

Groupe de travail Réseau
Request for Comments : 2369
Catégorie : En cours de normalisation
Traduction : Claude Brière de L'Isle

G. Neufeld, Nisto
J. Baer, SkyWeyr Technologies
juillet 1998

Utilisation des URL comme méta syntaxe pour les commandes centrales de liste de messagerie et leur transport à travers les champs d'en-tête de message

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.

Notice de Copyright

Copyright (C) The Internet Society (1998). Tous droits réservés.

Résumé

Les champs d'en-tête de spécification de commande de liste de diffusion sont un ensemble de champs structurés à ajouter aux messages électroniques envoyés par des listes de distribution de messages électroniques. Chaque champ contient normalement un URL (habituellement mailto [RFC2368]) qui localise les informations pertinentes ou qui effectue directement la commande. Les trois principaux champs d'en-tête décrits dans ce document sont List-Help, List-Subscribe, et List-Unsubscribe. Il y a trois autres champs d'en-tête qui sont décrits ici qui, bien qu'ils ne soient pas aussi largement applicables, auront leur utilité pour un nombre suffisant de listes de diffusion pour justifier leur formalisation dans ce document. Ce sont List-Post, List-Owner et List-Archive.

En incluant ces champs d'en-tête, les serveurs de listes rendront possible que les clients de messagerie fournissent un outil automatique aux usagers pour effectuer les fonctions de listes. Cela pourrait prendre la forme d'un élément de menu, d'un bouton à cliquer, ou autre élément d'interface d'utilisateur. L'intention est de simplifier la tâche de l'utilisateur en lui fournissant une interface courante avec les commandes souvent sibyllines et diverses de gestionnaire de liste de messagerie.

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans la RFC2119.

1. Introduction

Le présent document propose d'ajouter des champs d'en-tête supplémentaires aux messages électroniques envoyés par des listes de distribution de messages. Le contenu de chaque nouveau champ est normalement un URL - généralement mailto [RFC2368] - qui localise les informations pertinentes ou effectue directement la commande. Les agents de transport de messagerie (MTA, *Mail Transport Agent*) qui génèrent les champs d'en-tête DEVRAIENT normalement inclure une commande fondée sur mailto, en plus de tous autres protocoles utilisés, afin de prendre en charge les usagers qui n'ont pas accès à des protocoles non fondés sur la messagerie.

La mise en œuvre de ces champs sera facultative. Cependant, des gains de facilité et de fonctionnalités significatifs peuvent être obtenus de leur inclusion. De nombreux gestionnaires de listes, en particulier lorsque la proposition aura gagné un peu de terrain, PEUVENT choisir de mettre en œuvre seulement un ou deux de ces champs. Le champ List-Help est le champ individuel le plus utile car il fournit un point d'accès aux informations détaillées de prise en charge de l'utilisateur, et s'accommode de presque tous les ensembles existants de commandes de gestionnaires de liste. Les champs List-Subscribe et List-Unsubscribe sont aussi très utiles, mais ne peuvent pas décrire certaines syntaxes de gestionnaires de liste pour l'instant (celles qui exigent des substitutions de variables). Voir une explication à l'appendice A.5.

La description de la syntaxe de commande fournie par les champs peut être utilisée par les applications de client de messagerie pour procurer un accès d'utilisateur simplifié et cohérent aux fonctions de liste de distribution de messagerie électronique. Cela pourrait prendre la forme d'éléments de menu, de boutons à cliquer, ou d'autres éléments d'interface d'utilisateur. L'intention est de simplifier l'expérience de l'utilisateur, en fournissant une interface commune pour les commandes de gestionnaire de liste de messagerie qui sont souvent sibyllines et diverses.

Les considérations prises en compte ont été d'éviter de créer trop de champs, en évitant en même temps de surcharger les

champs individuels, tout en gardant une syntaxe claire et simple.

L'utilisation de ces champs ne supprime pas l'exigence de la prise en charge de l'adresse de commande -Request pour les listes de diffusion de la [RFC2142].

2. Syntaxe de commande

Les champs d'en-tête de liste sont soumis aux restrictions de codage et de caractères des en-têtes de messagerie électronique décrites dans la [RFC0822]. De plus, le contenu d'URL a les restrictions supplémentaires du jeu de caractères sûrs d'URL de la [RFC1738].

Le contenu des champs d'en-tête de liste consiste principalement en URL compris entre des crochets angulaires ('<', '>') les espaces blanches internes étant ignorées. Les MTA NE DOIVENT PAS insérer d'espaces entre les crochets, mais les applications clientes devraient traiter toute espace qui aurait pu être insérée par des MTA au mauvais comportement, comme des caractères à ignorer.

Une liste d'URL multiples, de remplacement, PEUT être spécifiée par une liste séparée par des points d'URL inclus entre des crochets angulaires. Les URL ont un ordre de préférence de gauche à droite. L'application cliente devrait utiliser le protocole le plus à gauche qu'elle prend en charge, ou dont elle sait comment y accéder par une application distincte. Par ce mécanisme, des protocoles comme http peuvent être spécifiés tout en fournissant encore la prise en charge basique de mailto pour les clients qui n'ont pas accès aux protocoles qui ne sont pas de messagerie. Le client devrait n'utiliser qu'un des URL disponibles pour une commande, n'en utilisant un des autres que si le premier utilisé échoue.

L'utilisation des URL permet celle de la syntaxe avec les URL existants qui prennent en charge les applications. Comme la norme pour les URL est étendue, les champs d'en-tête de liste vont tirer parti de ces extensions. De plus, l'utilisation des URL donne accès à plusieurs protocoles de transport (tels que ftp et http) bien qu'on s'attende à ce que le protocole "mailto" de la [RFC2368] concentre la plupart des utilisations de champs d'en-tête de listes. L'utilisation de protocoles non mailto devrait être considérée sous le point de vue des usagers qui n'ont pas accès au mécanisme spécifié (ceux qui n'ont qu'une messagerie électronique – sans accès à la Toile).

Les syntaxes de commande qui exigent des champs variables établis par le client (comme d'inclure l'adresse de messagerie de l'utilisateur au sein d'une commande) ne sont pas acceptées par cette mise en œuvre. Cependant, les systèmes qui utilisent de telles syntaxes DEVRAIENT quand même tirer parti du champ List-Help pour fournir à l'utilisateur des instructions détaillées en tant que de besoin, ou – peut être plus utilement – donner accès à certaines formes d'interface de commande structurée comme une forme fondée sur HTML.

Les complications supplémentaires de la prise en charge de champs variables au sein de la syntaxe de commande ont été déterminées comme trop difficiles à prendre en charge par le présent protocole et compromettraient la vraisemblance d'une mise en œuvre par les auteurs de logiciels.

Pour permettre de futures extensions, les applications client DOIVENT respecter les lignes directrices suivantes pour le traitement du contenu des champs d'en-tête décrits dans le présent document :

- 1) Sauf lorsque noté pour des champs spécifiques, si le contenu du champ (suivant toute espace blanche en tête, incluant les commentaires) commence par tout caractère autre que le crochet angulaire d'ouverture "<", le champ DEVRAIT être ignoré.
- 2) Tout caractère qui suit un URL inclus entre des crochets angulaires DEVRAIT être ignoré, sauf si une virgule est le premier caractère qui n'est pas une espace blanche ou un commentaire après le crochet angulaire de fermeture.
- 3) Si un sous-élément (élément séparé par une virgule) au sein du champ n'est pas un URL inclus entre crochets angulaires, le reste du champ (les sous-éléments actuels et tous les suivants) DEVRAIT être ignoré.

3. Champs d'en-tête de liste

Le présent document décrit les champs d'en-tête qui vont fournir la description de la syntaxe de commande pour les fonctions "cœur" et les fonctions secondaires clés de la plupart des listes de diffusion de messagerie. Les champs mis en œuvre sur une liste donnée DEVRAIENT être inclus dans tous les messages distribués par la liste (y compris les réponses de commandes aux utilisateurs individuels) et dans les autres messages qui s'appliquent clairement à une liste distincte. Il DOIT n'y avoir pas plus d'un de chaque champ présent dans tout message.

Ces champs DOIVENT être générés seulement par les listes de diffusion, pas par les utilisateurs finaux.

3.1 List-Help

Le champ List-Help est le plus important des champs d'en-tête décrits dans ce document. Il serait acceptable qu'un gestionnaire de liste n'inclue que ce champ, car par définition, il DEVRAIT conduire l'utilisateur à compléter les instructions pour toutes les autres commandes. Normalement, l'URL spécifié devrait demander le fichier d'aide, incorporant peut-être un formulaire HTML pour la liste des commandes, pour la liste, et autrement fournir l'accès à un site d'instruction de la Toile.

Exemples :

List-Help: <mailto:list@host.com?subject=help> (Instructions pour la liste)

List-Help: <mailto:list-manager@host.com?body=info>

List-Help: <mailto:list-info@host.com> (Informations sur la liste)

List-Help: <http://www.host.com/list/>, <mailto:list-info@host.com>

List-Help: <ftp://ftp.host.com/list.txt> (FTP), <mailto:list@host.com?subject=help>

3.2 List-Unsubscribe

Le champ List-Unsubscribe décrit la commande (de préférence en utilisant la messagerie) pour désinscrire directement l'utilisateur (en le retirant de la liste).

Exemples :

List-Unsubscribe: <mailto:list@host.com?subject=unsubscribe>

List-Unsubscribe: (Utiliser cette commande pour quitter la liste) <mailto:list-manager@host.com?body=unsubscribe%20list>

List-Unsubscribe: <mailto:list-off@host.com>

List-Unsubscribe: <http://www.host.com/list.cgi?cmd=unsub&lst=list>, <mailto:list-request@host.com?subject=unsubscribe>

3.3 List-Subscribe

Le champ List-Subscribe décrit la commande (de préférence en utilisant la messagerie) pour abonner directement l'utilisateur (demande d'ajout à la liste).

Exemples :

List-Subscribe: <mailto:list@host.com?subject=subscribe>

List-Subscribe: <mailto:list-request@host.com?subject=subscribe>

List-Subscribe: (Utiliser cette commande pour se joindre à la liste) <mailto:list-manager@host.com?body=subscribe%20list>

List-Subscribe: <mailto:list-on@host.com>

List-Subscribe: <http://www.host.com/list.cgi?cmd=sub&lst=list>, <mailto:list-manager@host.com?body=subscribe%20list>

3.4 List-Post

Le champ List-Post décrit la méthode pour envoyer à la liste. C'est normalement l'adresse de la liste, mais cela PEUT être un modérateur, ou éventuellement une autre forme de soumission. Pour le cas particulier d'une liste qui ne permet pas l'envoi (par exemple, une liste d'annonces) le champ List-Post peut contenir la valeur particulière "NO".

Exemples :

List-Post: <mailto:list@host.com>

List-Post: <mailto:moderator@host.com> (Les envois sont soumis à un modérateur)

List-Post: <mailto:moderator@host.com?subject=list%20posting>

List-Post: NO (les envois ne sont pas permis sur cette liste)

3.5 List-Owner

Le champ List-Owner identifie le chemin pour contacter un administrateur humain pour la liste. L'URL PEUT contenir l'adresse d'un administrateur de la liste, l'administrateur du système de messagerie, ou toute autre personne qui peut jouer le rôle de contact pour la liste. Il n'est pas besoin de spécifier List-Owner si c'est la même personne que l'administrateur du système de messagerie (le "postmaster").

Exemples :

List-Owner: <mailto:listmom@host.com> (Contacter personne pour de l'aide)

List-Owner: <mailto:grant@foo.bar> (Grant Neufeld)

List-Owner: <mailto:josh@foo.bar?Subject=list>

3.6 List-Archive

Le champ List-Archive décrit comment accéder aux archives de la liste.

Exemples :

List-Archive: <mailto:archive@host.com?subject=index%20list>

List-Archive: <ftp://ftp.host.com/pub/list/archive/>

List-Archive: <http://www.host.com/list/archive/> (Archive de la Toile)

4. Prise en charge des listes incorporées

Une liste qui est une sous liste d'une autre liste dans une hiérarchie de listes de diffusion incorporées va devoir modifier certains des champs d'en-tête de la List-, tout en en laissant d'autres comme la liste parente les a établis.

Les sous-listes DEVRAIENT retirer les champs List-Help, List-Subscribe, List-Unsubscribe et List-Owner de la liste parente, et DEVRAIENT insérer leur propre version de ces champs.

Si la sous-liste fournit ses propres archives, elle DEVRAIT remplacer le champ List-Archive par le sien propre. Autrement, elle DOIT laisser inchangé le champ List-Archive.

Selon la façon dont sont traités les envois à la liste, la sous-liste PEUT remplacer le champ List-Post. S'il est approprié ou non de remplacer List-Post est laissé à la détermination des gestionnaires de liste individuels. L'intention est que les envois soient distribués à tous les membres de la liste principale, List-Post ne devrait pas être changé par une sous-liste qui ferait que les envois ne seraient distribués qu'aux membres de la sous-liste.

5. Considérations pour la sécurité

Très peu de nouveaux problèmes de sécurité sont générés par la présente proposition. Les en-têtes de message sont un standard existant, conçu pour s'accommoder facilement de nouveaux types. Il peut y avoir des problèmes avec plusieurs champs insérés ou des en-têtes falsifiés, mais ce sont des problèmes inhérents à la messagerie Internet qui ne sont pas spécifiques du protocole décrit dans ce document. De plus, leurs implications sont relativement bénignes.

Les processeurs de listes de diffusion ne devraient pas permettre qu'un champ d'en-tête de liste généré par un usager passe dans leurs listes, de crainte qu'ils soient une source de confusion pour l'utilisateur et aient un potentiel de création de problèmes de sécurité.

Du côté du client, il peut y avoir des problèmes avec des envois ou des commandes envoyés par erreur. Il est nécessaire que l'usager ait une chance de confirmer toute action avant qu'elle soit exécutée. Dans le cas de mailto, il peut être approprié de créer le message correctement formaté sans l'envoyer, ce qui permet à l'usager de voir exactement ce qui arrive et lui donne l'opportunité d'approuver ou d'éliminer le message avant son envoi.

Toutes les considérations de sécurité pour l'utilisation des URL [RFC1738] s'appliquent également à ce protocole. Les applications de client de messagerie ne devraient pas prendre en charge les URL de champ d'en-tête de liste qui pourraient compromettre la sécurité du système de l'utilisateur. Ceci inclut le type d'URL "file://" qui peut éventuellement être utilisé pour déclencher l'exécution d'une application locale sur certains systèmes d'utilisateur.

6. Remerciements

Les nombreux participants aux listes de diffusion List-Header [5], ListMom-Talk [6], List-Managers et MIDA-Mail ont beaucoup contribué à la formation et à la structure de ce document.

Keith Moore <moore@cs.utk.edu> et Christopher Allen <ChristopherA@consensus.com> ont fourni des indications sur le processus de normalisation.

Annexe A. Discussion sur le fond

Cette proposition résulte de discussions commencées sur la liste de discussion ListMom-Talk [6]. Lorsque la discussion a atteint un niveau suffisant, une liste distincte a été formée pour discuter plus à fond la présente proposition, la liste Headers Mail List [5]. Nous avons inclus des résumés des questions clés soulevées, afin de montrer certaines des solutions de remplacement examinées et les raisons de nos décisions.

A.1 Champs d'en-tête multiples contre champ d'en-tête unique

L'utilisation d'un seul champ d'en-tête pour porter la méta-syntaxe de commande a été rejetée pour un certain nombre de raisons.

Un tel champ exigerait la création d'une nouvelle méta-syntaxe afin de décrire les commandes de liste (par opposition à l'utilisation de la syntaxe largement utilisée des URL qui a été choisie pour cette mise en œuvre). Toute couche de complexité et de nouveauté supplémentaire réduit la probabilité d'une mise en œuvre réelle parce qu'elle va exiger des travaux supplémentaires pour la prendre en charge. De plus, en utilisant la syntaxe d'URL existante, on peut profiter de l'expérience qu'a l'utilisateur final de cette syntaxe et de sa capacité à l'utiliser même si son application client ne prend pas en charge les champs d'en-tête de listes.

Restreindre le transport de la méta-syntaxe à l'utilisation d'un seul champ d'en-tête introduit aussi des complications avec les limitations de taille de champ d'en-tête. La plupart des commandes individuelles peuvent aisément être décrites sur une seule ligne, mais décrire une multitude de commandes peut prendre de nombreuses lignes dans le champ et court un plus grand risque d'être modifié par un serveur existant sur le chemin.

La mise en œuvre cliente est aussi plus facile avec plusieurs champs, car chaque commande peut être prise en charge et mise en œuvre individuellement, complètement indépendamment des autres. Donc, certains gestionnaires de liste ou client de messagerie peuvent choisir de mettre en œuvre un sous-ensemble des champs sur la base des besoins spécifiques de leurs listes individuelles.

Finalement, le format décrit dans ce document est simple et bien reconnu, ce qui réduit les chances d'erreurs dans la mise en œuvre et l'analyse.

A.2 URL contre listes de paramètres

Les URL ont déjà une syntaxe établie qui est souple, bien définie, et d'usage largement répandu. Alors que sa définition mûrit et s'étend, les capacités des champs de listes vont croître aussi, sans exiger de modification de cette proposition. Les URL sont bien préparés à traiter les protocoles et développements futurs, et peuvent facilement décrire les différents protocoles d'accès existants tels que mailto, http et ftp.

De nombreux clients ont déjà les fonctionnalités permettant de reconnaître, analyser et évaluer les URL, soit en interne, soit en passant la demande à une application d'aide. Cela rend la mise en œuvre plus facile et plus réaliste. Par exemple, cette faculté existante d'analyse des URL nous permet d'ajouter des prototypes de fonctions d'en-tête de liste aux clients de messagerie existants (Eudora et EMailer pour Macintosh) sans modifier leur code source.

A.3 Pourquoi ne pas simplement créer un langage standard de commande ?

Un langage de commande standard, pris en charge par tous les services de listes de la messagerie électronique, réduirait largement les problèmes de l'accès à la liste qui perturbent actuellement les services existants. Il réduirait la quantité d'informations à acquérir nécessaires aux utilisateurs finaux et permettrait de développer un certain nombre d'outils communs de prise en charge.

Cependant, une telle normalisation pose des problèmes dans les domaines de la prise en charge du multilinguisme et par rapport au besoin coutumier de listes de diffusion individuelles. Le développement d'une telle norme ferait vraisemblablement l'objet d'un lent taux d'adoption de la part des développeurs de logiciels et des fournisseurs de services de listes.

Cela n'empêche pas le développement d'une telle norme (en fait, cela suggère qu'on devrait commencer au plus tôt) mais nous avons besoin d'une solution qui puisse être largement acceptée par les services de liste actuels.

La plupart des syntaxes de commandes de gestionnaire de liste existantes peuvent être prises en charge sans langage de commande standard. En utilisant les URL, on permet des méthodes d'accès de remplacement comme le contrôle fondé sur la Toile que ne permettrait probablement pas un langage de commande standard.

Finalement, le soutien du client en faveur d'un langage de commande standard n'est pas du tout aussi clair ou nécessairement simple à mettre en œuvre. La diversité et le grand nombre de commandes existant aujourd'hui exigerait des interfaces d'utilisateur compliquées qui pourraient être difficiles à mettre en œuvre et générer de la confusion. En restreignant cette proposition aux fonctions principales, la mise en œuvre de client est beaucoup plus simple, ce qui augmente de façon significative la probabilité de mise en œuvre (on en veut pour preuve le soutien déjà annoncé par un certain nombre d'auteurs d'applications de client et de serveur).

A.4 Internationalisation

La prise en charge du multilinguisme relève de la norme sur les URL. Si les URL le prennent en charge, les champs d'en-tête List- le font aussi. C'est un autre avantage de l'utilisation des URL comme blocs de construction des champs d'en-tête de liste.

A.5 Substitution de variable

Des variables permettraient aux champs d'en-tête List- de s'accommoder de presque tous les gestionnaires de liste existants. Cependant, cela accroîtrait de façon incommensurable la complexité de la proposition toute entière, et impliquerait éventuellement de redéfinir l'URL standard, ou nous forcerait à utiliser quelque chose de plus compliqué (et donc plus difficile à mettre en œuvre) que les URL pour décrire la syntaxe de commande.

Les paramètres pourraient soit devoir être obligatoires (c'est-à-dire que l'agent d'utilisateur ne soumet pas le message si il ne sait pas quel texte substituer) soit on a besoin d'un moyen pour dire "si vous connaissez ce paramètre, ajoutez le texte ici ; autrement, faites "ceci" où "ceci" est soit : (a) substituer une chaîne constante, soit (b) un échec.

La raison pour laquelle vous voulez une facilité comme celle-là est que certaines applications de serveur de liste insistent pour avoir certains paramètres comme les noms des usagers, que l'agent d'utilisateur peut ou non connaître ; par exemple, des serveurs de liste insistent pour avoir un prénom et un nom de famille si vous fournissez l'un des deux.

Ce qui pourrait conduire à quelque chose comme la syntaxe en coquille de UNIX, dans laquelle $\${foo-bar}$ signifie de substituer la valeur du paramètre "foo" si "foo" est défini, et autrement de substituer la chaîne "bar". Peut-être $\$foo$ pourrait signifier "substituer la valeur du paramètre foo si il est défini, autrement substituer la chaîne vide".

Tout cela semble bien trop compliqué pour les gains espérés, en particulier dans la mesure où l'utilisation de variables peut souvent être évitée.

L'utilisation de variables dans la syntaxe de commande des services de liste apparaît en déclin et ne s'applique en aucun cas à toutes les commandes. Bien que les champs d'en-tête des commandes "unsubscribe" et "subscribe" puissent n'être pas utilisables par les systèmes qui exigent l'utilisation de variables, le champ "help" va toujours fournir aux utilisateurs finaux un point d'accès cohérent par lequel ils peuvent obtenir du soutien pour l'utilisation de la liste.

A.6 Pourquoi ne pas utiliser une partie MIME spécialisée au lieu des champs d'en-tête ?

Les parties MIME ont été prises en compte, mais comme la plupart des clients de messagerie électronique ne prennent pas en charge MIME ou ne sont pas équipés pour traiter de telles parties spécialisées – une telle mise en œuvre poserait des problèmes à l'utilisateur final. Il est aussi moins facile à de nombreux serveurs de liste de mettre en œuvre MIME que de nouveaux champs d'en-tête.

Cependant, on cherche à concevoir une partie MIME qui décrive plus complètement la syntaxe de commande de liste, tout autant qu'on essaye de trouver le moyen de la faire prendre en charge par le logiciel applicable.

A.7 Pourquoi inclure une commande Subscribe ?

"Subscribe" et "Unsubscribe" sont les commandes clés nécessaires par presque toute liste. Les autres commandes, telles que le mode résumé, ne sont pas aussi largement prises en charge.

De plus, les usagers qui se sont désabonnés (avant de partir en vacances, ou pour toute autre raison) peuvent vouloir se réabonner à une liste. Ou, un message peut être transmis/retransmis d'un abonné à un non abonné. Ou, l'utilisateur peut changer d'adresse et vouloir s'abonner à partir de sa nouvelle adresse. La disponibilité du champ List-Subscribe peut certainement aider dans tous ces cas.

A.8 Dangers de l'inflation des en-têtes

À quel point y a-t-il trop de champs d'en-tête ? Cela dépend de la liste. Sur certaines, la majorité des usagers ne va jamais être consciente de la présence d'un champ jusqu'à ce que le logiciel client fournisse pour lui une interface d'utilisateur de remplacement (autrement dit au champ Reply-To). Sur d'autres, les usagers vont souvent voir les champs d'en-tête des messages et seront capables de reconnaître la fonction des URL qui y sont contenus.

La souplesse apportée par le protocole décrit dans le présent document (en ce que les champs d'en-tête peuvent être mis en œuvre individuellement autant qu'approprié) donne aux administrateurs de liste suffisamment de marge de manœuvre pour satisfaire leurs besoins individuels.

Annexe B. Mise en œuvre client

B.1 Lignes directrices

Pour les commandes "mailto" fondées sur un URL, les applications client de messagerie électronique peuvent choisir de fournir des retours d'information spécialisés (tels que de présenter un dialogue ou une alerte) au lieu du message de commande réel, demandant une confirmation de commande à l'utilisateur. Le retour d'information devrait identifier la destination du message et la commande avec une explication plus descriptive. Par exemple :

"Voulez-vous envoyer la commande de désabonnement "unsubscribe somelist" à 'somelist-request@some.host.com' ? L'envoi de cette commande vous retirera de la liste en question."

Si l'utilisateur a plusieurs adresses de messagerie électronique prises en charge par le client de messagerie, l'application client devrait inviter l'utilisateur sur l'adresse à utiliser lors de l'abonnement ou effectuer quelque autre action où l'adresse à utiliser ne peut pas être spécifiquement déterminée. Lors d'une désinscription ou action similaire, l'adresse de souscription devrait être utilisée, sauf si elle n'est pas connue de l'application et ne peut être déterminée à partir des en-têtes de message.

B.2 Options de mise en œuvre

Les possibilités de mise en œuvre suivantes sont suggérées ici pour donner une idée des raisons pour lesquelles des nouveaux champs d'en-tête seront utiles, et pourquoi ils devraient être pris en charge.

Dans la plupart des cas, il peut être utile de désactiver l'interface des commandes, lorsqu'elles ne sont pas applicables au message sélectionné.

B.2.1 Combinaisons clés et lignes de commandes

Sur les systèmes fondés sur le texte qui utilisent des lignes de commande ou des combinaisons clés, chaque champ pourrait être mis en œuvre comme une commande distincte. Donc, une combinaison va abonner l'utilisateur, une autre va le désabonner, une troisième demander de l'aide, etc. Les commandes ne seraient disponibles que sur les messages qui contiennent les champs d'en-tête de liste.

B.2.2 Éléments de menu

Sur les systèmes graphiques qui ont des menus, ces commandes pourraient prendre la forme d'un menu ou sous-menu d'éléments. Par exemple, un menu "Listes" pourrait apparaître à la lecture des messages qui contiennent les champs d'en-

tête, avec des éléments nommés "Abonnement", "Désabonnement", "Obtenir de l'aide", "Envoyer un message à la liste", "Contacter le propriétaire de la liste" et "Accéder aux archives de la liste". Ce menu pourrait être désactivé lorsque il n'est pas applicable au message en cours ou disparaître entièrement.

B.2.3 Boutons poussoirs et palettes

Sur les systèmes à fenêtre graphique, des boutons pourraient être placés dans la fenêtre du message, une barre d'outils, ou une palette flottante. Chaque bouton pourrait correspondre à une commande, avec les noms "Abonnement", "Désabonnement", "Aide", "Envoi à la liste", "Propriétaire de la liste" et "Archive". Ces boutons ou palettes pourraient être désactivés lorsque non applicables au message en cours ou disparaître entièrement.

B.2.4 Retours d'informations pour l'utilisateur

Si on utilise une interface de dialogue (ou un autre élément de retour d'information) l'application client DOIT inclure une option pour que l'utilisateur revoie (et éventuellement modifie) le message avant de l'envoyer. L'application peut aussi trouver utile de fournir un lien avec une assistance sensible au contexte plus détaillée sur l'accès aux listes de messagerie électronique en général.

Références

- [RFC0822] D. Crocker, "Norme pour le [format des messages](#) de texte de l'ARPA-Internet", STD 11, août 1982.
- [RFC1738] T. Berners-Lee et autres, "[Localisateurs uniformes](#) de ressource (URL)", décembre 1994. (*P.S., voir les RFC4248 et 4266*)
- [RFC2142] D. Crocker, "Noms de [boîtes aux lettres](#) pour les services, rôles et fonctions communs", mai 1997. (*P.S.*)
- [RFC2368] P. Hoffman, L. Masinter, J. Zawinski, "Le schéma d'URL mailto", juillet 1998. (*P.S.*) (*Obsolète, voir la RFC6068*)
- [5] Liste de diffusion "List-Header" à list-header@list.nisto.com <URL:<http://www.nisto.com/listspec/mail/>>
<URL:<http://www.nisto.com/listspec/>>
- [6] Liste de diffusion "ListMom-Talk" à listmom-talk@skyweyr.com <URL:<http://cgi.skyweyr.com/ListMom.Home>>

Adresse des éditeurs

Joshua D. Baer
Box 273
4902 Forbes Avenue
Pittsburgh, PA 15213-3799
USA
mél : josh@skyweyr.com

Grant Neufeld
Calgary, Alberta
Canada
mél : grant@acm.org
Web : <http://www.nisto.com/>

Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (1998). 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 droits de reproduction ci-dessus et ce paragraphe soient 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 de 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 ou 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.