Groupe de travail Réseau
Request for Comments: 5537
RFC rendue obsolète : 1036
Catégorie : Sur la voie de la normalisation

R. Allbery, éd., Stanford University
C. Lindsey
novembre 2009
Traduction Claude Brière de L'Isle

Architecture et protocoles Netnews

Résumé
Le présent document définit l'architecture des systèmes Netnews et spécifie la manipulation et l'interprétation correcte des articles Netnews par les logiciels qui les génèrent, distribuent, mémorisent, et affichent. Il spécifie aussi les exigences qui doivent être satisfaites par tout protocole utilisé pour transporter et servir les articles Netnews.

Statut de ce mémoire
Le présent document spécifie un protocole sur la voie de la normalisation de l'Internet 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.

Table des matières

1. Introduction

1.1 Concepts de base

"Netnews" est un ensemble de protocoles pour générer, mémoriser, et restituer des "articles" de nouvelles dont le format est défini dans la [RFC5536], et pour les échanger parmi un lectorat qui est potentiellement largement distribué. Il est organisé autour de "newsgroups" *(groupes de nouvelles)* avec l'idée que chaque lecteur va être capable de voir tous les articles postés à chaque newsgroup auquel il participe. Ces protocoles utilisent couramment un algorithme d'arrosage qui propage les copies à travers un réseau de serveurs participants. Normalement, une seule copie est mémorisée par serveur, et chaque serveur la rend disponible à la demande aux lecteurs capable d'accéder à ce serveur.

"Usenet" est un réseau particulier publiquement accessible dans le monde entier fondé sur les protocoles Netnews. Il y a seulement un tel réseau possible ; il y a des déploiements des protocoles Netnews autres que Usenet (comme ceux internes à des organisations particulières). Le présent document discute de l'architecture et protocoles Netnews les plus généraux.

1.2 Domaine d'application

Le présent document définit l'architecture des systèmes Netnews et spécifie la manipulation et l'interprétation correcte des articles Netnews par les logiciels qui les génèrent, distribuent, mémorisent, et affichent. Il traite des problèmes de protocole qui sont indépendants des protocoles de transport comme le protocole de transfert des nouvelles du réseau (NNTP, *Network News Transfer Protocol*) [RFC3977], et spécifie les exigences que Netnews fait peser sur ces protocoles de transport sous-jacents. Il spécifie aussi le traitement des messages de contrôle.

Le format et la syntaxe des articles Netnews sont spécifiés dans la [RFC5536], qui devrait être lue en conjonction avec le présent document.

1.3 Notation des exigences

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119],.

1.4 Notation de la syntaxe

La syntaxe définie dans le présent document utilise la notation de forme Backus-Naur augmenté (ABNF) (incluant les règles centrales) définie dans la [RFC5234] et les constructions définies dans les [RFC5536] et [RFC5322].

Les règles d'ABNF définies ailleurs et utilisées dans le présent document sont :

CRLF = <voir la [RFC5234] Appendice B.1>
DIGIT = <voir la [RFC5234] Appendice B.1>
HTAB = <voir la [RFC5234] Appendice B.1>
SP = <voir la [RFC5234] Appendice B.1>
WSP = <voir la [RFC5234] Appendice B.1>
VCHAR = <voir la [RFC5234] Appendice B.1>

argument = <voir la [RFC5536] paragraphe 3.2.3>
article-locator = <voir la [RFC5536] paragraphe 3.2.14>
component = <voir la [RFC5536] paragraphe 3.1.4>

control-command = <voir la [RFC5536] paragraphe 3.2.3>
diag-keyword = <voir la [RFC5536] paragraphe 3.1.5>
diag-match = <voir la [RFC5536] paragraphe 3.1.5>
diag-other = <voir la [RFC5536] paragraphe 3.1.5>
dist-name = <voir la [RFC5536] paragraphe 3.2.4>
msg-id = <voir la [RFC5536] paragraphe 3.1.3>
newsgroup-name = <voir la [RFC5536] paragraphe 3.1.4>
path-diagnostic = <voir la [RFC5536] paragraphe 3.1.5>
path-identity = <voir la [RFC5536] paragraphe 3.1.5>
path-nodot = <voir la [RFC5536] paragraphe 3.1.5>
tail-entry = <voir la [RFC5536] paragraphe 3.1.5>
verb = <voir la [RFC5536] paragraphe 3.2.3>

display-name = <voir la [RFC5322] paragraphe 3.4>
local-part = <voir la [RFC5322] paragraphe 3.4.1>
mailbox = <voir la [RFC5322] paragraphe 3.4>

1.5 Définitions

Tout terme utilisé dans le présent document qui est défini au paragraphe 1.5 de la [RFC5536] est utilisé avec la définition qui y est donnée. De plus, les termes suivants vont être utilisés :

Une "hiérarchie" est l'ensemble de tous les newsgroups dont le nom partage un premier <component> (comme défini au paragraphe 3.1.4 de la [RFC5536]). Une "sous- hiérarchie" est l'ensemble de tous les newsgroups dont le nom partage plusieurs composants initiaux.

Un "serveur de nouvelles" est distingué par les rôles de "agent injecteur", "agent relayeur", et "agent servant". Un "agent injecteur" accepte un proto-article dans le but de le distribuer aux agents relayeurs et servants et donc aux lecteurs. Un "agent relayeur" accepte les articles provenant d'autres agents relayeurs ou agents injecteurs et les distribue aux autres agents relayeurs ou servants. Un "agent servant" reçoit un article d'un agent relayeur ou injecteur et le rend disponible aux lecteurs.

Un "agent d'utilisateur" est distingué par les rôles de "agent posteur" et "agent lecteur". Un "agent posteur" est un logiciel qui aide à la préparation d'un proto-article et le passe ensuite à un agent injecteur. Un "agent lecteur" est un logiciel qui restitue les articles depuis un agent servant pour la présentation à un lecteur.

"Injecter" un article est le traitement d'un proto-article par un agent injecteur. Normalement, cette action est faite une fois et seulement une fois pour un article donné. Une "injection multiple" est le passage du même article à plusieurs agents injecteurs, soit à la suite, soit en parallèle, par un ou plusieurs agents posteurs.

Une "passerelle" est un logiciel qui reçoit les articles de nouvelles et les convertit en messages d'une autre sorte (comme des messages électroniques de la [RFC5322]) reçoit des messages d'une autre sorte et les convertit en articles de nouvelles, ou porte des articles entre deux réseaux Netnews séparés.

2. Transport

Le moyen exact utilisé pour transmettre les articles d'un agent à un autre n'est pas spécifié. NNTP [RFC3977] est le mécanisme de transport le plus courant pour les réseaux Netnews. D'autres méthodes utilisées incluent le protocole de copie d'Unix à Unix (UUCP, *Unix-to-Unix Copy Protocol*] (largement utilisé dans les premiers jours de Usenet) et des supports magnétiques et optiques livrés physiquement. Tout mécanisme peut être utilisé en conjonction avec le présent protocole pourvu qu'il puisse satisfaire les exigences qui y sont spécifiées.

Les transports pour les articles Netnews DOIVENT traiter les articles de nouvelles comme des séquences d'octets non interprétées, excluant les valeurs %d00 (qui ne peuvent pas se produire dans les articles Netnews), %d13, et %d10 (qui DOIVENT seulement apparaître dans les articles Netnews comme une paire dans cet ordre et qui, ensemble, notent un séparateur de ligne). Ces octets sont les caractères US-ASCII [ASCII] NUL, CR, et LF, respectivement.

Note : cela correspond à la gamme des octets permis dans les données MIME 8bit [RFC2045]. Les transports pour Netnews ne sont pas obligés de prendre en charge la transmission des données MIME "binary".

En particulier, les transports DOIVENT porter tous les champs d'en-tête non modifiés (y compris les champs d'en-tête dans les objets message/rfc822 dans des corps d'article) même si ils contiennent des octets dans la gamme de 128 à 255. De plus, les transports pour les agents relayeurs et servants DOIVENT, et les transports pour les autres agents DEVRAIENT, porter les lignes même si elles excèdent 998 caractères, en particulier dans les corps d'article. (Cette exigence est plus stricte que celle des données MIME 8bit.) Ces exigences incluent les chemins de transport entre les agents posteurs, les agents injecteurs, les agents servants, et les agents lecteurs.

3. Devoirs des agents

Cette section spécifie les devoirs des agents impliqués dans la création, le relais, et le service des articles Netnews. Ce protocole est décrit en suivant la vie d'un article Usenet normal : il est préparé par un agent posteur, donné à un agent injecteur, transféré à travers un ou plusieurs agents relayeurs, accepté par un agent servant, et finalement récupéré par un agent lecteur. Les articles soumis à des groupes avec modération passent par un processus supplémentaire, qui est décrit séparément (voir le paragraphe 3.5.1 et l'étape 7 du paragraphe 3.5). Finalement, les devoirs et exigences supplémentaires d'une passerelle sont discutés.

À chaque étape, chaque agent est obligé d'effectuer un ensemble de vérifications et transformations de l'article. Elles sont décrites comme des séquences d'étapes à suivre, mais on devrait comprendre que c'est l'effet de ces séquences qui est important, et les mises en œuvre peuvent utiliser toute méthode qui produit les mêmes effets.

De nombreux serveurs de nouvelles combinent les fonctions d'agent injecteur, agent relayeur, et agent servant dans un seul paquetage logiciel. Pour les besoins de la présente spécification, de tels agents combinés devraient être traités conceptuellement comme un agent injecteur qui envoie des articles à un agent servant et, facultativement, à un agent relayeur. Les exigences des trois agents DOIVENT quand même être satisfaites quand le serveur de nouvelles effectue les fonctions de ces agents.

Sur les serveurs de nouvelles qui les acceptent, des messages de contrôle peuvent avoir des effets supplémentaires à ceux décrits ci-dessous. Ces effets sont décrits à la Section 5.

3.1 Principes généraux

Il y a deux principes importants que les mises en œuvre et administrateurs de nouvelles doivent garder en tête. Le premier est le principe bien connu de robustesse de l'Internet : être libéral dans ce qu'on accepte, et conservateur pour ce qu'on envoie.

Appliqué à Netnews, cela signifie principalement que les articles indésirables ou non conformes DEVRAIENT être rejetés aussitôt que possible, mais une fois qu'ils sont dans la circulation générale, les agents relayeurs et servants peuvent souhaiter les accepter lorsque possible plutôt que perdre l'information. Les agents posteurs et les agents injecteurs DEVRAIENT donc être maximalement stricts dans leur application de ce protocole et de la [RFC5536], et les agents lecteurs DEVRAIENT être robustes en présence de violations du format d'article Netnews lorsque possible.

Dans le cas de Netnews, il y a un principe encore plus important, dérivé d'un code de conduite beaucoup plus ancien, le serment d'Hippocrate (on peut donc appeler cela le principe d'Hippocrate) : D'abord, ne pas causer de dommages.

Il est vital de réaliser que des décisions qui pourraient être simplement sous optimales dans un plus petit contexte peuvent devenir des fautes dévastatrices quand elles sont amplifiées par les actions de milliers d'hôtes en quelques minutes.

Aucun agent Netnews n'est jamais obligé d'accepter un article. Il est courant pour les agents injecteurs, relayeurs, et les agents servants de rejeter des articles bien formés pour des raisons de politique locale (comme de ne pas souhaiter porter un newsgroup particulier ou tenter de filtrer les articles non désirés). Le présent document spécifie comment les articles sont traités si ils sont acceptés et spécifie des cas où ils doivent être rejetés, mais un agent PEUT toujours rejeter tout article pour d'autres raisons que celles mentionnées ici.

Un des buts principaux du protocole Netnews est de s'assurer que tous les lecteurs qui reçoivent un article particulier (identifié de façon univoque par le contenu de son champ d'en-tête Message-ID) voient un article identique, à part des divergences admissibles dans les en-têtes de trace et les métadonnées locales. En conséquence, les agents (autres que les

modérateurs) NE DOIVENT PAS modifier les articles d'autre façon que décrite ici. Les articles inacceptables DOIVENT être rejetés plutôt que corrigés.

3.2 Champ d'en-tête Path

Tous les composants de serveur de nouvelles (agents injecteurs, agents relayeurs, et agents servants) DOIVENT s'identifier quand ils traitent un article en ajoutant leur <path-identity> (définie au paragraphe 3.1.5 de la [RFC5536]) devant le champ d'en-tête Path. Les agents injecteurs DOIVENT aussi utiliser la même identité dans les champs d'en-tête Injection-Info qu'ils ajoutent, et les agents servants et relayeurs DEVRAIENT utiliser la même identité dans tout champ d'en-tête Xref qu'ils ajoutent.

La <path-identity> utilisée par un agent peut être choisie via une des méthodes suivantes (en ordre de préférence décroissant) :

1. Le nom de domaine pleinement qualifié (FQDN) du système où l'agent fonctionne.

2. Un nom de domaine pleinement qualifié (FQDN) au sein d'un domaine affilié aux administrateurs de l'agent et dont l'unicité est garantie par les administrateurs de ce domaine. Par exemple, l'unicité de "serveur.exemple.org" pourrait être garantie par l'administrateur de "exemple.org" même si il n'y a pas d'enregistrement du DNS pour serveur.exemple.org lui-même.

3. Un autre nom (arbitraire) de la forme d'un <path-nodot>, estimé être unique et enregistré au moins auprès de tous les autres serveurs de nouvelles auxquels cet agent relayeur ou agent injecteur envoie des articles. Cette option NE DEVRAIT PAS être utilisée sauf si les options antérieures sont indisponibles ou sauf si l'usage du nom est pour une longue durée.

Certaines mises en œuvre existantes traitent <path-identity> comme sensible à la casse, d'autres non. La <path-identity> DEVRAIT donc être toute en minuscules et les mises en œuvre DEVRAIENT comparer les identités de façon insensible à la casse.

3.2.1 Construction du champ d'en-tête Path

Si un agent relayeur ou servant reçoit un article d'un agent injecteur ou servant qui fait partie du même serveur de nouvelles, il PEUT laisser inchangé le champ d'en-tête Path de l'article. Autrement, chaque agent injecteur, relayeur, ou servant qui accepte un article DOIT mettre à jour le champ d'en-tête Path comme suit. Noter que le contenu du champ d'en-tête Path est construit de droite à gauche en ajoutant des éléments devant.

1. L'agent DOIT faire précéder de "!" le contenu du champ d'en-tête Path.

2. Un agent injecteur DEVRAIT faire précéder le contenu du champ d'en-tête Path du <path-diagnostic> "!.POSTED", facultativement suivi par "." et le FQDN ou adresse IP de la source,.

3. Un agent relayeur ou servant DEVRAIT faire précéder d'un <path-diagnostic> le contenu du champ d'en-tête Path, où le <path-diagnostic> est choisi comme suit :
   * Si la <path-identity> attendue de la source de l'article correspond à la <path-identity> la plus à gauche du contenu du champ d'en-tête Path, utiliser "!" (<diag-match>), résultant en deux "!" consécutifs.
   * Si la <path-identity> attendue de la source de l'article ne correspond pas, utiliser "!.MISMATCH." suivi par la <path-identity> attendue de la source ou son adresse IP.
   * Si l'agent relayeur ou servant ne veut pas ou n'est pas capable de vérifier la <path-identity>, utiliser "!.SEEN." suivi par le FQDN, l'adresse IP, ou la <path-identity> attendue de la source.
     La "<path-identity> attendue de la source de l'article" est une <path-identity> pour l'agent injecteur ou relayeur qui a passé l'article à cet agent relayeur ou servant, déterminée par les propriétés de la connexion via laquelle l'article a été reçu (par exemple, une identité d'authentification ou l'adresse IP de l'homologue). Il faut savoir que la [RFC1036] ne comportait pas de <path-diagnostic>. Les mises en œuvre qui précèdent la présente spécification vont ajouter seulement un caractère "!" entre les chaînes de <path-identity>.

4. L'agent PEUT alors faire précéder le contenu du champ d'en-tête Path de "!" ou "!!" suivi par une <path-identity> supplémentaire pour elle-même autre que sa principale. Utiliser "!!", et par là ajouter un <diag-match> puisque la <path-identity> est clairement vérifiée, est RECOMMANDÉ. Cette étape peut être répétée un nombre quelconque de fois. C'est permis pour les agents qui ont plusieurs <path-identity> (comme durant une transition de l'un à l'autre).

Chacune de ces <path-identity> DOIT satisfaire les exigences établies au paragraphe 3.2.

5. Finalement, l'agent DOIT faire précéder de sa <path-identity> principale le contenu du champ d'en-tête Path. La <path-identity> principale est la <path-identity> qu'il annonce normalement à ses homologues pour qu'ils génèrent les <path-diagnostic> comme décrit ci-dessus.

Tout agent qui modifie le champ d'en-tête Path PEUT le faire sauter à la ligne en insérant une espace blanche de pliage (FWS, *folding white space*) immédiatement après toute <path-identity> ou <diag-other> qu'il a ajouté (voir le paragraphe 3.1.5 de la [RFC5536] pour les localisations acceptables pour les FWS).

3.2.2 Exemple de champ d'en-tête Path

Voici un exemple d'un champ d'en-tête Path créé en suivant les règles pour les agents injecteurs et relayeurs.

```
Path: foo.fai.exemple!.SEEN.fai.exemple!foo-news
  !.MISMATCH.2001:DB8:0:0:8:800:200C:417A!bar.fai.exemple
  !!old.site.exemple!barbaz!!baz.fai.exemple
  !.POSTED.dialup123.baz.fai.exemple!not-for-mail
```

Cet article a été injecté par baz.fai.exemple comme indiqué par le <diag-keyword> "POSTED". L'injecteur a enregistré qu'il a reçu l'article provenant de dialup123.baz.fai.exemple. "not-for-mail" est une <tail-entry> courante.

L'article a été relayé à l'agent relayeur connu, au moins à old.site.exemple, comme "barbaz". Cet agent relayeur a confirmé à sa satisfaction que "baz.fai.exemple" était une <path-identity> attendue pour la source de l'article et a donc utilisé <diag-match> ("!") pour son <path-diagnostic>.

barbaz l'a relayé à old.site.exemple, qui ne prend pas en charge <diag-keyword> et donc a utilisé le vieux délimiteur "!". Cela indique que l'identité de "barbaz" n'a pas été vérifiée et peut avoir été falsifiée.

old.site.exemple l'a relayé à un serveur de nouvelles en utilisant la <path-identity> de bar.fai.exemple et en prétendant (en utilisant le <path-diagnostic> "!") avoir vérifié qu'il venait de old.site.exemple.

bar.isp.exemple l'a relayé à foo-news, qui, n'étant pas convaincu qu'il vient vraiment de bar.fai.exemple, a inséré le <diag-keyword> "MISMATCH" et a ensuite déclaré qu'il a reçu l'article de l'adresse IPv6 [2001:DB8:0:0:8:800:200C:417A]. (Ce n'est pas pour dire que bar.fai.exemple n'était pas une <path-identity> correcte pour cette source mais simplement que l'identité ne correspond pas aux attentes de foo-news.)

foo-news a alors passé l'article à foo.fai.exemple, qui a refusé de valider sa <path-identity> et a à la place ajouté le <diag-keyword> "SEEN" pour indiquer qu'il connaît la source de l'article comme étant fai.exemple. Cela peut être soit une <path-identity> attendue, soit le FQDN du système d'où il a reçu l'article. Probablement, foo.fai.exemple est un agent servant qui a ensuite livré l'article à un agent lecteur.

baz.fai.exemple, bar.fai.exemple, et foo-news ont replié le champ d'en-tête Path.

3.3 Historique d'article et suppression de dupliqués

Netnews utilise normalement un algorithme d'arrosage-remplissage pour la propagation des articles dans lequel chaque serveur de nouvelles offre les articles qu'il accepté à plusieurs homologues, et chaque serveur de nouvelles peut se voir offrir le même article provenant de plusieurs autres serveurs de nouvelles. Par conséquent la suppression des dupliqués est capitale ; si un serveur de nouvelles acceptait tous les article qui lui sont offerts, il pourrait sans nécessité accepter (et ensuite potentiellement retransmettre) des douzaines de copies de chaque article.

Les agents relayeurs et servants DOIVENT donc garder un enregistrement des articles qu'ils ont déjà vu et utiliser cet enregistrement pour rejeter les offres supplémentaires du même article. Cet enregistrement est appelé le fichier ou base de données "historique".

Chaque article est identifié de façon univoque par son identifiant de message, de sorte qu'un agent relayeur ou servant pourrait satisfaire cette exigence en mémorisant un enregistrement de chaque identifiant de message que cet agent a vu. Une telle base de données d'historique pourrait cependant croître sans limite donc il est courant et permis d'optimiser sur la base du champ d'en-tête Injection-Date ou Date d'un article comme suit. (Dans la discussion qui suit, la "date" d'un article

est définie comme étant la date représentée par son champ d'en-tête Injection-Date, si il est présent ; autrement, par son champ d'en-tête Date.)

- Les agents PEUVENT choisir un intervalle de coupure et rejeter tout article avec une date postérieure dans le passé à cet intervalle de coupure. Si cet intervalle est plus court que le temps que prend un article pour se propager à travers le réseau, l'agent pourrait rejeter un article qu'il n'a pas encore vu, donc il ne devrait pas être agressivement court pour Usenet, par exemple, un intervalle de coupure de pas moins de sept jours est conventionnel.

- Les agents qui appliquent une telle coupure PEUVENT alors éliminer de leurs bases de données d'historique les enregistrements d'articles qui ont des dates plus anciennes que la coupure. Si un tel article est offert à nouveau à l'agent, il va être rejeté à cause de la date de coupure, donc l'enregistrement d'historique n'est plus exigé pour supprimer les dupliqués.

- Autrement, les agents PEUVENT éliminer les enregistrements d'historique selon la date à laquelle l'article a été vu pour la première fois par cet agent plutôt que la date de l'article. Dans ce cas, l'intervalle de rétention d'historique DOIT être plus long d'au moins 24 heures que l'intervalle de coupure pour permettre les articles datés dans le futur. Cet intervalle correspond à l'erreur admise sur la date de l'article (voir le paragraphe 3.5).

Il y a juste deux stratégies de mise en œuvre pour l'historique d'article, tout au moins les plus courantes. Les agents relayeurs et servants ne sont pas obligés d'utiliser ces stratégies, seulement de satisfaire l'exigence de ne pas accepter un article plus d'une fois. Cependant, ces stratégies sont sûres et largement déployées, et les utilisateurs sont encouragés à utiliser l'une d'elles, en particulier si ils n'ont pas une grande expérience de Netnews et des effets subtils de son algorithme d'arrosage-remplissage.

3.4 Devoirs d'un agent envoyeur

Un agent posteur est le composant d'un agent d'utilisateur qui aide un posteur à créer un proto-article valide et à le transmettre à un agent injecteur.

Les agents posteurs DEVRAIENT s'assurer que les proto-articles qu'ils créent sont valides selon la [RFC5536] et toutes autres politiques applicables. Ils NE DOIVENT PAS créer de champ d'en-tête Injection-Info ; ce champ d'en-tête peut seulement être ajouté par l'agent injecteur.

Si le proto-article contient déjà les deux champs d'en-tête Message-ID et Date, les agents posteurs PEUVENT ajouter un champ d'en-tête Injection-Date à ce proto-article immédiatement avant de passer ce proto-article à un agent injecteur. Ils DEVRAIENT faire ainsi si le champ d'en-tête Date (représentant l'heure de composition du proto-article) est à plus d'un jour dans le passé de la date d'injection. Ils DOIVENT faire ainsi si le proto-article est soumis à plus d'un agent injecteur ; voir le paragraphe 3.4.2.

Le champ d'en-tête Injection-Date est nouveau dans cette révision du protocole Netnews et est conçu pour permettre au champ d'en-tête Date de contenir la date de composition (comme recommandé au paragraphe 3.6.1 de la [RFC5322]) même si le proto-article n'est pas à injecter avant un certain temps après sa composition. Cependant, on note que toutes les mises en œuvre précédant la présente spécification ignorent le champ d'en-tête Injection-Date et utilisent à la place le champ d'en-tête Date pour rejeter les articles plus anciens que leur intervalle de coupure (voir le paragraphe 3.3) et les agents injecteurs qui précèdent la présente spécification n'ajoutent pas d'en-tête Injection-Date. Les articles avec un champ d'en-tête Date substantiellement dans le passé vont encore être rejetés par les mises en œuvre qui précèdent cette spécification, sans considération du champ d'en-tête Injection-Date, et donc peuvent souffrir d'une plus pauvre propagation.

Contrairement à la [RFC5322], qui implique que la ou les boîtes aux lettres dans le champ d'en-tête From devraient être celles du ou des posteurs, un posteur qui ne souhaite pas, pour une raison quelconque, utiliser sa propre boîte aux lettres PEUT utiliser toute boîte aux lettres se terminant dans le domaine de niveau supérieur ".invalid" [RFC2606].

Les agents posteurs destinés à être utilisés par des posteurs ordinaires DEVRAIENT rejeter toute tentative de poster un article qui annule ou se substitue (via le champ d'en-tête Supersedes) à un autre article dont le posteur n'est pas l'auteur ou l'envoyeur.

3.4.1 Proto-articles

Un proto-article est un article dans le format utilisé par un agent posteur quand il offre cet article à un agent injecteur. Il peut omettre certains champs d'en-tête qui peuvent être mieux fournis par l'agent injecteur et ne va pas contenir les champs

d'en-tête qui sont à ajouter par l'agent injecteur. Un proto-article est seulement pour transmission à un agent injecteur et NE DEVRAIT PAS être transmis à un autre agent.

Un proto-article a le même format qu'un article normal excepté que les champs d'en-tête Injection-Info et Xref NE DOIVENT PAS être présents, que le champ d'en-tête Path NE DEVRAIT PAS contenir un <diag-keyword> "POSTED", et que tous les champs d'en-tête obligatoires suivants PEUVENT être omis : Message-ID, Date, et Path. À tous les autres égards, un proto-article DOIT être un article Netnews valide. En particulier, les champs d'en-tête qui peuvent être omis NE DOIVENT PAS être présents avec un contenu invalide.

Si un agent posteur a l'intention d'offrir le même proto-article à plusieurs agents injecteurs, les champs d'en-tête Message-ID, Date, et Injection-Date DOIVENT être présents et identiques dans toutes les copies du proto-article. Voir le paragraphe 3.4.2.

3.4.2 Injections multiples d'articles

Dans certaines circonstances (par exemple, quand on poste à plusieurs réseaux, supposés disjoints, quand on utilise des agents injecteurs avec une connexité épisodique, ou quand on désire une redondance supplémentaire) un agent posteur peut souhaiter offrir le même article à plusieurs agents injecteurs. Dans ce cas inhabituel, le but n'est pas de créer plusieurs articles indépendants mais plutôt d'injecter le même article à plusieurs points et de laisser la facilité normale de suppression de dupliqués de Netnews (voir le paragraphe 3.3) s'assurer que tout agent accepte l'article seulement une fois, même si des réseaux supposés disjoints ont des liaisons inattendues.

Chaque fois que possible, une injection multiple DEVRAIT être faite en offrant le même proto-article à plusieurs agents injecteurs. L'agent posteur DOIT fournit les champs d'en-tête Message-ID, Date, et Injection-Date, et le proto-article offert à chaque agent injecteur DOIT être identique.

Dans certains cas, offrir le même proto-article à tous les agents injecteurs peut n'être pas possible (comme quand il y a une passerelle, après l'injection, les articles trouvés sur un réseau Netnews pour un autre supposé non connecté). Dans ce cas, l'agent posteur DOIT supprimer tout champ d'en-tête Xref et renommer ou supprimer tous les champs d'en-tête Injection-Info, Path, et autres champs d'en-tête de trace avant de le passer à un autre agent injecteur. (Cela reconvertit l'article en un proto-article.) Il DOIT conserver inchangés les champs d'en-tête Message-ID, Date, et Injection-Date. Il NE DOIT PAS ajouter de champ d'en-tête Injection-Date si il manque dans l'article existant.

Note : des injections multiples risquent naturellement de dupliquer les articles. De multiples injections après l'injection, en reconvertissant un article en un proto-article et en l'injectant à nouveau, risquent de plus des boucles, la perte des informations de trace, la répétition involontaire de l'injection dans les mêmes réseaux, et autres problèmes. Elle devrait être faite avec soin et seulement quand il n'y a pas d'autre solution. L'exigence de conserver les champs d'en-tête Message-ID, Date, et Injection-Date minimise la possibilité de boucle et assure que l'article nouvellement injecté n'est pas traité comme un nouvel article séparé.

Une injection multiple d'un article qui énumère un ou plusieurs newsgroups modérés dans son champ d'en-tête Newsgroups DEVRAIT seulement être faite par un modérateur et DOIT seulement être faite après que le proto-article a été approuvé pour tous les groupes modérés auquel il va être posté et après qu'un champ d'en-tête Approved a été ajouté (voir le paragraphe 3.9). L'injection multiple d'un article non approuvé destiné à des newsgroups modérés va normalement seulement résulter en ce que le modérateur reçoive plusieurs copies, et si l'état du newsgroup n'est pas cohérent à travers tous les agents injecteurs, il peut en résulter la duplication de l'article ou d'autres problèmes.

3.4.3 Suiveurs

Un suiveur *(followup)* est un article qui contient une réponse au contenu d'un article antérieur, son précurseur. En plus de ses devoirs normaux, un agent posteur qui prépare un suiveur est aussi soumis aux exigences suivantes. Partout dans ce qui suit, il est déclaré que, par défaut, un champ d'en-tête est dit être hérité d'un de ces champs d'en-tête dans le précurseur, cela signifie que son contenu initial est à copier du contenu de ce champ d'en-tête précurseur (avec des changements de saut à la ligne permis). Cependant, les posteurs PEUVENT alors outrepasser ce comportement par défaut avant le postage.

En dépit de la pratique historique de certains agents posteurs, le champ d'en-tête Keywords NE DEVRAIT PAS être hérité par défaut de l'article précurseur.

1. Si le champ d'en-tête Followup-To de l'article précurseur consiste en "poster", le suiveur NE DOIT PAS être posté par

défaut mais, par défaut, doit être envoyé comme message électronique à l'adresse données dans le champ d'en-tête Reply-To ou From du précurseur, suivant les règles pour une réponse de messagerie électronique [RFC5322]. Cette action PEUT être outrepassée par le posteur, auquel cas l'agent posteur devrait continuer comme si le champ d'en-tête Followup-To dans le précurseur n'existait pas.

2. Le champ d'en-tête Newsgroups DEVRAIT, par défaut, être hérité du champ d'en-tête Followup-To du précurseur si il est présent ; autrement, il est hérité du champ d'en-tête Newsgroups du précurseur.

3. Le champ d'en-tête Subject DEVRAIT par défaut être hérité de celui du précurseur. La chaîne sensible à la casse "Re: " (incluant l'espace après les deux-points) PEUT être ajoutée devant le contenu de son champ d'en-tête Subject sauf si il commence déjà par cette chaîne.

Note : L'ajout de "Re: " ne sert aucune fonction du protocole et n'est donc pas exigé, mais il est largement attendu et ne pas le faire serait surprenant.

4. Le champ d'en-tête Distribution DEVRAIT, par défaut, être hérité du champ d'en-tête Distribution du précurseur, si il est présent.

5. Le suiveur DOIT avoir un champ d'en-tête References se référant à son précurseur, construit en accord avec le paragraphe 3.4.4.

3.4.4 Construction du champ d'en-tête References

La procédure suivante est à utiliser chaque fois qu'un article précédent (le "parent") est à référencer dans le champ d'en-tête References d'un nouvel article, soit parce que le nouvel article est suiveur et le parent est son précurseur, soit pour une autre raison.

Le contenu du champ d'en-tête References du nouvel article DOIT être formé à partir du contenu du champ d'en-tête References du parent si il est présent, suivi par le contenu du champ d'en-tête Message-ID du parent. Si le parent avait un en-tête References, un FWS comme défini dans la [RFC5536] DOIT être ajouté entre son contenu et le contenu du champ d'en-tête Message-ID.

Si le champ d'en-tête References résultant devrait, après dépliage, excéder 998 caractères (incluant son nom de champ mais pas le CRLF final) il DOIT être ajusté (et autrement PEUT être ajusté). Ajuster signifie de supprimer un certain nombre d'identifiants de message de son contenu, sauf le premier identifiant de message et les deux derniers qui NE DOIVENT PAS être supprimés.

Une propriété essentielle du champ d'en-tête References, garantie par la procédure ci-dessus et qu'il est EXIGÉ qu'elle soit conservée par toute extension au présent protocole, est qu'un article NE DOIT PAS précéder un de ses parents.

3.5 Devoirs d'un agent injecteur

Un agent injecteur prend un proto-article provenant d'un agent posteur et soit le transmet à un modérateur, soit le passe à un ou des agents relayeurs ou servants. Un agent injecteur a la responsabilité principale de s'assurer que tout article qu'il injecte se conforme aux règles des normes Netnews. L'administrateur d'un agent injecteur est aussi supposé avoir une certaine responsabilité à l'égard du reste du réseau Netnews auquel il est connecté pour les articles que l'agent injecteur accepte.

Les agents injecteurs, quand ils rejettent des articles, sont invités à communiquer la raison du rejet à l'agent posteur en utilisant toute facilité fournie par le transport sous-jacent. L'agent injecteur est dans une position unique pour communiquer la raison du rejet ; les agents relayeurs et les agents servants ont normalement à rejeter les messages en silence. L'agent injecteur porte donc beaucoup de la charge du diagnostic des agents posteurs en dérangement ou de communiquer les violations de politique aux posteurs.

Un agent injecteur DOIT avoir une liste disponible (éventuellement vide) de groupes modérés pour lesquels il accepte les articles, et les adresses de soumission correspondantes. Il DEVRAIT avoir disponible une liste de newsgroups valides pour collecter les articles non postés à un newsgroup valide et donc qui vont probablement être éliminés en silence par les agents relayeurs et servants. Généralement, un agent injecteur est déployé en conjonction avec un agent servant et tient ces listes sur la base des messages de contrôle reçus par l'agent servant.

Un agent injecteur traite les proto-articles comme suit :

1. Il DEVRAIT vérifier que l'article provient d'une source de confiance (par exemple, en s'appuyant sur la capacité d'autorisation du transport sous-jacent utilisé pour parler à l'agent posteur).

2. Il DOIT rejeter tout proto-article qui n'a pas les champs d'en-tête obligatoires appropriés pour un proto-article, qui a des champs d'en-tête Injection-Info ou Xref, qui a un champ d'en-tête Path contenant le <diag-keyword> "POSTED", ou qui n'est pas syntaxiquement valide comme défini par la [RFC5536]. Il DEVRAIT rejeter tout proto-article qui contient un champ d'en-tête déconseillé pour Netnews (voir, par exemple, la [RFC3798]). Il PEUT rejeter tout proto-article qui contient des champs d'en-tête de trace (par exemple, NNTP-Posting-Host) indiquant qu'il a déjà été injecté par un agent injecteur qui n'a pas ajouté de Injection-Info ou Injection-Date.

3. Il DEVRAIT rejeter tout article dont le champ d'en-tête Injection-Date ou Date est plus de 24 heures à l'avenir (et PEUT utiliser une marge de moins de 24 heures). Il DEVRAIT rejeter tout article dont le champ d'en-tête Injection-Date est trop loin dans le passé (plus vieux que l'intervalle de coupure d'un agent relayeur que l'agent injecteur utilise, par exemple). Il DEVRAIT de même rejeter tout article dont le champ d'en-tête Date est trop loin dans le passé, car tous les serveurs de nouvelles ne prennent pas en charge Injection-Date et seul l'agent injecteur peut fournir un message d'erreur utile à l'agent posteur. Dans tous les cas, cet intervalle NE DEVRAIT PAS être plus court que 72 heures dans le passé.

4. Il DEVRAIT rejeter tout proto-article dont le champ d'en-tête Newsgroups ne contient pas au moins un <newsgroup-name> pour un groupe valide, ou qui contient un <newsgroup-name> réservé pour des objets spécifiques par le paragraphe 3.1.4 de la [RFC5536] sauf si cet objet spécifique ou un accord local s'applique au proto-article traité. Le crosspostage à des newsgroups inconnus n'est pas interdit pourvu qu'au moins un des newsgroups dans l'en-tête Newsgroups soit valide.

5. Les champs d'en-tête Message-ID et Date avec les contenus appropriés DOIVENT être ajoutés quand ils ne sont pas présents dans le proto-article.

6. L'agent injecteur NE DOIT altérer en aucune façon le corps de l'article (incluant de changer le Content-Transfer-Encoding). Il PEUT ajouter d'autres champs d'en-tête non déjà fournis par le posteur, mais les agents injecteurs sont invités à utiliser l'en-tête Injection-Info pour de telles informations et à minimiser l'ajout d'autres en-têtes. Il NE DEVRAIT PAS altérer, supprimer, ou réordonner de champ d'en-tête existant sauf le champ d'en-tête Path. Il NE DOIT PAS altérer ou supprimer de champ d'en-tête Message-ID existant.

7. Si l'en-tête Newsgroups contient un ou plusieurs groupes modérés et si le proto-article ne contient pas un champ d'en-tête Approved, l'agent injecteur DOIT soit le transmettre à un modérateur comme spécifié au paragraphe 3.5.1, soit, si ce n'est pas possible, le rejeter. Cette transmission DOIT être faite après l'ajout des en-têtes Message-ID et Date si nécessaire, et avant d'ajouter les en-têtes Injection-Info et Injection-Date.

8. Autrement, un champ d'en-tête Path avec une <tail-entry> DOIT être ajouté si il n'est pas déjà présent.

9. L'agent injecteur DOIT ensuite mettre à jour le champ d'en-tête Path comme décrit au paragraphe 3.2.1.

10. Un champ d'en-tête Injection-Info DEVRAIT être ajouté pour identifier la source de l'article et éventuellement d'autres informations de trace comme décrit au paragraphe 3.2.8 de la [RFC5536].

11. Si le proto-article a déjà un champ d'en-tête Injection-Date, il NE DOIT PAS être modifié ni remplacé. Si le proto-article a les deux champs d'en-tête Message-ID et Date, un champ d'en-tête Injection-Date NE DOIT PAS être ajouté, car le proto-article peut avoir été injecté plusieurs fois par un agent posteur qui précède cette norme. Autrement, l'agent injecteur DOIT ajouter un champ d'en-tête Injection-Date contenant la date et l'heure courante.

12. Finalement, l'agent injecteur transmet l'article à un ou plusieurs agents relayeurs, et le processus d'injection est terminé.

3.5.1 Transmission de messages à un modérateur

Un agent injecteur DOIT transmettre le proto-article au modérateur du groupe modéré le plus à gauche dans la liste du champ d'en-tête Newsgroups, généralement par message électronique. Il y a deux façons standard pour faire cela :

1. Le proto-article complet est encapsulé, champs d'en-tête et tout le reste, dans le message. Ce DEVRAIT être fait en créant un message électronique avec un Content-Type de application/news-transmission avec le paramètre d'usage réglé

à "moderate". Le corps NE DEVRAIT PAS contenir d'autre contenu que le message. Cette méthode a l'avantage de supprimer tout conflit possible entre les champs d'en-tête Netnews et de messagerie électronique et tout changement à ces champs durant le transport à travers la messagerie électronique.

2. Le proto-article est envoyé comme un message électronique avec l'ajout de tous les champs d'en-tête requis pour un message électronique comme défini dans la [RFC5322], et éventuellement avec l'ajout des autres champs d'en-tête conventionnels dans la messagerie électronique, comme To et Received. Le champ d'en-tête existant Message-ID DEVRAIT être conservé.

Bien que ces deux méthodes aient été utilisées dans le passé et que la première ait de clairs avantages techniques, la seconde est d'usage plus courant et de nombreux modérateurs ne sont pas prêts à traiter les messages dans le premier format. En conséquence, la première méthode NE DEVRAIT PAS être utilisée sauf si le modérateur auquel l'article est transmis est connu pour être capable de traiter cette méthode.

Note : Déduire l'adresse de messagerie du modérateur d'un groupe sort du domaine d'application de ce document. On mentionnera, cependant, qu'une méthode courante est d'utiliser un service de transmission qui traite les soumissions pour de nombreux groupes modérés. Pour une compatibilité maximum avec les serveurs de nouvelles existants, ces services de transmission forment généralement l'adresse de soumission pour un groupe modéré en remplaçant chaque "." dans le <newsgroup-name> par "-" et en utilisant ensuite cette valeur comme <local-part> d'une <mailbox> formée en ajoutant un domaine d'ensemble.

La transmission des proto-articles aux modérateurs via la messagerie électronique est la méthode la plus générale et la plus courante dans les grands réseaux Netnews comme Usenet, mais tout moyen de transmettre l'article qui le préserve sans l'injecter PEUT être utilisé. Par exemple, si l'agent injecteur a accès à une base de données utilisée par le modérateur pour mémoriser les proto-articles attendant le traitement, il peut placer le proto-article directement dans cette base de données. De telles méthodes peuvent être plus appropriées pour de plus petits réseaux Netnews.

3.6 Devoirs d'un agent relayeur

Un agent relayeur accepte d'injecter des articles provenant d'agents injecteurs et d'autres agents relayeurs et les passe à des agents relayeurs ou servants. Pour éviter d'outrepasser les politiques d'agent injecteur et la falsification des en-têtes Path et Injection-Info, les agents relayeurs DEVRAIENT accepter seulement les articles provenant d'agents de confiance.

Un article NE DEVRAIT PAS être relayé si l'agent envoyeur n'a pas été configuré à fournir, et l'agent receveur à recevoir, au moins un des <newsgroup-name> dans son champ d'en-tête Newsgroups et au moins un des <dist-name> dans son champ d'en-tête Distribution (si il est présent). Exceptionnellement, les messages de contrôle créant ou supprimant des newsgroups (messages de contrôle newgroup ou rmgroup, par exemple) DEVRAIENT être relayés si le groupe affecté apparaît dans son champ d'en-tête Newsgroups et si les deux agents relayeurs envoyeur et receveur sont configurés à relayer un newsgroup de ce nom (qu'un tel newsgroup existe ou non).

Afin d'éviter des tentatives de relais inutiles, un article NE DEVRAIT PAS être relayé si la <path-identity> de l'agent receveur (ou un de ses alias connus) apparaît comme <path-identity> (excluant dans la <tail-entry> ou suivant un <diag-keyword> "POSTED") dans son champ d'en-tête Path.

Un agent relayeur traite un article comme suit :

1. Il DOIT rejeter tout article sans un champ d'en-tête Newsgroups ou Message-ID, ou sans un champ d'en-tête Injection-Date ou Date.

2. Il DOIT examiner le champ d'en-tête Injection-Date ou, si il est absent, le champ d'en-tête Date, et rejeter l'article si cette date est de plus de 24 heures dans le futur. Il PEUT rejeter les articles avec des dates dans le futur avec une marge plus petite que 24 heures.

3. Il DOIT rejeter tout article qui a déjà été accepté. Si il met en œuvre un des mécanismes décrits au paragraphe 3.3, cela signifie que il DOIT rejeter tout article dont la date tombe en dehors de l'intervalle de coupure car il ne va pas savoir si de tels articles ont ou non été acceptés précédemment.

4. Il DEVRAIT rejeter tout article qui n'inclut pas tous les champs d'en-tête obligatoires. Il PEUT rejeter tout article qui contient des champs d'en-tête qui n'ont pas de contenu valide.

5. Il DEVRAIT rejeter tout article qui correspond à un message de contrôle d'annulation déjà-reçu ou le contenu du champ d'en-tête Supersedes d'un article accepté, pourvu que l'agent relayeur ait choisi (sur la base de la politique de site locale) d'honorer ce message de contrôle d'annulation ou ce champ d'en-tête Supersedes.

6. Il PEUT rejeter tout article sans champ d'en-tête Approved posté à un newsgroup connu pour être modéré. Cette pratique est fortement encouragée, mais il n'est pas exigé que les informations nécessaires pour le faire soient conservées par un agent relayeur.

7. Il DOIT mettre à jour le champ d'en-tête Path comme décrit au paragraphe 3.2.1.

8. Il PEUT supprimer tout champ d'en-tête Xref déjà présent. Il PEUT ajouter un nouveau champ d'en-tête Xref pour son propre usage (mais on rappelle que la [RFC5536] permet au plus un tel champ d'en-tête).

9. Finalement, il passe l'article à d'autres agents relayeurs et servants pour lesquels il est configuré à envoyer des articles.

Les agents relayeurs DEVRAIENT, lorsque c'est possible dans le transport sous-jacent, informer l'agent qui a passé l'article à l'agent relayeur si l'article est rejeté. Les agents relayeurs NE DOIVENT PAS informer d'autre entité externe du rejet d'un article sauf si cette entité externe a explicitement demandé à être informée de telles erreurs.

Les agents relayeurs NE DOIVENT PAS altérer, supprimer, ou réarranger de partie d'un article sauf pour les champs d'en-tête Path et Xref. Ils NE DOIVENT en aucune façon modifier le corps des articles. Si un article n'est pas acceptable comme il est, l'article DOIT être rejeté plutôt que modifié.

3.7 Devoirs d'un agent servant

Un agent servant accepte les articles provenant d'un agent relayeur ou d'un agent injecteur, les mémorise, et les rend disponibles aux agents lecteurs. Les articles sont normalement indexés par newsgroup et <article-locator> (paragraphe 3.2.14 de la [RFC5536]) généralement sous la forme d'un nombre décimal.

Si l'agent servant mémorise les articles par newsgroup, les messages de contrôle NE DEVRAIENT PAS être mémorisés dans les newsgroups dont la liste figure dans le champ d'en-tête Newsgroups du message de contrôle. Ils DEVRAIENT plutôt être mémorisés dans un newsgroup dans la hiérarchie "control", qui est réservée pour cela. Conventionnellement, les messages de contrôle sont mémorisés dans les newsgroups nommés pour le type de message de contrôle (comme "control.cancel" pour les messages de contrôle d'annulation).

Un agent servant DOIT avoir disponible une liste (qui peut être vide) de groupes modérés pour lesquels il accepte des articles de sorte qu'il puisse rejeter les articles non approuvés postés à des groupes modérés. Fréquemment, un agent servant est déployé en combinaison avec un agent injecteur et peut utiliser la même liste que l'agent injecteur.

Un agent servant traite les articles comme suit :

1. Il DOIT rejeter tout article qui n'inclut pas tous les champs d'en-tête obligatoires ou tout article qui contient des champs d'en-tête qui n'ont pas un contenu valide.

2. Il DOIT examiner le champ d'en-tête Injection-Date ou, si il est absent, le champ d'en-tête Date, et rejeter l'article si cette date est de plus de 24 heures dans le futur. Il PEUT rejeter les articles avec des dates dans le futur avec une marge plus petite que 24 heures.

3. Il DOIT rejeter tout article qui a déjà été accepté. Si il met en œuvre un des mécanismes décrits au paragraphe 3.3, cela signifie que il DOIT rejeter tout article dont la date tombe en dehors de l'intervalle de coupure car il ne va pas savoir si de tels articles ont ou non été acceptés précédemment.

4. Il DEVRAIT rejeter tout article qui correspond à un message cancel déjà reçu et honoré, ou à un champ d'en-tête Supersedes, suivant les mêmes règles qu'un agent relayeur (paragraphe 3.6).

5. Il DOIT rejeter tout article sans champ d'en-tête Approved posté à tout newsgroup mentionné comme modéré.

6. Il DOIT mettre à jour le champ d'en-tête Path comme décrit au paragraphe 3.2.1.

7. Il DOIT supprimer tout champ d'en-tête Xref de chaque article (sauf quand il est spécialement configuré pour préserver

les <article-locator> établis par le site envoyeur). Il PEUT (et généralement veut) alors ajouter un nouveau champ d'en-tête Xref (mais on rappelle que la [RFC5536] permet au plus un tel champ d'en-tête).

8. Finalement, il mémorise l'article et le rend disponible aux agents lecteurs.

Les agents servants NE DOIVENT PAS créer de nouveaux newsgroups simplement parce que un <newsgroup-name> non reconnu se produit dans un champ d'en-tête Newsgroups. Les Newsgroups sont normalement créés via des messages de contrôle (paragraphe 5.2.1).

Les agents servants NE DOIVENT PAS altérer, supprimer, ou réarranger de partie d'un article, sauf pour les champs d'en-tête Path et Xref. Ils NE DOIVENT en aucune façon modifier le corps des articles. Si un article n'est pas acceptable tel qu'il est, l'article DOIT être rejeté plutôt que modifié.

3.8 Devoirs d'un agent lecteur

Comme un agent lecteur est seulement un participant passif dans un réseau Netnews, il n'y a pas d'exigence spécifique du protocole qui pèse sur lui. Voir dans [USEAGE] les recommandations de bonnes pratiques.

3.9 Devoirs d'un modérateur

Un modérateur reçoit des articles de nouvelles, généralement par messagerie électronique, décide si il les approuve et, dans ce cas, les passe à un agent injecteur ou les transmet à d'autres modérateurs.

Les articles sont normalement reçus par le modérateur dans un message électronique, encapsulé comme un objet de Content-Type application/news-transmission (ou éventuellement encapsulé mais sans un champ d'en-tête Content-Type explicite) ou directement comme un message électronique contenant déjà tous les champs d'en-tête appropriés pour un article Netnews (voir le paragraphe 3.5.1). Les modérateurs qui peuvent recevoir des articles via messagerie électronique DEVRAIENT être prêts à accepter des articles dans l'un ou l'autre format.

Le protocole ne fait pas de restriction sur les critères utilisés pour accepter ou rejeter les messages ou sur les modifications qu'un modérateur peut faire à un message (champs d'en-tête et corps) avant de l'injecter. Les bonnes pratiques recommandées sont cependant de minimiser les changements requis. Les modérateurs doivent être conscients que les modifications faites aux articles peuvent invalider les signatures créées par le posteur ou les modérateurs précédents. Voir dans [USEAGE] plus de recommandations des bonnes pratiques.

Les modérateurs traitent les articles comme suit :

1. Ils décident d'approuver ou rejeter un proto-article et, si ils l'approuvent, préparent le proto-article pour l'injection. Si le proto-article a été reçu comme un message électronique non encapsulé, cela va exiger de le reconvertir en un proto-article Netnews valide. Si l'article est rejeté, il est normalement rejeté pour tous les newsgroups auxquels il était posté et rien de plus n'est fait. Si il est approuvé, le modérateur poursuit avec les étapes suivantes.

2. Si le champ d'en-tête Newsgroups contient plus de newsgroups modérés pour lesquels l'approbation n'a pas été déjà donnée, ils peuvent soit arriver à un accord avec les autres modérateurs sur la disposition de l'article, soit plus généralement ajouter une indication (identifiant le modérateur et le nom du newsgroup) qu'ils ont approuvé l'article et le transmettre ensuite au modérateur du newsgroup non approuvé le plus à gauche. Cette transmission DEVRAIT être faite suivant la procédure du paragraphe 3.5.1. Elle PEUT être faite par une rotation des <newsgroup-name> dans le champ d'en-tête Newsgroups afin que le newsgroup non approuvé le plus à gauche soit le newsgroup modéré le plus à gauche dans ce champ et ensuite de le poster, laissant l'agent injecteur faire la transmission. Cependant, quand on utilise ce mécanisme, il DOIT d'abord s'assurer que l'article ne contient pas de champ d'en-tête Approved.

3. Si le champ d'en-tête Newsgroups ne contient pas d'autre groupe modéré non approuvé, ils ajoutent un champ d'en-tête Approved (voir le paragraphe 3.2.1 de la [RFC5536]) identifiant le modérateur et, autant que possible, tous les autres modérateurs qui ont approuvé l'article. Le modérateur qui effectue ces étapes prend la responsabilité de s'assurer que l'article a été approuvé par les modérateurs de tous les newsgroups modérés auxquels il a été posté.

4. Les modérateurs sont invités à conserver le champ d'en-tête Message-ID sauf si il est invalide ou si l'article a été significativement changé par rapport à sa forme originale. Les modérateurs sont aussi invités à conserver le champ d'en-tête Date sauf si il apparaît être périmé (72 heures ou plus dans le passé) pour des raisons comprises par le modérateur (comme des délais dans le processus de modération) auquel cas ils PEUVENT substituer la date courante. Tout champ

d'en-tête Injection-Date, Injection-Info, ou Xref déjà présent DOIT être supprimé.

5. Tout champ d'en-tête Path DOIT soit être supprimé, soit tronqué pour garder seulement les entrées qui suivent son <diag-keyword> "POSTED", si il en est.

6. Le modérateur passe alors l'article à un agent injecteur, ayant d'abord observé tous les devoirs d'un agent posteur.

3.10 Devoirs d'une passerelle

Une passerelle transforme un article du format du message natif au format d'un autre support, ou traduit les messages d'un autre support en articles de nouvelles, ou transforme des articles en proto-articles pour injection dans un réseau Netnews séparé. L'encapsulation d'un article de nouvelles dans un message de type MIME application/news-transmission, ou la désencapsulation qui la suit, n'est pas du ressort de la passerelle car elle n'implique pas de transformation de l'article.

Il y a deux types de base de passerelle, la passerelle sortante qui transforme un article de nouvelles en un type différent de message, et la passerelle entrante qui transforme un message provenant d'un autre réseau en un proto-article de nouvelles et l'injecte dans un réseau Netnews. Ils sont traités séparément ci-dessous.

La transformation d'un article en un autre support a de très fortes chances d'élimination ou d'interférence avec la protection inhérente au système de nouvelles contre les articles dupliqués. Le problème le plus courant causé par les passerelles est celui des boucles qui réinjectent de façon répétée les articles précédemment postés. Pour empêcher cela, une passerelle DOIT prendre des précautions contre les boucles, comme on le précise ci-dessous.

Les transformations appliquées au message DEVRAIENT être aussi minimes que possible tout en accomplissant quand même la fonction de passerelle. Chaque changement fait par une passerelle casse potentiellement une propriété d'un des supports ou perd des informations, et donc seules les transformations rendues nécessaires par les différences entre les supports devraient être appliquées.

Si une passerelle bidirectionnelle (une passerelle à la fois entrante et sortante) est établie entre Netnews et un autre support, les passerelles entrantes et sortantes DEVRAIENT être coordonnées pour éviter une réinjection involontaire des articles qui y passent. Une passerelle circulaire (passerelle d'un message dans un autre support et ensuite retour dans Netnews) NE DEVRAIT PAS être faite ; l'encapsulation de l'article DEVRAIT être plutôt utilisée lorsque c'est nécessaire.

Une passerelle bidirectionnelle sûre entre une liste de diffusion et un newsgroup est beaucoup plus facile si le newsgroup est modéré. Les postages au groupe modéré et les soumissions à la liste de diffusion peuvent alors passer par un seul point qui fait le passage nécessaire et ensuite envoie en même temps le message au newsgroup et à la liste de diffusion, éliminant la plupart des possibilités de boucles. Une passerelle bidirectionnelle entre une liste de diffusion et un newsgroup non modéré est par contre difficile à faire correctement et est beaucoup plus fragile. Les Newsgroups destinés à être passés bidirectionnellement à une liste de diffusion DEVRAIENT donc être modérés lorsque possible, même si le modérateur est une simple passerelle et un agent injecteur qui traitent correctement le crosspostage à d'autres groupes modérés et passent autrement tout le trafic.

3.10.1 Devoirs d'une passerelle sortante

Du point de vue de Netnews, une passerelle sortante est juste un type particulier d'agent lecteur. La nature exacte de ce que la passerelle sortante va devoir faire aux articles dépend du support auquel les articles sont passés. Parce que cela soulève un danger de boucles dû à la possibilité qu'une ou plusieurs passerelles entrantes correspondantes reviennent de ce support à Netnews, le fonctionnement de la passerelle sortante est sujet à des contraintes supplémentaires.

La pratiques suivantes sont encouragées pour toute passerelle sortante, sans considération de si elle est connue pour être une passerelle entrante en relation, à la fois comme des mesures de précaution et comme lignes directrices pour la qualité de mise en œuvre :

1. L'identifiant de message de l'article de nouvelles devrait être préservé si c'est possible, de préférence comme ou dans l'identifiant univoque correspondant de l'autre support. Cependant, si il n'est pas préservé de cette façon, alors il devrait au moins être préservé comme commentaire dans le message. Cela aide beaucoup à prévenir les boucles.

2. Les Date et Injection-Date de l'article de nouvelles devraient aussi être préservées si possible, pour des raisons similaires.

3. Le message devrait être étiqueté d'une certaine façon pour empêcher sa réinjection dans Netnews. Cela peut être impossible à faire sans connaissance des passerelles entrantes potentielles, mais il vaut mieux essayer de fournir une indication même si cela ne réussit pas ; au moins, une indication lisible par l'homme que l'article ne devrait pas être repassé à Netnews peut aider à localiser un problème humain.

4. Les messages de contrôle Netnews ne devraient pas être passés à un autre support sauf si ils vont avoir une signification dans ce support.

3.10.2 Devoirs d'une passerelle entrante

La passerelle entrante a la responsabilité de s'assurer que toutes les exigences de ce protocole sont satisfaites par les articles qu'elle forme. En plus de ses devoirs particuliers comme passerelle, elle supporte tous les devoirs et responsabilités d'un agent posteur, et elle a les mêmes responsabilités qu'un agent relayeur pour rejeter les articles qu'elle a déjà passés.

Une passerelle entrante NE DOIT PAS passer deux fois le même message. Il peut n'être pas possible de s'assurer de cela en face de mutilation ou modification du message, mais au minimum, une passerelle, quand elle reçoit une copie d'un message qu'elle a déjà passé et qui est identique excepté les champs d'en-tête de trace (comme Received dans la messagerie électronique ou Path dans Netnews) NE DOIT PAS passer le message à nouveau. Une passerelle entrante DEVRAIT prendre des précautions contre l'outrepassement de cette règle par des modifications du message qui peuvent être anticipées.

Les articles de nouvelles préparés par des passerelles DOIVENT être des proto-articles de nouvelles valides (voir le paragraphe 3.4.1). Cela exige souvent que la passerelle synthétise un article conforme à partir d'une entrée non conforme. La passerelle DOIT alors passer l'article à un agent injecteur, pas directement à un agent relayeur.

Les passerelles entrantes NE DOIVENT PAS passer des messages de contrôle (articles contenant un champ d'en-tête Control ou Supersedes) sans supprimer ou renommer ce champ d'en-tête. Les passerelles PEUVENT cependant générer des messages de contrôle d'annulation pour les messages qu'elles ont passés. Si une passerelle reçoit un message dont elle peut déterminer qu'il est un équivalent valide d'un message de contrôle d'annulation dans le support quelle transpose, elle DEVRAIT éliminer ce message sans le passer, générer d'elle même un message de contrôle d'annulation correspondant, et injecter ce message de contrôle d'annulation.

Note : Il n'est pas question que des passerelles de messagerie à nouvelles soient utilisées pour poster des messages de contrôle, mais l'encapsulation devrait être utilisée à la place pour ces cas. Les passerelles par leur nature même sont particulièrement enclines aux boucles. Les rejets d'articles normaux sont assez mauvais ; les rejets de messages de contrôle avec une signification particulière pour le système de nouvelles, résultant éventuellement en une forte charge de traitement ou même en l'envoi de messages électroniques envoyés pour chaque message reçu, sont catastrophiques. Il est de loin préférable de construire un système spécifique pour poster les messages de contrôle qui peut faire les vérifications appropriées de cohérence et l'authentification de l'origine du message.

Si il y a un identifiant de message qui remplit un rôle similaire à celui du champ d'en-tête Message-ID dans les nouvelles, il DEVRAIT être utilisé dans la formation de l'identifiant de message de l'article de nouvelles, peut-être avec les transformations requises pour satisfaire l'exigence d'unicité de Netnews et avec la suppression de tous les commentaires de façon à se conformer à la syntaxe du paragraphe 3.1.3 de la [RFC5536]. De telles transformations DEVRAIENT être conçues de façon que deux messages avec le même identifiant génèrent le même champ d'en-tête Message-ID.

Note : Les identifiants de message jouent un rôle central dans la prévention de dupliqués, et leur usage correct par les passerelles va faire beaucoup pour empêcher les boucles. Netnews exige cependant que les identifiants de message soient uniques, et donc les identifiants de message provenant d'autres supports peuvent ne pas convenir pour une utilisation sans modification. Un équilibre doit être trouvé par la passerelle entre préserver l'information utilisée pour empêcher les boucles et générer des identifiants de message uniques.

Exceptionnellement, si il y a plusieurs passerelles entrantes pour un ensemble particulier de messages, chacun pour un newsgroup différent, chacune DEVRAIT générer un identifiant de message unique pour cette passerelle. Chaque passerelle entrante DOIT néanmoins s'assurer qu'elle ne passe pas deux fois le même message.

Note : Considérons l'exemple de deux passerelles d'une liste de diffusion donnée dans deux newsgroups Usenet séparés, qui tous deux préservent l'identifiant de message électronique. Chaque newsgroup peut alors recevoir une portion des messages (différents sites voyant différentes portions). Dans ce cas, où il n'y a pas une passerelle "officielle", une autre méthode de génération des identifiants de message doit être utilisée pour éviter des collisions. Il serait évidemment préférable qu'il y ait seulement une passerelle qui crossposte, mais cela peut n'être pas possible à coordonner.

Si aucune information de date n'est disponible, la passerelle PEUT fournir un champ d'en-tête Date avec la date courante de la passerelle. Si seulement des informations partielles sont disponibles (comme une date mais pas d'heure) cela DEVRAIT être complété comme une Date complète en ajoutant des valeurs par défaut plutôt que en éliminant cette information. C'est seulement dans des circonstances très exceptionnelles que les informations de date devraient être éliminées, car elles jouent un rôle important dans la prévention de la réinjection de vieux messages.

Une passerelle entrante DOIT ajouter un champ d'en-tête Sender à l'article de nouvelles qu'elle forme contenant la <mailbox> de l'administrateur de la passerelle. Les problèmes avec la passerelle peuvent être rapportés à cette <mailbox>. La portion <display-name> de cette <mailbox> DEVRAIT indiquer que l'entité responsable de l'injection du message est une passerelle. Si le message original avait déjà un champ d'en-tête Sender, il DEVRAIT être renommé en Original-Sender afin que son contenu puisse être préservé. Voir au paragraphe 3.10.3 la spécification de ce champ d'en-tête.

3.10.3 Champ d'en-tête Original-Sender

Le champ d'en-tête Original-Sender comporte le contenu d'un champ d'en-tête Sender dans un message original reçu par une passerelle entrante, le préservant alors que la passerelle entrante ajoute son propre champ d'en-tête Sender. La syntaxe de "content" utilise celle définie dans les [RFC5536] et [RFC5322].

header =/ Original-Sender-header
Original-Sender-header = "Original-Sender" ":" SP Original-Sender-content
Original-Sender-content = mailbox

L'entrée du répertoire d'en-tête permanent de message pour ce champ d'en-tête est :

Nom de champ d'en-tête : Original-Sender
Protocole applicable : Netnews
Statut : standard
Auteur/contrôleur des changements : IETF
Document de spécification : RFC 5537

3.10.4 Exemple de passerelle

Pour illustrer le type de précautions qui devraient être prises contre les boucles, voici un exemple de mesures prises par une combinaison particulière de passerelles de messagerie à nouvelles et de nouvelles à messagerie conçue pour traiter le passage bidirectionnel entre listes de diffusion et groupes non modérés :

1. La passerelle de nouvelles à messagerie préserve l'identifiant de message de l'article de nouvelles dans le message électronique généré. La passerelle de messagerie à nouvelles préserve de même l'identifiant de message de messagerie, pourvu qu'il soit syntaxiquement valide pour Netnews. Cela permet à la suppression incorporée du système de nouvelles des dupliqués de servir comme première ligne de défense contre les boucles.

2. La passerelle de nouvelles à messagerie ajoute un champ d'en-tête X-* à tous les messages qu'elle génère. La passerelle de messagerie à nouvelles élimine tous les messages entrants qui contiennent ce champ d'en-tête. Ceci est robuste contre les gestionnaires de liste de diffusion qui remplacent l'identifiant de message et contre tout nombre de bonds de messagerie, pourvu que les autres champs d'en-tête du message soient préservés.

3. La passerelle de messagerie à nouvelles ajoute le nom d'hôte d'où elle a reçu le message électronique au contenu du champ d'en-tête Path. La passerelle de nouvelles à messagerie refuse de passer tout message qui contient le nom du serveur de liste dans son champ d'en-tête Path (y compris dans la section de queue). Ceci est robuste contre toute quantité de mélange des champs d'en-tête du message par la liste de diffusion, pourvu que le message électronique passe seulement par un bond.

4. La passerelle de messagerie à nouvelles est conçue pour ne jamais générer de rebonds à l'envoyeur de l'enveloppe. À la place, les articles qui sont rejetés par le serveur de nouvelles (pour des raisons qui ne garantissent pas l'élimination en silence du message) résultent en un message de rebond envoyé à une adresse d'erreurs qui est connue pour ne pas transmettre à des listes de diffusion. De cette façon, elles peuvent être traitées pas les administrateurs de nouvelles.

Ces précautions ont prouvé leur efficacité en pratique pour empêcher les boucles pour cette application particulière (passerelle bidirectionnelle entre listes de diffusion et newsgroups distribués localement où les deux passerelles peuvent être ensemble). Une passerelle générale à des newsgroups mondiaux pose des difficultés supplémentaires ; on doit être très attentif aux configurations étranges, comme un newsgroup passé à une liste de diffusion qui est à son tour passé à un newsgroup différent.

4. Types de prise en charge

Le présent document définit plusieurs types de prise en charge, qui ont été enregistrées par l'IANA comme indiqué dans la [RFC4288].

Le type de support message/news, tel qu'enregistré précédemment par l'IANA, est ici déclaré obsolète. L'intention de ce type de support était de définir une façon standard de transmettre des articles de nouvelles via la messagerie électronique pour la lecture humaine. Cependant, il n'a jamais été largement mis en œuvre, et son traitement par défaut comme application/octet-stream par les agents qui ne le reconnaissaient pas était contre productif. Le type de support message/rfc822 (défini au paragraphe 5.2.1 de la [RFC2046]) DEVRAIT être utilisé à sa place.

La définition mise à jour du type de support MIME message/news est :
Nom de type MIME : message
Nom de sous type MIME : news
Paramètres exigés : aucun
Paramètres facultatifs : aucun
Considérations de codage : les mêmes que pour message/rfc822
Considérations de sécurité : les articles de nouvelles peuvent constituer des "messages de contrôle", qui peuvent avoir des effets sur le système de nouvelles d'un hôte au delà du simple ajout d'informations. Comme les messages de contrôle peuvent se produire dans un flux normal de nouvelles, la plupart des hôtes sont déjà convenablement défendus contre les effets indésirables, mais la transmission d'articles de nouvelles via la messagerie peut outrepasser les défenses de type pare-feu. Lire un article de nouvelles transmis par messagerie n'implique pas d'autres risques que ceux de la messagerie, mais le soumettre au logiciel de nouvelles pour son traitement devrait être fait avec prudence.
Considérations d'interopérabilité : rarement utilisé, et donc souvent traité comme application/octet-stream. La disposition devrait par défaut être "inline".
Spécification publiée : RFC 5537
Applications qui utilisent ce type de support : certaines vieilles messageries et certains nouveaux agents d'utilisateur.
Usage prévu : OBSOLÈTE
Auteur : Henry Spencer
Contrôleur des changements : IETF

4.1 application/news-transmission

Le type de support application/news-transmission est destiné à l'encapsulation d'articles de nouvelles complets où l'intention est que le receveur devrait alors les injecter dans Netnews. Ce type d'application fournit une des méthodes pour envoyer par messagerie électronique les articles aux modérateurs (paragraphe 3.5.1) et peut être utilisé pour porter les messages à un agent injecteur. Cette encapsulation supprime le besoin de transformer un message électronique en un proto-article Netnews et donne un moyen d'envoyer un article Netnews en utilisant MIME à travers un support de transport qui ne prend pas en charge les données "8bit".

La définition du type de support MIME de application/news-transmission est :
Nom de type MIME : application
Nom de sous type MIME : news-transmission
Paramètres exigés : aucun
Paramètres facultatifs : un et seulement un de "usage=moderate", "usage=inject", ou "usage=relay".
Considérations de codage : un codage de transfert différent de celui de l'article transmis PEUT être fourni pour assurer une transmission correcte sur certains supports de transport 7bit.
Considérations de sécurité : les articles de nouvelles peuvent constituer des "messages de contrôle", qui peuvent avoir des

effets sur le système de nouvelles d'un hôte au delà du simple ajout d'information. Comme les messages de contrôle peuvent se produire dans des flux de nouvelles normaux, la plupart des hôtes sont déjà convenablement défendus contre des effets indésirables, mais la transmission d'articles de nouvelles via la messagerie peut outrepasser les défenses de type pare-feu.

Spécification publiée : RFC 5537
Partie de corps : un proto-article complet prêt pour injection dans Netnews ou un article relayé à un autre agent.
Applications qui utilisent ce type de support : agents injecteurs, modérateurs Netnews.
Usage prévu : COMMUN
Contrôleur des changements : IETF

usage=moderate indique que l'article est destiné à un modérateur, usage=inject à un agent injecteur, et usage=relay à un agent relayeur. L'entité qui reçoit l'article peut seulement mettre en œuvre un type d'agent, et dans ce cas le paramètre PEUT être omis.

Contrairement à l'enregistrement précédent de ce type de support, les lots d'articles ne sont pas permis comme partie de corps. Plusieurs messages ou un message avec plusieurs parties application/news-transmission peuvent être utilisés à la place.

4.2 application/news-groupinfo

Le type de support application/news-groupinfo est utilisé en conjonction avec le message de contrôle newgroup (paragraphe 5.2.1). Sa partie de corps contient de brèves informations sur un newsgroup : le nom du newsgroup, sa description, et son statut de modération.

La définition du type de support MIME de application/news-groupinfo est :
Nom de type MIME : application
Nom de sous type MIME : news-groupinfo
Paramètres exigés : aucun
Paramètres facultatifs : charset, qui DOIT être un jeu de caractères enregistré pour l'utilisation avec les types de texte MIME. Il a la même syntaxe que le paramètre défini pour text/plain [RFC2046]. Il spécifie le jeu de caractères de la partie de corps. Si il n'est pas donné, le jeu de caractères par défaut est l'US-ASCII [ASCII].
Considérations de codage : un codage 7bit ou 8bit DOIT être utilisé pour conserver la compatibilité.
Considérations de sécurité : aucune.
Considérations d'interopérabilité : la disposition devrait par défaut être "inline".
Applications qui utilisent ce type de support : producteurs de message de contrôle, agents relayeurs, agents servants.
Spécification publiée : RFC 5537
Usage prévu : COMMUN
Contrôleur des changements : IETF

Le contenu de la partie de corps application/news-groupinfo est défini comme :

```
groupinfo-body = [ newsgroups-tag CRLF ]
                 newsgroups-line CRLF
newsgroups-tag = %x46.6F.72 SP %x79.6F.75.72 SP %x6E.65.77.73.67.72.6F.75.70.73 SP %x66.69.6C.65.3A
; sensible à la casse
; "Pour votre fichier newsgroups :"
newsgroups-line = newsgroup-name [ 1*HTAB newsgroup-description ] [ *WSP moderation-flag ]
newsgroup-description = eightbit-utext *( *WSP eightbit-utext )
moderation-flag = SP "(" %x4D.6F.64.65.72.61.74.65.64 ")"          ; ESPACE + sensible à la casse "(Moderated)"
eightbit-utext = VCHAR / %d127-255
```

Ce format inhabituel est rétro compatible avec l'examen du corps des messages de contrôle newgroup pour les descriptions faites par les mises en œuvre Netnews qui précèdent la présente spécification. Bien que facultatif, la <newsgroups-tag> DEVRAIT être incluse pour la rétro compatibilité.

La <newsgroup-description> NE DOIT PAS contenir d'occurrence de la chaîne "(Moderated)". Les newsgroups modérés DOIVENT être marqués en ajoutant le texte sensible à la casse " (Moderated)" à la fin.

Bien qu'un paramètre charset soit défini pour ce type de support, la plupart des logiciels existants ne comprennent pas les champs d'en-tête MIME ou ne traitent pas correctement les descriptions dans des jeux de caractères divers. Utiliser un jeu

de caratères US-ASCII lorsque possible est donc RECOMMANDÉ ; si ce n'est pas possible, UTF-8 [RFC3629] DEVRAIT être utilisé. Sans considération du jeu de caractères utilisé, les contraintes de la grammaire ci-dessus DOIVENT être satisfaites et le <newsgroup-name> DOIT être représenté dans ce jeu de caractères en utilisant les mêmes octets que ceux qui auraient été utilisés avec un jeu de caractères US-ASCII.

4.3 application/news-checkgroups

Le type de support application/news-checkgroups contient une liste de newsgroups dans une ou des hiérarchies, incluant leur description et statut de modération. Il est principalement utilisé avec le message de contrôle checkgroups (voir le paragraphe 5.2.3).

La définition du type de support MIME de application/news-checkgroups est :
Nom de type MIME : application
Nom de sous type MIME : news-checkgroups
Paramètres exigés : aucun
Paramètres facultatifs : charset, qui DOIT être un jeu de caractères enregistré pour l'utilisation avec les types de texte MIME. Il a la même syntaxe que le paramètre défini pour text/plain [RFC2046]. Il spécifie le jeu de caractères de la partie de corps. Si il n'est pas donné, le jeu de caractères par défaut est l'US-ASCII [ASCII].
Considérations de codage : un codage 7bit ou 8bit DOIT être utilisé pour conserver la compatibilité.
Considérations de sécurité : ce type de support donne seulement un moyen pour transporter une liste de newsgroups et ne fournit aucune information indiquant si l'envoyeur est autorisé à déclarer quels newsgroups devraient exister dans une hiérarchie. Cette autorisation doit être obtenue par d'autres moyens.
Considérations d'interopérabilité : la disposition devrait par défaut être "inline".
Applications qui utilisent ce type de support : producteurs de message de contrôle, agents relayeurs, agents servants.
Spécification publiée : RFC 5537
Usage prévu : COMMUN
Contrôleur des changements : IETF

Le contenu de la partie de corps application/news-checkgroups est définie comme :

```
checkgroups-body = *( valid-group CRLF )
valid-group = newsgroups-line                    ; voir le paragraphe 4.2
```

Les mêmes restrictions sur charset, <newsgroup-name>, et <newsgroup-description> que pour application/news-groupinfo s'appliquent pour ce type de support.

Un message application/news-checkgroups peut contenir des informations pour une ou plusieurs hiérarchies et est considéré complet pour toute hiérarchie pour laquelle il contient un <valid-group> sauf si il est accompagné par des informations externes qui limitent sa portée (comme un paramètre <chkscope> à un message de contrôle checkgroups, comme décrit au paragraphe 5.2.3). En d'autres termes, une partie de corps application/news-checkgroups consistant en

```
exemple.moderated     un newsgroup modéré (Moderated)
exemple.test          un groupe d'essai non modéré
```

est une déclaration que la hiérarchie exemple.* contient deux newsgroups, exemple.moderated et exemple.test, et pas d'autre. Ce type de support NE DOIT donc PAS être utilisé pour porter des informations partielles sur une hiérarchie ; si un groupe provenant d'une hiérarchie donnée est présent, tous les groupes qui existent dans cette hiérarchie DOIVENT être mentionnés dans la liste sauf si sa portée est limitée par des informations externes, auquel cas tous les groupes DEVRAIENT figurer dans la liste.

Des espaces sont utilisées dans cet exemple pour des raisons de formatage. Dans un message réel, le nom de newsgroup et sa description DOIVENT être séparés par une ou plusieurs tabulations (HTAB, ASCII %d09) pas par des espaces.

5. Messages de contrôle

Un message de contrôle est un article qui contient un champ d'en-tête Control et indique par là qu'une action devrait être entreprise par un agent autre que de distribution et d'affichage. Tout article contenant un champ d'en-tête Control (défini au paragraphe 3.2.3 de la [RFC5536]) est un message de contrôle. De plus, l'action d'un article contenant un champ d'en-tête Supersedes est décrite ici ; bien qu'un tel article ne soit pas un message de contrôle, il spécifie une action similaire au

message de contrôle "cancel".

La <control-command> d'un champ d'en-tête Control comporte un <verb>, qui indique l'action à entreprendre, et une ou plusieurs valeurs de <argument>, qui fournissent les détails. Pour certains messages de contrôle, le corps de l'article est aussi significatif. Chaque <verb> reconnu (le type de message de contrôle) est décrit dans un paragraphe séparé ci-dessous. Les agents PEUVENT accepter d'autres types de message de contrôle que ceux spécifiés ci-dessous, et DOIVENT soit ignorer, soit rejeter ces messages de contrôle avec des types non reconnus. Les définitions syntaxiques des valeurs valides des <argument> et les restrictions sur les corps de message de contrôle sont données dans les paragraphes sur chaque type de message de contrôle.

Contrairement à la [RFC1036], la présence d'un champ d'en-tête Subject commençant par la chaîne "cmsg " NE DOIT PAS causer l'interprétation d'un article comme étant un message de contrôle. Les agents PEUVENT rejeter un article qui a un tel champ d'en-tête Subject et pas de champ d'en-tête Control comme ambigu. De même, la présence d'un <newsgroup-name> se terminant en ".ctl" dans le champ d'en-tête Newsgroups ou la présence d'un champ d'en-tête Also-Control NE DOIT PAS causer l'interprétation de l'article comme étant un message de contrôle.

5.1 Authentification et autorisation

Les messages de contrôle spécifient des actions au-dessus et en-dessous du traitement normal d'un article et sont donc de potentiels vecteurs d'abus et d'action non autorisée. Il n'y a, à présent, pas de moyen normalisé d'authentifier l'envoyeur d'un message de contrôle ou de vérifier que le contenu d'un message de contrôle a été envoyé par l'envoyeur prétendu. Il y a, cependant, des mécanismes d'authentification non normalisés d'usage courant, comme [PGPVERIFY].

Les agents qui agissent sur les messages de contrôle DEVRAIENT prendre des mesures pour authentifier les messages de contrôle avant d'agir sur eux, selon la politique d'autorisation locale. Que ce soit fait via l'utilisation d'un protocole d'authentification non normalisé, par comparaison avec les informations obtenues par un autre protocole, par un examen humain, ou par d'autres moyens, n'est pas spécifié par le présent document. De futures extensions ou révisions de ce protocole sont supposées normaliser un mécanisme de signature numérique.

Les agents sont supposés avoir leur propre politique locale d'autorisation pour les messages de contrôle qui vont être honorés. Aucun agent Netnews n'est jamais obligé d'agir sur un message de contrôle. Les descriptions qui suivent spécifient les actions que demande un message de contrôle, mais un agent PEUT toujours refuser d'agir sur un message de contrôle.

5.2 Messages de contrôle de groupe

Un message de contrôle de groupe est tout type de message de contrôle qui demande une mise à jour de la liste des newsgroups connus d'un serveur de nouvelles. Les types standard de message de contrôle de groupe sont "newgroup", "rmgroup", et "checkgroups".

Avant d'honorer un message de contrôle de groupe, un agent DOIT vérifier le ou les newsgroups affectés par ce message de contrôle et refuser de créer des newsgroups non conformes aux restrictions du paragraphe 3.1.4 de la [RFC5536].

Tous les messages de contrôle de groupe DOIVENT avoir un champ d'en-tête Approved (paragraphe 3.2.1 de la [RFC5536]). Les messages de contrôle de groupe sans un champ d'en-tête Approved NE DEVRAIENT PAS être honorés.

Les messages de contrôle de groupe qui affectent des groupes spécifiques (messages de contrôle newgroup et rmgroup, par exemple) DEVRAIENT inclure le <newsgroup-name> pour le ou les groupes affectés dans leur champ d'en-tête Newsgroups. D'autres newsgroups PEUVENT être inclus dans le champ d'en-tête Newsgroups afin que le message de contrôle atteigne plus de serveurs de nouvelles, mais du fait des règles particulières de relayage pour les messages de contrôle de groupe (voir le paragraphe 3.6) ceci n'est normalement pas nécessaire et peut être excessif.

5.2.1 Message de contrôle newgroup

Le message de contrôle newgroup demande que le groupe spécifié soit créé ou, si il existe déjà, que son statut de modération ou sa description soit changé. La syntaxe de son champ d'en-tête est :

control-command =/ Newgroup-command
Newgroup-command = "newgroup" Newgroup-arguments
Newgroup-arguments = 1*WSP newsgroup-name [1*WSP newgroup-flag]

newgroup-flag = "moderated"

Si la demande est honorée, le statut de modération du groupe DEVRAIT être réglé en accord avec la présence ou l'absence du fanion <newgroup-flag> "moderated". "moderated" est le seul fanion défini par le protocole. D'autres fanions PEUVENT être définis par des extensions à ce protocole et acceptés par les agents. Si un agent ne reconnaît pas le <newgroup-flag> d'un message de contrôle newgroup, il DEVRAIT ignorer ce message de contrôle.

Le corps d'un message newgroup DEVRAIT contenir une entité de type application/news-groupinfo spécifiant la description du newsgroup, soit comme corps entier, soit comme une entité au sein d'un objet multipart/mixed [RFC2046]. Si une telle entité est présente, le statut de modération qui y est spécifié DOIT correspondre au statut de modération spécifié par le <newgroup-flag>. Le corps d'un message newgroup PEUT contenir d'autres entités (encapsulées dans multipart/mixed) qui fournissent des informations supplémentaires sur le newsgroup ou les circonstances du message de contrôle.

En l'absence d'une entité application/news-groupinfo, un serveur de nouvelles PEUT chercher le corps du message sur la ligne "Pour votre fichier newsgroups :" et prendre la ligne suivante comme <newsgroups-line>. Avant la normalisation de application/news-groupinfo, c'était la convention pour fournir la description d'un newsgroup.

Si la demande est honorée et contient la description d'un newsgroup, et si le serveur de nouvelles qui l'honore mémorise des descriptions de newsgroup, la description de newsgroup mémorisée DEVRAIT être mise à jour avec la description spécifiée dans le message de contrôle, même si aucune autre propriété du groupe n'a changé.

5.2.1.1 Exemple de message de contrôle newgroup

Un message de contrôle newgroup demandant la création du newsgroup modéré exemple.admin.info.

```
From: "exemple.* Administrator" <admin@noc.exemple>
Newsgroups: exemple.admin.info
Date: 27 Feb 2002 12:50:22 +0200
Subject: cmsg newgroup exemple.admin.info moderated
Approved: admin@noc.exemple
Control: newgroup exemple.admin.info moderated
Message-ID: <ng-exemple.admin.info-20020227@noc.exemple>
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="nxtprt"
Content-Transfer-Encoding: 8bit

C'est un message de contrôle MIME.
    --nxtprt
    Content-Type: application/news-groupinfo; charset=us-ascii

    For your newsgroups file:
    exemple.admin.info      À propos des groupes exemple.* (Moderated)

    --nxtprt
    Content-Type: text/plain; charset=us-ascii

Un newsgroup modéré pour des annonces sur de nouveaux newsgroups dans la hiérarchie exemple.*.

    --nxtprt--
```

Des espaces sont utilisées dans cet exemple pour des raisons de formatage. Dans un message réel, le nom de newsgroup et sa description DOIVENT être séparés par une ou plusieurs tabulations (HTAB, ASCII %d09) pas des espaces.

5.2.2 Message de contrôle rmgroup

Le message de contrôle rmgroup demande que le groupe spécifié soit supprimé de la liste des groupes valides d'un serveur de nouvelles. La syntaxe de son champ d'en-tête Control est :

control-command =/ Rmgroup-command

Rmgroup-command = "rmgroup" Rmgroup-arguments
Rmgroup-arguments = 1*WSP newsgroup-name

Le corps du message de contrôle PEUT contenir n'importe quoi, généralement un texte explicatif.

5.2.3 Message de contrôle checkgroups

Le message de contrôle checkgroups contient une liste de tous les groupes valides dans une hiérarchie avec les descriptions et le statut de modération. Il demande qu'un serveur de nouvelles mette à jour sa liste des newsgroups valides pour cette hiérarchie pour inclure les groupes spécifiés, supprime tous groupes non spécifiés, et mette à jour les descriptions de groupe et leur statut de modération pour correspondre à ceux donnés dans le message de contrôle checkgroups. La syntaxe de ce champ d'en-tête Control est :

control-command =/ Checkgroup-command
Checkgroup-command = "checkgroups" Checkgroup-arguments
Checkgroup-arguments = [chkscope] [chksernr]
chkscope = 1*(1*WSP ["!"] newsgroup-name)
chksernr = 1*WSP "#" 1*DIGIT

Un message checkgroups est interprété comme une liste exhaustive des groupes valides dans toutes les hiérarchies ou sous hiérarchies avec un préfixe figurant dans la liste de l'argument <chkscope>, à l'exclusion de toute sous hiérarchie où l'argument <chkscope> a un préfixe "!". Pour les cas complexes avec plusieurs arguments <chkscope>, on commence par une liste vide des groupes, on inclut tous les groupes dans le message de contrôle checkgroups qui correspondent à l'argument <chkscope> sans préfixe "!", et ensuite on exclut tous les groupes qui correspondent aux arguments <chkscope> avec un préfixe "!". On suit cette méthode sans considération de l'ordre des arguments <chkscope> dans le champ d'en-tête Control.

Si aucun argument <chkscope> n'est donné, il s'applique à toutes les hiérarchies pour lesquelles des déclarations de groupe apparaissent dans le corps du message.

Comme beaucoup des logiciels existants ne respectent pas l'argument <chkscope>, le corps du message de contrôle checkgroups NE DOIT PAS contenir de déclaration de groupe pour des newsgroups en dehors de la portée prévue et DEVRAIT contenir une liste de newsgroups correcte même pour les sous hiérarchies exclues avec des termes de <chkscope> "!". Les serveurs de nouvelles DOIVENT cependant honorer <chkscope> comme spécifié ici.

L'argument <chksernr> peut être tout entier positif. Si il est présent, il DOIT s'accroître avec chaque changement de la liste de newsgroups, NE DOIT jamais décroître, et DOIT être inclus dans tous les messages de contrôle checkgroups suivants avec la même portée. Si il est fourni, les serveurs de nouvelles DEVRAIENT se souvenir de la valeur de <chksernr> des précédents message de contrôle checkgroups honorés pour une hiérarchie ou sous hiérarchie particulière et refuser d'honorer tout message de contrôle checkgroups suivant pour la même hiérarchie ou sous hiérarchie avec un plus petite valeur de <chksernr> ou sans valeur de <chksernr>.

Il n'y a pas de limite supérieure à la longueur de <chksernr>, autre que la limitation sur la longueur des champs d'en-tête. Les mises en œuvre peuvent donc vouloir faire des comparaisons en bourrant de zéro la plus courte des deux valeurs <chksernr> sur la gauche et en faisant alors une comparaison de chaîne, plutôt que de supposer que <chksernr> peut être manipulé comme un nombre.

Par exemple, le champ d'en-tête Control suivant

> Control: checkgroups de !de.alt #2009021301

indique que le corps du message va faire la liste de tous les newsgroups dans la hiérarchie de.*, à l'exception de la sous hiérachie de.alt.*, et ne devrait pas être honoré si un message de contrôle checkgroups avec un numéro de série supérieur à 2009021301 a été honoré précédemment. Le numéro de série dans cet exemple a été formé de la date *(13 février 2009)* en format AAAAMMJJ, suivie par un numéro de séquence de deux chiffres *(01)* dans cette date.

Le corps du message est une entité de type application/news-checkgroups. Il DEVRAIT être déclaré comme tel avec les en-têtes MIME appropriés, mais les serveurs de nouvelles DEVRAIENT interpréter les messages checkgroups qui manquent des en-têtes MIME appropriés comme si le corps était de type application/news-checkgroups pour la rétro compatibilité.

5.3 Message de contrôle cancel

Le message de contrôle cancel demande que l'article cible soit retiré de la circulation et de l'accès. La syntaxe de ce champ d'en-tête Control est :

```
control-command =/ Cancel-command
Cancel-command = "cancel" Cancel-arguments
Cancel-arguments = 1*WSP msg-id
```

L'argument identifie l'article à annuler par son identifiant de message. Le corps du message de contrôle PEUT contenir n'importe quoi, généralement un texte explicatif.

Un agent servant qui choisit d'honorer un message d'annulation DEVRAIT rendre l'article indisponible aux agents lecteurs (peut-être en le supprimant complètement). Si le message de contrôle cancel arrive avant l'article qu'il cible, les serveurs de nouvelles qui choisissent de l'honorer DEVRAIENT se souvenir de l'identifiant de message qui a été annulé et rejeter l'article annulé quand il arrive.

Pour mieux s'assurer qu'il va être relayé aux mêmes serveurs de nouvelles que le message original, un message de contrôle cancel DEVRAIT avoir le même champ d'en-tête Newsgroups que le message qu'il annule.

Les messages de contrôle Cancel qui font la liste des newsgroups modérés dans leur champ d'en-tête Newsgroups DOIVENT contenir un champ d'en-tête Approved comme tout autre article dans un newsgroup modéré. Cela signifie que les annulations postées à un newsgroup modéré vont normalement être envoyées d'abord au modérateur pour approbation. En dehors des newsgroups modérés, les messages cancel ne sont pas obligés de contenir de champ d'en-tête Approved.

Contrairement à la [RFC1036], les messages de contrôle cancel ne sont pas obligés de contenir des champs d'en-tête From et Sender correspondant au message cible. Cette exigence encourageait seulement les producteurs d'annulation à dissimuler leur identité et ne fournit aucune sécurité.

5.4 Champ d'en-tête Supersedes

La présence d'un champ d'en-tête Supersedes dans un article demande que l'identifiant de message donné dans ce champ d'en-tête soit supprimé exactement de la même manière que si c'était la cible d'un message de contrôle cancel. En conséquence, les serveurs de nouvelles DEVRAIENT appliquer à un champ d'en-tête Supersedes les mêmes vérifications d'authentification et d'autorisation que celles qu'ils appliqueraient pour les messages de contrôle cancel. Si le champ d'en-tête Supersedes est honoré, le serveur de nouvelles DEVRAIT prendre les mêmes actions que quand il honore un message de contrôle cancel pour cet article cible.

L'article qui contient le champ d'en-tête Supersedes, que celui-ci soit honoré ou non, DEVRAIT être traité comme un article normal et NE DEVRAIT PAS recevoir le traitement particulier de message de contrôle décrit au paragraphe 3.7.

5.5 Messages de contrôle ihave et sendme

Les messages de contrôle ihave et sendme mettent en œuvre un prédécesseur du protocole NNTP [RFC3977]. Ils sont largement obsolètes dans l'Internet mais voient encore une utilisation en conjonction avec certains protocoles de transport comme UUCP [UUCP]. Les serveurs de nouvelles ne sont pas obligés de les prendre en charge.

Les messages de contrôle ihave et sendme partagent une syntaxe similaire pour leurs champs d'en-tête Control et leurs corps :

```
control-command =/ Ihave-command
Ihave-command = "ihave" Ihave-arguments
Ihave-arguments = 1*WSP *( msg-id 1*WSP ) relayer-name
control-command =/ Sendme-command
Sendme-command = "sendme" Sendme-arguments
Sendme-arguments = Ihave-arguments
relayer-name = path-identity                    ; voir le paragraphe 3.1.5 de la [RFC5536]
ihave-body = *( msg-id CRLF )
sendme-body = ihave-body
```

Le corps de l'article consiste en une liste de <msg-id>, un par ligne. Les identifiants de message DEVRAIENT être mis

dans le corps de l'article, pas dans le champ d'en-tête Control, mais les serveurs de nouvelles PEUVENT reconnaître et traiter les identifiants de message dans le champ d'en-tête Control pour la rétro compatibilité. Les identifiants de message NE DOIVENT PAS être mis dans le champ d'en-tête Control si ils sont présents dans le corps du message de contrôle.

Le message ihave déclare que l'agent relayeur désigné a reçu des articles avec les identifiants de message spécifiés, qui peuvent intéresser les agents relayeurs qui reçoivent le message ihave. Le message sendme demande que l'agent qui le reçoit envoie les articles qui ont les identifiants de message spécifiés à l'agent relayeur désigné. Au contraire de la [RFC1036], le nom du relayeur DOIT être donné comme dernier argument dans le champ d'en-tête Control.

À réception du message sendme (et sur décision de l'honorer) l'agent receveur envoie le ou les articles demandés. Le mécanisme de transmission n'est pas spécifié dans ce document et est arrangé entre les sites impliqués.

Ces messages de contrôle sont normalement envoyés comme des articles en point à point entre deux sites et ne sont pas envoyés ensuite sur d'autres sites. Les newsgroups qui commencent par "to." sont réservés pour de telles communications en point à point et sont formés en ajoutant devant "to." un <relayer-name> pour former un <newsgroup-name>. Les articles avec un tel groupe dans leur champs d'en-tête Newsgroups NE DEVRAIENT PAS être envoyés à un serveur de nouvelles autre que celui identifié par <relayer-name>.

5.6 Messages de contrôle obsolètes

Les types de message de contrôle suivants sont déclarés obsolètes par ce document et NE DEVRAIENT PAS être envoyés ou honorés :

sendsys
version
whogets
senduuname

6. Considérations sur la sécurité

Netnews est conçu pour une large dissémination de messages publics et offre peu en matière de sécurité. La protection que Netnews a contre l'abus et l'usurpation d'identité est fournie principalement par la couche de transport sous-jacente. Dans les grands réseaux Netnews où on ne peut pas s'appuyer sur les serveurs de nouvelles pour appliquer les exigences d'authentification et d'autorisation à la couche transport, les articles peuvent être facilement falsifiés et largement lus, et les identités des envoyeurs d'article et la confidentialité des articles ne peuvent pas être assurées.

Voir à la Section 5 de la [RFC5536] d'autres considérations de sécurité relatives au format des articles.

6.1 Compromission de l'intégrité du système

Les messages de contrôle posent un problème de sécurité particulier car agir sur des messages de contrôle non autorisés peut causer la suppression de newsgroups, la suppression d'articles, et la création non voulue de newsgroups. Les administrateurs de serveurs de nouvelles DEVRAIENT donc prendre des mesures pour vérifier l'authenticité des messages de contrôle comme discuté au paragraphe 5.1. Les articles contenant des champs d'en-tête Supersedes sont effectivement des messages de contrôle d'annulation et DEVRAIENT être soumis aux mêmes vérifications que discuté au paragraphe 5.4. Actuellement, de nombreux sites ignorent tous les messages de contrôle cancel et les champs d'en-tête Supersedes du fait de la difficulté de les authentifier et de leur abus répandu.

Les messages de contrôle Cancel ne sont pas obligés d'avoir le même champ d'en-tête Newsgroups que les messages qu'ils annulent. Comme ils sont parfois traités avant que le message original soit reçu, il peut n'être pas possible de vérifier que le champs d'en-tête Newsgroup correspond. Cela permet à un posteur malveillant d'injecter un message de contrôle cancel pour un article dans un newsgroup modéré sans ajouter de champ d'en-tête Approved au message de contrôle, et de cacher les messages de contrôle cancel malveillants à certains agents lecteurs en utilisant un champ d'en-tête Newsgroups différent de sorte que le message de contrôle cancel n'est pas accepté par tous les serveurs de nouvelles qui ont accepté le message original.

Tous les agents devraient être conscients que tout contenu d'article, en particulier incluant le contenu du champ d'en-tête Control, est potentiellement non fiable et malveillant. Des articles peuvent être construits de façon syntaxiquement invalide pour tenter de submerger les mémoires tampons internes, violer des hypothèses cachées, ou exploiter des faiblesses de mise

en œuvre. Par exemple, des mises en œuvre de serveur de nouvelles ont été attaquées avec succès via l'inclusion d'un code coquille Unix dans le champ d'en-tête Control. Tout contenu d'article, et en particulier le contenu de message de contrôle, DEVRAIT être traité avec soin et rigoureusement vérifié avant toute action sur la base du contenu de l'article.

Un posteur malvaillant peut ajouter un champ d'en-tête Approved pour outrepasser le processus de modération d'un newsgroup modéré. Les agents injecteurs DEVRAIENT vérifier que les messages approuvés pour un newsgroup modéré sont injectés par le modérateur en utilisant les informations d'authentification provenant du transport sous-jacent ou d'un autre mécanisme d'authentification en accord avec le modérateur. Il n'y a, à présent, aucune méthode normalisée d'authentifier l'approbation des messages aux groupes modérés, bien que des méthodes non normalisées d'authentification comme [PGPMOOSE] soient d'utilisation courante.

Un serveur de nouvelles malveillant participant à un réseau Netnews peut outrepasser les vérifications effectuées par les agents injecteurs, falsifier les champs d'en-tête Path et les autres informations de trace (comme les champs d'en-tête Injection-Info) et autrement compromettre les exigences d'autorisation d'un réseau Netnews. Les serveurs de nouvelles DEVRAIENT utiliser les facilités du transport sous-jacent pour authentifier leur homologues et rejeter les articles provenant d'agents injecteurs et relayeurs qui ne suivent pas les exigences de ce protocole ou du réseau Netnews.

6.2 Déni de service

Le fonctionnement approprié des newsgroups individuels peut être perturbé par le postage excessif d'articles non voulus, par le postage répété d'articles identiques ou presque identiques, par le postage de suivis qui sont sans relation avec leurs précurseurs ou qui citent entièrement leurs précurseurs avec l'ajout de matériel supplémentaire minimal (en particulier si ce processus est réitéré), en crosspostant, ou en demandant des suivis à des newsgroups sans aucune relation, et en abusant de messages de contrôle et du champ d'en-tête Supersedes pour supprimer des articles ou newsgroups.

De tels articles destinés à dénier le service, ou d'autres articles de nature offensante, peut aussi avoir leurs adresses From ou Reply-To réglées à des adresses de messagerie valides mais incorrectes, causant donc l'arrivée de gros volumes de messages sur les vrais possesseurs de ces adresses. Les utilisateurs et les agents devraient toujours être conscients que les informations d'identification dans les articles peuvent être falsifiées.

Un posteur malveillant peut empêcher un article d'être vu à un site particulier en incluant dans le champ d'en-tête Path du proto-article la <path-identity> de ce site. L'utilisation du <diag-keyword> "POSTED" par les agents injecteurs pour marquer le point d'injection peut empêcher cette attaque.

La principale responsabilité pour empêcher de telles attaques repose sur les agents injecteurs, qui peuvent appliquer les vérifications d'authentification et d'autorisation via le transport sous-jacent et empêcher ces tentatives d'attaque de déni de service de poster des messages. Si des agents injecteurs spécifiques manquent à assumer leurs responsabilités, ils peuvent être exclus du réseau Netnews en configurant les agents relayeurs à rejeter les articles qu'ils génèrent.

Une plainte malveillante peut soumettre une copie modifiée d'un article (avec un champ d'en-tête Injection-Info altéré, par exemple) à l'administrateur d'un agent injecteur dans une tentative pour discréditer l'auteur de cet article et même de faire révoquer ses privilèges de posteur. Les administrateurs DEVRAIENT donc obtenir une copie véritable de l'article de leur propre agent servant avant d'effectuer une action en réponse à une telle plainte.

6.3 Fuites

Les articles qui sont destinés à avoir une distribution restreinte dépendent de la bonne volonté de chaque site qui les reçoit. Les restrictions à la dissémination et à la rétention d'articles peuvent être demandées via les champs d'en-tête Archive et Distribution, mais de telles demandes ne peuvent pas être appliquées par le protocole.

L'algorithme d'arrosage utilisé par les transports Netnews comme NNTP [RFC3977] est extrêmement bon pour trouver tout chemin par lequel les articles peuvent quitter un sous réseau avec des limites supposées restrictives, et un effort administratif substantiel est nécessaire pour éviter cela. Les organisations qui souhaitent contrôler de telles fuites sont invitée à désigner un petit nombre de passerelles pour traiter tous les échanges de nouvelles avec le monde extérieur.

Le message de contrôle sendme (paragraphe 5.5) tel qu'il est encore utilisé jusqu'à présent, peut être utilisé pour demander des articles auxquels le demandeur ne devrait pas avoir accès.

7. Considérations relatives à l'IANA

L'IANA a enregistré les types de supports suivants, décrits ailleurs dans ce document, à utiliser avec le champ d'en-tête Content-Type, dans l'arborescence de l'IETF en accord avec les procédures de la [RFC4288].

application/news-transmission (4.1)
application/news-groupinfo (4.2)
application/news-checkgroups (4.3)

application/news-transmission est un changement d'un enregistrement précédent.

L'IANA a enregistré le nouveau champ d'en-tête, Original-Sender, dans le répertoire permanent des champs d'en-tête de message, en utilisant le gabarit du paragraphe 3.10.3.

L'IANA a changé le statut du type de support message/news en "OBSOLÈTE". Le type de support message/rfc822 devrait être utilisé à la place. Un gabarit mis à jour est inclus à la Section 4.

8. Références

8.1 Références normatives

[ASCII] American National Standard for Information Systems, "Coded Character Sets - 7-Bit American National Standard Code for Information Interchange (7-Bit ASCII)", ANSI X3.4, 1986.

[RFC2046] N. Freed et N. Borenstein, "Extensions de messagerie Internet multi-objets (MIME) Partie 2 : Types de support", novembre 1996. *(D. S., MàJ par* 2646, 3798, 5147, 6657, 8098)

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. *(MàJ par* RFC8174)

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003, DOI 10.17487/RFC3629.

[RFC4288] N. Freed et J. Klensin, "Spécifications du type de support et procédures d'enregistrement", BCP 13, décembre 2005.

[RFC5234] D. Crocker, P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", janvier 2008. (STD0068)

[RFC5322] P. Resnick, éd., "Format du message Internet", octobre 2008. (*Remplace* RFC2822) (MàJ RFC4021) *(D.S.)*

[RFC5536] K. Murchison, C. Lindsey, D. Kohn, "Format des articles pour Netnews", novembre 2009. *(Remplace* RFC1036 *(P.S.)*

8.2 Références pour information

[PGPMOOSE] Rose, G., "PGP Moose", novembre 1998.

[PGPVERIFY] Lawrence, D., "Signing Control Messages", août 2001, <ftp://ftp.isc.org/pub/pgpcontrol/FORMAT>.

[RFC1036] M. Horton et R. Adams, "Norme pour l'échange de messages USENET", décembre 1987. *(Remplacée par* RFC5536)

[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.)

[RFC2606] D. Eastlake 3rd et A. Panitz, "Noms réservés de niveau supérieur du DNS", BCP 32, juin 1999.

[RFC3798] T. Hansen et G. Vaudreuil, éd., "Notification de disposition de message", mai 2004. (*MàJ par* RFC5337,

RFC6533) *(D.S.; Rendue obsolète par* RFC8098)

[RFC3977] C. Feather, "Protocole de transfert de nouvelles du réseau (NNTP)", octobre 2006. *(P.S., MàJ par* RFC 6048

[RFC1849] H. Spencer, "Fils de 1036" : Format et transmission d'articles de nouvelles", octobre 1995. (*Obsolète, voir* 5536, 5537)

[USEAGE] Lindsey, C., "Usenet Best Practice", Travail en cours, mars 2005.

[UUCP] O'Reilly, T. and G. Todino, "Managing UUCP and Usenet", O'Reilly & Associates Ltd., janvier 1992.

Appendice A. Changements aux protocoles existants

Le présent document prescrit de nombreux changements, précisions, et nouvelles caractéristiques par rapport au protocole décrit dans la [RFC1036]. En particulier :

- Un nouveau format de champ d'en-tête Path rétro compatible qui permet une incorporation normalisée des informations supplémentaire de trace et d'authentification est maintenant RECOMMANDÉ. Voir le paragraphe 3.2. Le retour à la ligne de l'en-tête Path est permis.
- L'élagage du champ d'en-tête References est EXIGÉ, et un mécanisme pour le faire est défini.
- L'ajout du nouveau champ d'en-tête Injection-Date est exigé dans certaines circonstances pour les agents posteurs (paragraphe 3.4.2) et les agents injecteurs (paragraphe 3.5) et DOIT être utilisé par les serveurs de nouvelles pour les vérifications de date (paragraphe 3.6). Les agents injecteurs sont aussi fortement encouragés à mettre toutes les informations de trace locales dans le nouveau champ d'en-tête Injection-Info.
- Un nouveau type de support est défini pour transmettre les articles Netnews à travers d'autres supports (paragraphe 4.1) et les modérateurs DEVRAIENT se préparer à recevoir des soumissions dans ce format (paragraphe 3.5.1).
- Certains messages de contrôle (paragraphe 5.6) sont déclarés obsolètes, et la signification particulière de "cmsg" au début d'un champ d'en-tête Subject est supprimée.
- Des types de supports supplémentaires sont définis pour améliorer la structuration, la spécification, et le traitement automatique des messages de contrôle (paragraphes 4.2 et 4.3).
- Deux nouveaux paramètres facultatifs sont ajoutés au message de contrôle "checkgroups".
- Le type de support message/news est déclaré obsolète.
- Les messages de contrôle "Cancel" ne sont plus obligés d'avoir les champs d'en-tête From et Sender correspondants à ceux du message cible, car cette exigence n'ajoute rien à la sécurité réelle.
- Le paramètre "relayer-name" dans le champ d'en-tête Control des messages de contrôle ihave et sendme est maintenant exigé.

De plus, de nombreuses étapes du protocole et exigences de vérification d'article qui n'étaient pas mentionnées ou étaient ambiguës dans la [RFC1036] mais sont largement mises en œuvre par les serveurs Netnews ont été normalisées et spécifiées en détails.

Appendice B. Remerciements

Le présent document est le résultat de douze années d'efforts et le nombre de personnes qui ont contribué à son contenu est trop grand pour qu'on les mentionne individuellement. Nos remerciement vont à tous les membres passés et présents du groupe de travail USEFOR de l'IETF et de la liste de diffusion qui l'accompagne.
Nos remerciements particuliers à Henry Spencer, dont le projet [RFC1849] a servi de base initiale à ce document.

Adresse des auteurs

Russ Allbery (éditeur)
Stanford University
P.O. Box 20066
Stanford, CA 94309
US
mél : rra@stanford.edu
URI : http://www.eyrie.org/~eagle/

Charles H. Lindsey
5 Clerewood Avenue
Cheadle
Cheshire SK8 3JU
UK
téléphone : +44 161 436 6131