée d'égalité par rapport aux URN. Comme mentionné dans plusieurs documents sur les URN, l'idée d'égalité est très spécifique du domaine. Par exemple, un URN qui pointe sur une carte météorologique pour un jour particulier et un URN pointant sur la carte qui change de jour en jour NE SERAIT PAS retournée dans cet exemple parce qu'elles pointent sur des ressources différentes. Un autre concept d'équivalence temporaire est à l'œuvre. Ce service traite plutôt de ressources qui ont deux noms différents alors qu'il y a un lien entre les noms qui a été accepté par les deux allocataires de noms. C'est-à-dire que les deux espaces de noms DOIVENT s'être mis d'accord pour que chaque nom puisse être utilisé à la place de l'autre et que la signification ne change pas.

4.8 I2Ns (d'un URI à des URN)

- * Nom : d'un URI à des URN
- * Mnémonique : I2NS
- * Nombre d'opérandes : 1
- * Type de chaque opérande : le premier opérande est un URI.
- * Format de chaque opérande : le premier opérande est codé comme un URI.
- * Algorithme : opaque
- * Résultat : une liste d'URN
- * Erreurs :
 - o URI mal formé
 - o URI syntaxiquement valide mais qui n'existe sous aucune forme.
 - o l'URI existe mais il n'y a pas de résultat disponible à partir de cette opération.
 - o l'URI a existé dans le passé mais on ne sait plus rien sur lui.
 - o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Dénî de service (voir à I2L)

Cette opération retourne simplement zéro, une ou plusieurs URN suivant les mêmes critères et précautions que l'opération I2N.

4.9 I=I (l'URI est-il égal à l'URI)

- * Nom : URI = URI
- * Mnémonique : I=I
- * Nombre d'opérandes : 2
- * Type de chaque opérande : les deux opérandes sont des URI.
- * Format de chaque opérande : les deux opérandes sont codés comme des URI.
- * Algorithme : opaque
- * Résultat : VRAI ou FAUX
- * Erreurs :
 - o URI mal formés
 - o les URI sont syntaxiquement valides mais n'existent sous aucune forme.
 - o les URI existent mais il n'y a pas de résultat disponible à partir de cette opération.

- o les URI ont existé dans le passé mais on ne sait plus rien sur eux.
- o accès refusé
- * Considérations pour la sécurité :
 - o Redirection malveillante (voir à I2L)
 - o Déni de service (voir à I2L)

Cette opération est utilisée pour déterminer si deux URI donnés sont considérés comme égaux par le serveur auquel la question est posée. L'algorithme utilisé pour déterminer l'égalité est opaque. Aucune assertion n'est faite sur le fait que les URI exhibent ou non des caractéristiques d'URN ou d'URL.

5. Type de support Internet text/uri-list

Plusieurs types de demandes de service de résolution, comme I2Ls, I2Ns, résultent en une liste d'URI qui sont retournés au client. Le type de support Internet text/uri-list est défini comme fournissant un format simple pour le traitement automatique de telles listes d'URI.

Voici une copie de l'enregistrement par l'IANA du type de support text/uri-list.

Date: Fri, 18 Apr 97 08:36:07 PDT
From: Ron Daniel Jr. <rdaniel@lanl.gov>
To: iana@iana.org, rdaniel@lanl.gov
Subject: Request for MIME media type Text/IETF Tree - uri-list

Name : Ron Daniel Jr.
E-mail : rdaniel@lanl.gov
MIME media type name : Text
MIME subtype name : IETF Tree -uri-list
Required parameters : none
Optional parameters : charset

Actuellement, les URI peuvent être représentés en utilisant l'US-ASCII. Cependant, il y a de nombreux URI non standard qui utilisent des jeux de caractères particuliers. La discussion sur la meilleure façon de réaliser l'internationalisation des URI est en cours. Cet enregistrement sera mise à jour une fois que la discussion de la façon dont les jeux de caractères d'URI peuvent être acceptée sera arrivée à une conclusion.

Considérations sur le codage : Certains protocoles de transfert, comme SMTP, mettent des limites à la longueur des lignes. Les très longs URI peut dépasser ces limites. Les systèmes doivent donc être prêts à utiliser un codage de transfert de contenu convenable. On prévoit que ce cas devrait être assez rare.

Considérations pour la sécurité : Les logiciels clients devraient être au fait des considérations pour la sécurité des URI. Par exemple, l'accès à certains URI peut résulter en l'envoi d'une menace de mort à l'égard d'un chef d'état, déclenchant fréquemment la visite du service de protection approprié. Accéder à d'autres URI peut résulter en obligations financières, ou en l'accès à des ressources considérées comme inappropriées par votre employeur.

Bien que le fournisseur légitime d'une uri-list puisse exploiter ces propriétés pour le bien ou pour le mal, il est très vraisemblable que des uri-list seront falsifiées afin d'exploiter ces caractéristiques des URI.

De plus, le potentiel de recherche et de recherche inverse de l'uri-list peut attirer les analystes de trafic. Les listes d'URI peuvent aussi révéler des informations confidentielles, comme la localisation d'informations sensibles.

À cause de ces considérations, des mesures de confidentialité externes devaient être disponibles pour protéger les réponses d'uri-list quand c'est approprié.

Considérations d'interopérabilité : aucune n'est connue.

Spécification publiée : Les localisateurs de ressource uniforme (URL) et les noms de ressource uniforme (URN) sont deux instances d'une classe plus générale d'identifiants connus comme identifiants de ressource uniforme (URI). Les méthodes de résolution d'URN souhaitent fréquemment retourner des listes d'URL pour une ressource afin de réaliser la tolérance aux fautes et l'équilibrage de charge. Le format text/uri-list est destiné à être un format très simple pour communiquer de telles listes d'URL (et d'URN) sous une forme convenable pour le traitement automatique.

Le format des ressources text/uri-list est :

- 1) Toute ligne commençant par le caractère '#' est une ligne de commentaires et est ignorée durant le traitement. (Noter que les URI peuvent contenir le caractère '#', de sorte qu'il n'est un caractère de commentaire que lorsque il est le premier caractère d'une ligne.)
- 2) Les lignes restantes qui ne sont pas des lignes de commentaires devront être des URI (URN ou URL) codées conformément aux spécifications d'URL ou d'URN (RFC2141, RFC1738 et RFC2396). Chaque URI devra apparaître sur une ligne et une seule. Les très longs URI ne sont pas cassés dans le format text/uri-list. Les codages de transfert de contenu peuvent être utilisés pour mettre en application les limitations de longueur de ligne.
- 3) Comme pour tous les formats text/*, les lignes sont terminées par une paire CRLF.

Dans les applications où un URI a été transposé en une liste d'URI, la première ligne de la réponse text/uri-list DEVRAIT être un commentaire donnant l'URI original.

Un exemple du format figure ci-dessous :

```
# urn:isbn:0-201-08372-8
http://www.huh.org/books/foo.html
http://www.huh.org/books/foo.pdf
ftp://ftp.foo.org/books/foo.txt
```

Applications qui utilisent ce support : les résolveurs d'URN sont les applications initiales. Les clients et mandataires de la Toile sont les applications qui devront vraisemblablement prendre en charge ce format à l'avenir.

Informations supplémentaires :

1. Numéros magiques : aucun pour l'instant
2. Extensions de fichiers : on recommande .uris ou .uri
3. Code de type de fichier Macintosh : URI recommandé

Ce type de support est le produit du groupe de travail URN de l'IETF.

Personne à contacter pour des informations complémentaires :

1. Nom : Ron Daniel Jr.
2. mél : rdaniel@lanl.gov

Usage prévu : utilisation limitée

Le type de support text/uri-list est destiné à être utilisé dans les applications qui utilisent les URI pour la réplique de ressources.

Auteur/Contrôleur des modifications : Ron Daniel Jr.

Los Alamos National Laboratory
rdaniel@lanl.gov

Dans les applications où un URI a été transposé en une liste d'URI, comme les réponses aux demandes I2Ls, la première ligne de la réponse de text/uri-list DEVRAIT être un commentaire donnant l'URI original. Un exemple de ce résultat pour la demande I2L est montré à la Figure 1.

6. Considérations pour la sécurité

Les communications avec un serveur peuvent être de nature sensible. Certains serveurs vont détenir des informations qui ne devraient être livrées qu'à des utilisateurs autorisés. Les résultats provenant des serveurs peuvent être la cible de mystifications, en particulier lorsque les transactions de commerce électronique sont courantes et qu'on peut gagner de l'argent en redirigeant les usagers sur des sites pirates plutôt que sur ceux qui payent des redevances aux détenteurs de droits. Les demandes de serveurs peuvent intéresser les analystes de trafic. Les demandes peuvent aussi être l'objet de mystifications.

Le message d'erreur "Accès refusé" suppose que le système au sein duquel les opérations sont effectuées peut porter un concept de contrôle d'accès authentifié. Donc, le message "Accès refusé" ne devrait être retourné que par les systèmes qui ont une méthode appropriée de détermination du contrôle d'accès.

7. Références

- [RFC1630] T. Berners-Lee, "Identifiants de ressource universels dans la Toile mondiale ; [syntaxe unificatrice](#) pour l'expression des noms et adresses des objets du réseau utilisés sur la Toile mondiale", juin 1994. (*Information*)
- [RFC2168] R. Daniel, M. Mealling, "Résolution des identifiants de ressource uniformes avec le système des noms de domaines", juin 1997. (*Obsolète, voir [RFC3401](#), [RFC3402](#), [RFC3403](#), [RFC3404](#)*) (*MàJ par [RFC2915](#)*) (*Expérimentale*)
- [RFC2141] R. Moats, "[Syntaxe des URN](#)", mai 1997.
- [RFC2045] N. Freed et N. Borenstein, "Extensions de [messagerie Internet](#) multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. (*D. S., MàJ par les [RFC2184](#), [2231](#), [5335](#).*)
- [RFC2276] K. Sollins, "Principes d'[architecture de la résolution](#) de nom de ressource uniforme", janvier 1998. (*MàJ par [RFC3401](#)*) (*Information*)
- [RFC1736] J. Kunze, "Recommandations fonctionnelles pour les localisateurs de ressource Internet", février 1995. (*Info*)
- [RFC1738] T. Berners-Lee et autres, "[Localisateurs uniformes](#) de ressource (URL)", décembre 1994. (*Obsolète, voir les [RFC4248](#) et [4266](#)*)

8. Adresse des auteurs

Michael Mealling
Network Solutions
505 Huntmar Park Drive
Herndon, VA 22070
téléphone : (703) 742-0400
Fax : (703) 742-9552
mél : michaelm@rwhois.net

Ron Daniel
Advanced Computing Lab, MS B287
Los Alamos National Laboratory
Los Alamos, NM, USA, 87545
téléphone : (505) 665-0597
Fax : (505) 665-4939
mél : rdaniel@lanl.gov

9. Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (1999). Tous droits réservés.

Ce document et les traductions de celui-ci peuvent être copiés et diffusés, et les travaux dérivés qui commentent ou expliquent autrement ou aident à sa mise en œuvre peuvent être préparés, copiés, publiés et distribués, partiellement ou en totalité, sans restriction d'aucune sorte, à condition que l'avis de copyright ci-dessus et ce paragraphe soit inclus sur toutes ces copies et œuvres dérivées. Toutefois, ce document lui-même ne peut être modifié en aucune façon, par exemple en supprimant le droit d'auteur ou les références à l'Internet Society ou d'autres organisations Internet, sauf si c'est nécessaire à l'élaboration des normes Internet, auquel cas les procédures pour les droits de reproduction définies dans les processus des normes pour l'Internet doivent être suivies, ou si nécessaire pour le traduire dans des langues autres que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Société Internet, ses successeurs ou ayants droit.

Ce document et les renseignements qu'il contient sont fournis "TELS QUELS" et l'INTERNET SOCIETY et l'INTERNET ENGINEERING TASK FORCE déclinent toute garantie, expresse ou implicite, y compris mais sans s'y limiter, toute garantie que l'utilisation de l'information ici présente n'enfreindra aucun droit ou aucune garantie implicite de commercialisation ou d'adaptation à un objet particulier.

Remerciement

Le financement de la fonction d'éditeur des RFC est actuellement assuré par la Internet Society.