

Groupe de travail Réseau
Request for Comments : 2017
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

N. Freed, Innosoft International
K. Moore, University of Tennessee
A. Cargille, président du GT
octobre 1996

Définition du type URL d'accès de corps extérieur MIME

Statut du présent mémoire

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

1. Résumé

Le présent mémoire définit un nouveau type d'accès pour les parties de message/corps externe MIME pour les localisateurs uniformes de ressource (URL, *Uniform Resource Locator*). Les URL permettent des schémas pour accéder à des objets externes via un nombre croissant de protocoles, incluant HTTP, Gopher, et TELNET. Un ensemble initial de schémas d'URL est défini dans la [RFC1738].

2. Introduction

Les extensions multi objet de message Internet (MIME, *Multipurpose Internet Message Extensions*) définissent une facilité par laquelle un objet peut contenir une référence ou pointeur à une forme de données plutôt qu'aux données réelles elles-mêmes. Cette facilité est incorporée dans le type de support du message/corps externe défini dans la [RFC1521]. L'utilisation de cette facilité est croissante comme moyen de préserver la bande passante lorsque de gros objets sont envoyés à de grandes listes de diffusion.

Chaque référence de message/corps externe doit spécifier un mécanisme par lequel des données réelles peuvent être restituées. Ces mécanismes sont appelés des types d'accès, et la [RFC1521] définit un ensemble initial de types d'accès : "FTP", "ANON-FTP", "TFTP", "LOCAL-FILE", et "MAIL-SERVER".

Les localisateurs uniformes de ressources (URL, *Uniform Resource Locator*) fournissent aussi un moyen par lequel les données distantes peuvent être restituées automatiquement. Chaque chaîne d'URL commence par une spécification de schéma, qui à son tour spécifie comment la chaîne restante doit être utilisée en conjonction avec un certain protocole pour restituer les données. Cependant, les schémas d'URL existent pour des opérations de protocole qui n'ont pas de type d'accès de message/corps externe MIME correspondant. Enregistrer un type d'accès pour les URL fournit donc au message/corps externe un accès aux mécanismes de restitution des URL qui ne sont pas actuellement disponibles comme type d'accès. Cela donne aussi accès à tous les mécanismes futurs pour lesquels des schémas d'URL seront développés.

Ce type d'accès n'est pas seulement destiné à être utilisé avec les URL qui restituent bien quelque chose. D'autres mécanismes d'URL, par exemple, mailto, ne peuvent pas être utilisés dans ce contexte.

3. Définition du type d'accès URL

Le type d'accès URL se définit comme suit :

- (1) Le nom du type d'accès est URL.
- (2) Un nouveau paramètre de type de contenu message/corps externe est utilisé pour mémoriser réellement la chaîne d'URL. Le nom du paramètre est aussi "URL", et ce paramètre est obligatoire pour ce type d'accès. La syntaxe et l'utilisation de ce paramètre sont spécifiées dans le paragraphe suivant.
- (3) La zone de corps fantôme du message/corps externe n'est pas utilisée et devrait être laissée en blanc.

Par exemple, le message suivant illustre comment le type d'accès URL est utilisé :

Content-type: message/external-body; access-type=URL;
 URL="http://www.foo.com/file"

Content-type: text/html
 Content-Transfer-Encoding: binary

CECI N'EST EN RÉALITÉ PAS LE CORPS !

3.1 Syntaxe et utilisation du paramètre URL

En utilisant la notation ANBF et les définitions de la [RFC0822] et de la [RFC1521], la syntaxe du paramètre URL est la suivante :

paramètre-URL := <"> mot-URL *(*LWSP-char mot-URL) <">

mot-URL := jeton ; Ne doit pas excéder 40 caractères de long

La syntaxe d'une chaîne URL réelle est donnée dans la [RFC1738]. Les chaînes URL peuvent être de toutes longueurs et peuvent contenir des caractères arbitraires. Cela génère des problèmes lorsque les URL sont incorporés dans des en-têtes de partie de corps MIME qui reviennent à zéro selon les règles de la [RFC0822]. C'est pour cette raison qu'ils sont transformés en un paramètre URL par l'inclusion dans une spécification de type de contenu de message/corps externe comme suit :

- (1) On fait une vérification pour s'assurer que toutes les occurrences de caractères ESPACE, CTL, double guillemets, barre oblique inverse, et caractère à 8 bits dans la chaîne URL sont déjà codés en utilisant le schéma de codage d'URL spécifié dans la [RFC1738]. Toutes les occurrences non codées de ces caractères doivent être codées. Noter que le résultat de cette opération n'est rien de plus qu'une représentation différente de l'URL d'origine.
- (2) La chaîne d'URL résultante est cassée en sous-chaînes d'au plus 40 caractères.
- (3) Chaque sous chaîne est placée dans une chaîne paramètre-URL comme un mot-URL, séparé par une ou plusieurs espaces. Noter que les guillemets qui les entourent sont toujours exigés dans la mesure où tous les URL contiennent un ou plusieurs deux-points, et que les deux-points sont des caractères tspecial selon la [RFC1521].

L'extraction de la chaîne URL du paramètre-URL est encore plus simple : les guillemets qui l'entourent et toutes les espaces linéaires sont retirés et les matériaux qui restent sont la chaîne d'URL.

L'exemple qui suit montre comment est traité un long URL :

Content-type: message/external-body; access-type=URL;
 URL="ftp://ftp.deepdirs.org/1/2/3/4/5/6/7/8/9/10/11/12/13/14/15/16/17/18/20/21/file.html"

Content-type: text/html
 Content-Transfer-Encoding: binary

CECI N'EST EN RÉALITÉ PAS LE CORPS !

Certains URL peuvent donner accès à plusieurs versions du même objet dans différents formats. Le mécanisme HTTP d'URL a cette capacité, par exemple. Cependant, les applications ne peuvent pas s'attendre à recevoir quelque chose dont le type ne serait pas en accord avec celui qui est exprimé dans le message/corps externe, et peut en fait avoir déjà fait des choix irrévocables sur la base de ces informations.

Du fait de ces considérations, les restrictions suivantes sont imposées : lorsque les URL sont utilisés dans le contexte d'un type d'accès, seules les versions d'un objet dont le type de contenu sont en accord avec celles spécifiées dpar l'en-tête du message interne/corps extérieur peuvent être restituées et utilisées.

4. Considérations pour la sécurité

Les problème de sécurité de l'utilisation des URL dans le contexte d'un type d'accès MIME ne sont pas différents de ceux qui surviennent lors de leur utilisation dans d'autres contextes. Les considérations de sécurité spécifiques associées à chaque type d'URL sont exposées dans le document de définition de l'URL.

Noter que le champ Content-MD5 peut être utilisé en conjonction avec tout type d'accès de message/corps externe pour fournir une vérification d'intégrité. Cela assure que l'objet référencé est réellement celui que prévoyait celui qui a généré le message. Ce n'est pas un service de signature et il ne devrait pas être confondu avec une signature, mais il est néanmoins assez utile dans de nombreuses situations.

5. Remerciements

Les auteurs sont reconnaissants des retours et révisions fournis par John Beck et John Klensin.

6. Références

- [RFC0822] D. Crocker, "Norme pour le [format des messages de texte](#) de l'ARPA-Internet", STD 11, août 1982. (*Obsolète, remplacée par la RFC5322*)
- [RFC1521] N. Borenstien et N. Freed, "MIME (Extensions [multi-usage de messagerie Internet](#)) Partie 1 : Mécanismes pour spécifier et décrire le format des corps de message Internet", septembre 1993. (*Rendue obsolète par les RFC2045 à 2049*)
- [RFC1590] J. Postel, "Procédures d'enregistrement des types de support" mars 1994. (*Information*) (*Obsolète, remplacée par les RFC2045-49*)
- [RFC1738] T. Berners-Lee et autres, "[Localisateurs uniformes de ressource](#) (URL)", décembre 1994. (*P.S., Obsolète, voir les RFC4248 et 4266*)

7. Adresse des auteurs

Ned Freed
Innosoft International, Inc.
1050 East Garvey Avenue South
West Covina, CA 91790
USA
téléphone : +1 818 919 3600
fax : +1 818 919 3614
mél : ned@innosoft.com

Keith Moore
Computer Science Dept.
University of Tennessee
107 Ayres Hall
Knoxville, TN 37996-1301
USA
mél : moore@cs.utk.edu