

Groupe de travail Réseau
Request for Comments : 3516
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

L. Nerenberg, Orthanc Systems

avril 2003

Extension de contenu binaire à IMAP4

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) The Internet Society (2003). Tous droits réservés

Résumé

Le présent mémoire définit l'extension "Binaire" au protocole d'accès au message Internet (IMAP4, *Internet Message Access Protocol*). Il fournit aux clients et serveurs IMAP4 un mécanisme pour échanger des données de corps de message sans utiliser le codage de transfert de contenu MIME.

1. Conventions utilisées dans le document

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans le BCP 14, [RFC2119].

L'abréviation "CTE" signifie "content-transfer-encoding", c'est-à-dire "codage de transfert de contenu".

2. Introduction

Les extensions MIME à la messagerie Internet permettent la transmission de contenu de message non textuel (binaire) [RFC2045]. Comme les transports traditionnels pour la messagerie ne sont pas toujours capables de passer des données binaires de façon transparente, MIME fournit des schémas de codage qui permettent qu'un contenu binaire soit transmis sur des transports qui ne serait autrement pas capables de le faire.

La redondance du codage MIME de ce contenu peut être considérable dans certains contextes (par exemple, les liaisons radio lentes, les flux multimédia). Réduire la redondance associée aux schémas de CTE comme le base64 peut apporter une réduction notable de la consommation de ressources. L'extension Binaire laisse le serveur effectuer le décodage de CTE avant de transmettre les données du message au client.

3. Considérations de codage de transfert de contenu

Chaque section de corps IMAP4 a un codage de transfert de contenu MIME. (Ceux qui n'ont pas un en-tête Codage de transfert de contenu explicite sont implicitement étiquetés comme contenu "7bit".) Dans la terminologie de la [RFC2045], le CTE spécifie à la fois un algorithme de décodage et le domaine des données décodées. Dans ce mémoire, "décodage" se réfère à l'étape de décodage de CTE décrite dans la [RFC2045].

Certains CTE utilisent une transformation de codage d'identité. Pour ces CTE, aucun décodage n'est exigé, cependant le domaine des données sous-jacentes peut ne pas pouvoir être exprimé dans le protocole IMAP4 (par exemple, un contenu MIME "binaire" contenant des octets NUL). Pour s'accommoder de ces cas, l'extension Binaire introduit un nouveau type d'élément de protocole littéral qui est pleinement transparent au huit bits.

Donc, le traitement par le serveur de la commande FETCH BINARY implique deux étapes logiques :

- 1) effectuer tout décodage en relation avec le CTE
- 2) déterminer le domaine des données décodées.

L'étape 2 est nécessaire pour déterminer quel élément de protocole devrait être utilisé pour transmettre les données décodées. (Voir les détails sous Extensions de réponse à FETCH.)

4. Cadre pour l'extension IMAP4 Binaire

Le présent mémoire définit les extensions suivantes à la [RFC3501].

4.1 Identification des capacités

Les serveurs IMAP4 qui prennent en charge la présente extension DOIVENT inclure "BINARY" dans la liste en réponse à la commande CAPABILITY.

4.2 Extensions à la commande FETCH

Cette extension définit trois nouveaux éléments de données de la commande FETCH.

BINARY<section-binaire>[<partial>]

Demande que la section spécifiée soit transmise après avoir effectué le décodage en rapport avec le CTE.

L'argument <partial>, si il est présent, demande qu'un sous ensemble des données soit retourné. La sémantique d'une commande FETCH BINARY partielle est la même que pour une commande FETCH BODY partielle, à l'exception que les arguments <partial> se réfèrent aux données de la section décodée.

BINARY.PEEK<section-binaire>[<partial>]

Une autre forme de FETCH BINARY qui n'établit pas implicitement le fanion \Seen.

BINARY.SIZE<section-binaire>

Demande la taille décodée de la section (c'est-à-dire, la taille à attendre en réponse à la demande FETCH BINARY correspondante).

Note: Les auteurs de clients doivent faire attention car ce peut être une opération coûteuse pour certaines mises en œuvre de serveur. Produire cette demande sans nécessité pourrait résulter en une dégradation des performances due au fait que les serveurs doivent calculer la valeur chaque fois que la demande est produite.

4.3 Extensions à la réponse à FETCH

Cette extension définit deux nouveaux éléments de données de réponse à FETCH.

BINARY<section-binaire>[<<nombre>>]

Un <nstring> ou <literal8> exprimant le contenu de la section spécifiée après avoir retiré tout codage en rapport avec le CTE. Si <nombre> est présent, il se réfère au décalage au sein des données de la section décodée.

Si le domaine des données décodées est "8bit" et si les données ne contiennent pas d'octet NUL, le serveur DEVRAIT retourner les données dans une <chaîne> plutôt que dans un <literal8>; cela permet au client de déterminer si les données en "8bit" contiennent l'octet NUL sans avoir à examiner explicitement le flux des données à la recherche d'octets NUL.

Si le serveur ne sait pas comment décoder le CTE de la section, il DOIT faire échouer la demande et produire une réponse "NO" qui contient le code de réponse étendue "UNKNOWN-CTE".

BINARY.SIZE<section-binaire>

Taille de la section après avoir retiré tout codage en rapport avec le CTE. La valeur retournée DOIT correspondre à la taille de la <nstring> ou du <literal8> qui sera retourné par la demande FETCH BINARY correspondante.

Si le serveur ne sait pas comment décoder le CTE de la section, il DOIT faire échouer la demande et produire une réponse "NO" qui contient le code de réponse étendue "UNKNOWN-CTE".

4.4 Extensions à la commande APPEND

La commande APPEND est étendue pour permettre au client d'ajouter des données contenant des octets NUL en utilisant la

syntaxe <literal8>. Le serveur PEUT modifier le CTE des données ajoutées, cependant une telle transformation NE DOIT PAS résulter en une perte de données.

Si la boîte aux lettres de destination ne prend pas en charge la mémorisation de contenu binaire, le serveur DOIT faire échouer la demande et produire une réponse "NO" qui contient le code de réponse étendue "UNKNOWN-CTE".

5. En-têtes codés MIME

La [RFC2047] définit un codage qui permet du texte non US-ASCII dans les en-têtes de message. Ce codage n'est pas le même que le codage de transfert de contenu appliqué aux corps de message, et les transformations de décodage décrites dans le présent mémoire ne s'appliquent pas au texte d'en-tête codé de la [RFC2047]. Un serveur NE DOIT PAS effectuer de conversion de texte d'en-tête codé selon la [RFC2047] en réponse à une demande FETCH ou APPEND binaire.

6. Considérations de mise en œuvre

Les clients et serveurs de messagerie ont été notoirement laxistes dans leur adhésion à la convention CRLF de l'Internet pour terminer les lignes de données textuelles dans les protocoles de l'Internet. Lors de l'envoi de données utilisant l'extension Binary, les serveurs DOIVENT s'assurer que les sections textuelles en mode ligne seront toujours transmises en utilisant la syntaxe de terminaison de ligne de IMAP4 par un CRLF, sans considération de la représentation de mémorisation sous-jacente des données sur le serveur.

Un serveur peut choisir de mémoriser le contenu binaire de corps de message dans un format non codé. Sans considération de la représentation de mémorisation interne utilisée, le serveur DOIT produire des réponses BODYSTRUCTURE qui décrivent le message même si les sections codées en binaire sont codées dans un CTE acceptable pour la spécification IMAP4 de base. De plus, le résultat d'un FETCH BODY DOIT retourner le contenu du corps de message dans le format décrit par la réponse FETCH BODYSTRUCTURE correspondante.

Bien qu'il soit permis au serveur de modifier le CTE des données <literal8> de APPEND, cela ne devrait être fait que lorsque c'est absolument nécessaire. Des changements gratuits de codage vont rendre inutiles la plupart des opérations cryptographiques qui ont été effectuées sur le message.

Cette extension fournit une optimisation qui est utile dans certaines situations spécifiques. Elle n'évite pas aux clients de fournir les fonctionnalités de base (décodage de transfert de contenu) qui devraient être disponibles dans tous les clients de messagerie. Les clients qui prennent en charge cette extension DEVRAIENT être prêts à effectuer leurs propres opérations de décodage de CTE.

7. Syntaxe formelle du protocole

La spécification de syntaxe qui suit utilise la notation en forme Backus-Naur augmenté (ABNF) comme utilisée dans la [RFC2234], et incorpore par référence le cœur des règles définies dans ce document.

Cette syntaxe augmente la grammaire spécifiée dans la [RFC3501].

append =/ "APPEND" SP mailbox [SP flag-list] [SP date-time] SP literal8

fetch-att =/ "BINARY" ["PEEK"] section-binary [partial] / "BINARY.SIZE" section-binary

literal8 = "~{" number "}" CRLF *OCTET ; <number> représente le nombre d'octets dans la chaîne de réponse.

msg-att-static =/ "BINARY" section-binary SP (nstring / literal8) / "BINARY.SIZE" section-binary SP number

partial = "<" number "." nz-number ">"

resp-text-code =/ "UNKNOWN-CTE"

section-binary = "[" [section-part] "]"

8. Références normatives

- [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 [2184](#), [2231](#), [5335](#).*)
- [RFC2047] K. Moore, "MIME ([Extensions de messagerie Internet](#) multi-objets) Partie trois : extensions d'en-tête de message pour texte non ASCII", novembre 1996. (*MàJ par [RFC2184](#), [RFC2231](#)*) (*D.S.*)
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.
- [RFC2234] D. Crocker et P. Overell, "[BNF augmenté pour les spécifications de syntaxe](#) : ABNF", novembre 1997. (*Obsolète, voir [RFC5234](#), STD068*)
- [RFC3501] M. Crispin, "Protocole d'[accès au message Internet - version 4rev1](#)", mars 2003. (*MàJ par [RFC4466](#), [RFC4469](#), [RFC4551](#), [RFC5032](#), [RFC5182](#)*) (*P.S.*)

9. Considérations sur la sécurité

La présente extension ne pose pas de problème de sécurité supplémentaire connu au delà de ceux décrits dans le protocole de base par la [RFC3501].

10. Notice de propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourrait être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur répertoire en ligne des IPR de l'IETF à <http://www.ietf.org/ipr>.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf-ipr@ietf.org.

11. Adresse de l'auteur

Lyndon Nerenberg
Orthanc Systems
1606 - 10770 Winterburn Road
Edmonton, Alberta
Canada T5S 1T6

mél : lyndon@orthanc.ab.ca

12. Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2003). Tous droits réservés.

Le présent document et ses traductions peuvent être copiés et fournis aux tiers, et les travaux dérivés qui les commentent ou les expliquent ou aident à leur mise en œuvre peuvent être préparés, copiés, publiés et distribués, en tout ou partie, sans restriction d'aucune sorte, pourvu que la déclaration de droits de reproduction ci-dessus et le présent paragraphe soient inclus dans toutes telles copies et travaux dérivés. Cependant, le présent document lui-même ne peut être modifié d'aucune façon, en particulier

en retirant la notice de droits de reproduction ou les références à la Internet Society ou aux autres organisations Internet, excepté autant qu'il est nécessaire pour le besoin du développement des normes Internet, auquel cas les procédures de droits de reproduction définies dans les procédures des normes Internet doivent être suivies, ou pour les besoins de la traduction dans d'autres langues que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Internet Society ou ses successeurs ou ayant droits.

Le présent document et les informations y contenues sont fournies sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.