Groupe de travail Réseau
Request for Comments : 5435
 Catégorie : Sur la voie de la normalisation
 Traduction Claude Brière de L'Isle

A. Melnikov, éd., Isode Limited
 B. Leiba, éd.,
 W. Segmuller, IBM T.J. Watson Research Center
 T. Martin, Endless Crossword
 janvier 2009

Filtrage de messagerie Sieve : extension pour les notifications

Statut du présent mémoire

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

Notice de droits de reproduction

Copyright (c) 2009 IETF Trust et les personnes identifiées comme auteurs du document. Tous droits réservés.

Le présent document est soumis au BCP 78 et aux dispositions légales de l'IETF Trust qui se rapportent aux documents de l'IETF (<http://trustee.ietf.org/license-info>) en vigueur à la date de publication de ce document. Prière de revoir ces documents avec attention, car ils décrivent vos droits et obligations par rapport à ce document.

Résumé

Les utilisateurs tiennent beaucoup à être notifiés le plus rapidement possible de la réception d'un nouveau message. La plupart de ces méthodes impliquent de vérifier périodiquement la présence de nouveaux messages. Une méthode poussée traitée par l'agent de livraison final donne aux utilisateurs une notification plus rapide et épargne les ressources du serveur. Le présent document ne spécifie pas la méthode de notification, mais il est espéré qu'en utilisant l'infrastructure existante de messagerie instantanée comme les messages du service de messages courts (SMS, *Short Message Service*) du protocole extensible de messagerie et de présence (WMPP, *Extensible Messaging and Presence Protocol*) ou du système mondial de communications mobiles (GSM, *Global System for Mobile Communications*) cela deviendra populaire. Le présent document décrit une extension au langage de filtrage de messagerie Sieve qui permet aux utilisateurs de donner des règles spécifique sur quand et comment ces notifications devraient être envoyées.

Table des matières

1. Introduction.....	2
1.1 Conventions utilisées dans ce document.....	2
2. Identifiant de capacité.....	2
3. Action Notify.....	2
3.1 Syntaxe et sémantique de l'action Notify.....	2
3.2 Paramètre Notify "method".....	2
3.3 Étiquette Notify ":from".....	3
3.4 Étiquette Notify ":importance".....	3
3.5 Étiquette Notify ":options".....	3
3.6 Étiquette Notify ":message".....	3
3.7 Exemples.....	4
3.8 Exigences pour les spécifications de méthodes de notification.....	5
4. Essai valid_notify_method.....	6
5. Essai notify_method_capability.....	6
6. Modificateur encodeur à l'action "set".....	7
7. Interactions avec d'autres actions Sieve.....	7
8. Considérations sur la sécurité.....	7
9. Considérations relatives à l'IANA.....	9
9.1 Enregistrement de l'extension Sieve.....	9
9.2 Nouveau registre pour les mécanismes de notification Sieve.....	9
9.3 Nouveau registre pour les paramètres de capacité de notification.....	9
10. Remerciements.....	10
11. Références.....	10
11.1 Références normatives.....	10
11.2 Références pour information.....	10

Adresse des auteurs.....11

1. Introduction

Ceci est une extension au langage Sieve défini par la [RFC5228] pour fournir des notifications instantanées. Il définit la nouvelle action "notify".

Le présent document ne spécifie pas les méthodes de notification. Des exemples de possibles méthodes de notification sont la messagerie électronique et XMPP. Pour permettre la portabilité des scripts qui utilisent des notifications, la mise en œuvre de la méthode de la [RFC5436] est obligatoire. D'autres méthodes disponibles vont dépendre de la mise en œuvre et de la configuration du système.

1.1 Conventions utilisées dans ce document

Les conventions de notation sont celles de la Section 11 de la [RFC5228], incluant l'utilisation de la [RFC5234].

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].

2. Identifiant de capacité

La chaîne de capacités associée à l'extension définie dans ce document est "enotify".

3. Action Notify

3.1 Syntaxe et sémantique de l'action Notify

```
Usage : notify [":from" string]
         [":importance" <"1" / "2" / "3">]
         [":options" string-list]
         [":message" string]
         <method: string>
```

L'action "notify" spécifie qu'une notification devrait être envoyée à un utilisateur. Le format de la notification est défini par la mise en œuvre et est aussi affecté par la méthode de notification utilisée (voir le paragraphe 3.2). Cependant, tout le contenu spécifié dans le paramètre "message" DEVRAIT être inclus.

3.2 Paramètre Notify "method"

Le paramètre de position "method" identifie la méthode de notification qui va être utilisée ; c'est un URI [RFC3986]. Par exemple, la méthode de notification peut être un URI tel [RFC3966] avec un numéro de téléphone pour envoyer des SMS, ou un URI XMPP [RFC3920] contenant un identifiant XMPP [RFC5122].

Les valeurs d'URI prises en charge vont être spécifiques du site, mais la prise en charge de la méthode de la [RFC5436] est EXIGÉE afin d'assurer l'interopérabilité. Si un schéma d'URI est spécifié que ne prend pas en charge la mise en œuvre, la notification DOIT causer une condition d'erreur au démarrage. Les scripts Sieve peuvent vérifier les méthodes acceptées en utilisant l'essai valid_notify_method pour être sûrs qu'ils utilisent seulement celles qui sont prises en charge, pour éviter de telles conditions d'erreur.

Si le paramètre "method" contient un schéma d'URI supporté, alors l'URI DOIT être vérifié quant à sa validité syntaxique. Une syntaxe d'URI invalide ou une extension d'URI non prise en charge DOIT causer une erreur. Une mise en œuvre PEUT appliquer d'autres restrictions sémantiques sur les URI -- par exemple, de restreindre les numéros de téléphone dans un URI tel: à une zone géographique particulière -- et va traiter les violations de telles restrictions sémantiques comme des erreurs.

3.3 Étiquette Notify ":from"

Une étiquette ":from" peut être utilisée pour spécifier l'auteur de la notification. La syntaxe de la valeur de ce paramètre est spécifique de la méthode. Les mises en œuvre DEVRAIENT vérifier la syntaxe en accord avec la spécification de la méthode de notification et générer une erreur quand une étiquette ":from" syntaxiquement invalide est spécifiée.

Afin de minimiser/empêcher la falsification de la valeur d'auteur, les mises en œuvre DEVRAIENT imposer des restrictions aux valeurs qui peuvent être spécifiées dans une étiquette ":from". Par exemple, une mise en œuvre peut restreindre cette valeur à être un membre d'une liste d'adresses d'auteurs connus ou à appartenir à un domaine particulier. Il est suggéré que les valeurs qui ne satisfont pas à ces restrictions soient simplement ignorées plutôt que de causer l'échec de l'action "notify".

3.4 Étiquette Notify ":importance"

L'étiquette ":importance" spécifie l'importance de la livraison rapide de la notification, comme elle est perçue par le propriétaire du script Sieve. L'étiquette ":importance" est suivie d'une valeur numérique représentée par une chaîne : "1" (haute importance), "2" (importance normale), et "3" (faible importance). Si aucune importance n'est donnée, la valeur par défaut de "2" DEVRAIT être supposée. Une méthode de notification PEUT traiter la valeur d'importance comme un indicateur de transport. Par exemple, elle pourrait livrer les notifications de haute importance plus vite que les notifications d'importance normale ou faible. Certaines méthodes de notification permettent aux utilisateurs de spécifier leur état d'activité (par exemple, "occupé" ou "loin du clavier"). Si la méthode de notification fournit cette information, elle DEVRAIT être utilisée pour envoyer sélectivement les notifications. Si, par exemple, l'utilisateur se marque comme "occupé", une méthode de notification peut exiger qu'une notification d'importance "3" ne soit pas envoyée ; cependant, l'utilisateur pourrait être notifié d'une notification d'importance plus élevée.

Si la méthode de notification permet aux utilisateurs de filtrer les messages sur la base de certains paramètres dans le message, les utilisateurs DEVRAIENT être capables de filtrer sur la base de l'importance. Si la méthode de notification ne prend pas en charge l'importance, alors ce paramètre DOIT être ignoré. Une mise en œuvre PEUT inclure la valeur d'importance dans le message par défaut (paragraphe 3.6) si une n'est pas fournie.

3.5 Étiquette Notify ":options"

L'étiquette ":options" est utilisée pour envoyer des paramètres supplémentaires de la méthode de notification. L'interprétation des paramètres est spécifique de la méthode. Le présent document ne spécifie aucun paramètre supplémentaire.

Chaque chaîne dans la liste des chaînes d'options a la syntaxe suivante : "<optionname>=<value>" où optionname a l'ABNF [RFC5234] suivant :

```

l-d = ALPHA / DIGIT
l-d-p = l-d / "." / "-" / "_"
optionname = l-d *l-d-p
value = *(%x01-09 / %x0B-0C / %x0E-FF)

```

3.6 Étiquette Notify ":message"

L'étiquette ":message" spécifie les données du message à inclure dans la notification. La totalité de la chaîne DEVRAIT être envoyée, mais les mises en œuvre PEUVENT abréger le message pour des raisons techniques ou esthétiques. Si le paramètre ":message" est absent, un message par défaut spécifique de la mise en œuvre est utilisé. Sauf spécification contraire d'un mécanisme de notification particulier, une mise en œuvre va par défaut contenir au moins la valeur du champ d'en-tête "From" et la valeur du champ d'en-tête "Subject" est RECOMMANDÉE.

Pour construire des messages plus complexes, l'extension notify peut être utilisée avec l'extension Sieve variables [RFC5229], comme montré dans les exemples ci-dessous.

3.7 Exemples

Exemple 1 :

```
require ["enotify", "fileinto", "variables"];
```

```

if header :contains "from" "boss@exemple.org" {
  notify :importance "1"
    :message "Ceci est probablement très important"
      "mailto:alm@exemple.com";
  # Ne pas envoyer d'autres notifications
  stop;
}

if header :contains "to" "sievemailinglist@exemple.org" {
  # :matches est utilisé pour obtenir la valeur de l'en-tête Subject
  if header :matches "Subject" "*" {
    set "subject" "${1}";
  }

  # :matches est utilisé pour obtenir la valeur de l'en-tête From
  if header :matches "From" "*" {
    set "from" "${1}";
  }

  notify :importance "3"
    :message "[SIEVE] ${from}: ${subject}"
      "mailto:alm@exemple.com";
  fileinto "INBOX.sieve";
}

```

Exemple 2 :

```

require ["enotify", "fileinto", "variables", "envelope"];

if header :matches "from" "*@*.exemple.org" {
  # :matches est utilisé pour obtenir l'adresse de MAIL FROM
  if envelope :all :matches "from" "*" {
    set "env_from" "[really: ${1}]";
  }

  # :matches est utilisé pour obtenir la valeur de l'en-tête Subject
  if header :matches "Subject" "*" {
    set "subject" "${1}";
  }

  # :matches est utilisé pour obtenir l'adresse de l'en-tête From
  if address :matches :all "from" "*" {
    set "from_addr" "${1}";
  }

  notify :message "${from_addr}${env_from}: ${subject}"
    "mailto:alm@exemple.com";
}

```

Exemple 3 :

```

require ["enotify", "variables"];

set "notif_method"
"xmpp:tim@exemple.com?message;subject=SIEVE;body=You%20got%20mail";

if header :contains "subject" "Your dog" {
  set "notif_method" "tel:+14085551212";
}

if header :contains "to" "sievemailinglist@exemple.org" {
  set "notif_method" "";
}

```

```

}

if not string :est "${notif_method}" "" {
    notify "${notif_method}";
}

if header :contains "from" "boss@exemple.org" {
    # :matches est utilisé pour obtenir la valeur de l'en-tête Subject
    if header :matches "Subject" "*" {
        set "subject" "${1}";
    }

    # Je n'ai pas besoin de notification de haute importance pour votre information
    if not header :contains "subject" "FYI:" {
        notify :importance "1" :message "BOSS: ${subject}"
            "tel:+14085551212";
    }
}
}

```

3.8 Exigences pour les spécifications de méthodes de notification

Ce paragraphe décrit les exigences pour les documents qui définissent des méthodes de notification spécifiques de Sieve.

Les mécanismes de notification NE DOIVENT PAS ajouter de nouvelles étiquettes Sieve à l'action "notify".

Une méthode de notification PEUT permettre la modification du texte final de notification -- par exemple, en le tronquant si il excède une limite de longueur ou en modifiant des caractères qui ne peuvent pas être représentés dans le jeu de caractères cible. Les caractères dans le texte de notification qui ne peuvent pas être représentés par la méthode de notification DEVRAIENT être remplacés par un symbole indiquant un caractère inconnu. Les modifications permises DOIVENT être documentées dans la description de la méthode de notification.

Une méthode de notification PEUT ignorer les paramètres spécifiés dans l'action "notify".

Une méthode de notification PEUT recommander la valeur de message par défaut à utiliser si l'argument ":message" n'est pas spécifié.

Les notifications DEVRAIENT inclure des horodatages, si la méthode de notification permet leur transmission en dehors du message textuel. Les méthodes de mise en œuvre qui peuvent seulement transmettre des horodatages dans le message textuel PEUVENT les inclure dans le message textuel.

Une notification DOIT inclure des moyens d'identifier/tracer son origine afin de permettre à un receveur d'arrêter les notifications ou de trouver comment contacter l'expéditeur. Cette exigence est destinée à aider à tracer une origine de notifications mal configurée ou abusive.

Les méthodes NE DEVRAIENT PAS inclure des autres informations étrangères non spécifiées dans les paramètres de l'action "notify".

Les méthodes DOIVENT spécifier quels paramètres d'URI (si il en est) doivent être ignorés, lesquels doivent être utilisés dans la notification résultante, et lesquels doivent causer une erreur.

Les méthodes DOIVENT spécifier quelles valeurs sont retournées par l'essai `notify_method_capability` (Section 5) en particulier pour la capacité de notification "online".

Si il y a des erreurs dans l'envoi de la notification, l'interpréteur Sieve DEVRAIT ignorer la notification et ne pas la réessayer indéfiniment. L'interpréteur Sieve PEUT réduire les notifications ; si il le fait, une demande d'envoi d'une notification PEUT être ignorée en silence. Les documents qui décrivent les méthodes de notification DEVRAIENT décrire comment les réessais, la réduction de taux, la suppression de doublés (si il en est) etc. sont à traiter par les mises en œuvre.

4. Essai `valid_notify_method`

Usage : `valid_notify_method` <notification-uris: string-list>

L'essai `valid_notify_method` est vrai si les méthodes de notification mentionnées dans l'argument `notification-uris` sont pris en charge et si ils sont valides syntaxiquement (y compris les paramètres d'URI) et sémantiquement (y compris les restrictions de sémantique spécifiques de la mise en œuvre). Cet essai DOIT effectuer exactement la même validation que celle qui serait effectuée sur le paramètre "method" de l'action "notify".

L'essai n'est vrai que si TOUTES les méthodes de notification de la liste sont prises en charge et valides.

Exemple 4 (partiel) :

```
if not valid_notify_method ["mailto:",
    "http://gw.exemple.net/notify?test"] {
    stop;
}
```

5. Essai `notify_method_capability`

Usage : `notify_method_capability` [COMPARATOR] [MATCH-TYPE]
 <notification-uri: string>
 <notification-capability: string>
 <key-list: string-list>

L'essai `notify_method_capability` restitue la capacité de notification spécifiée par la chaîne `notification-capability` qui est spécifique du `notification-uri` et la confronte aux valeurs spécifiées dans la `key-list`. L'essai réussit si une correspondance apparaît. Le type de confrontation est par défaut ":", et le comparateur par défaut est "i;ascii-casemap".

Le paramètre `notification-capability` est insensible à la casse.

L'essai `notify_method_capability` DOIT échouer inconditionnellement si l'URI de notification spécifié est syntaxiquement invalide (comme déterminé par l'essai `valid_notify_method`, Section 4) ou spécifie une méthode de notification non prise en charge. Cependant cela NE DOIT PAS causer une erreur.

L'essai `notify_method_capability` DOIT échouer inconditionnellement si l'élément `notification-capability` spécifié n'est pas connu de l'interpréteur Sieve. Un script NE DOIT PAS échouer avec une erreur si l'élément n'existe pas. Cela permet que soient écrits des scripts qui traitent en douceur les éléments non existants

Le présent document définit une seule valeur de `notification-capability` "online", qui est décrite ci-dessous. Des valeurs de capacité de notification supplémentaires peuvent être définies en utilisant la procédure définie au paragraphe 9.3.

L'extension "relational" [RFC5231] ajoute un type de correspondance appelé ":", et le compte d'un essai `notify_method_capability` est 0, si l'information retournée est la chaîne vide, ou 1.

Pour la `notification-capability` "online", l'essai `notify_method_capability` peut correspondre à une des valeurs de `key-list` suivantes :

- o "oui" - l'entité identifiée par le `notification-uri` peut recevoir immédiatement une notification notify. Noter que même après que cette valeur est retournée, il n'est pas garanti que l'entité va réellement être capable de recevoir une notification immédiatement ou même la recevoir du tout. Des erreurs de transport, la politique du receveur, etc. peuvent l'empêcher.
- o "non" - l'entité identifiée par le `notification-uri` n'est actuellement pas disponible pour recevoir une notification immédiate.
- o "peut-être" - l'interpréteur Sieve ne peut pas déterminer si l'entité identifiée par le `notification-uri` est en ligne ou non.

Exemple 5 :

```
require ["enotify"];

if notify_method_capability
  "xmpp:tim@exemple.com?message;subject=SIEVE"
  "Online"
  "oui" {
  notify :importance "1" :message "Vous avez un message"
    "xmpp:tim@exemple.com?message;subject=SIEVE";
  } else {
  notify :message "Vous avez un message" "tel:+14085551212";
  }
}
```

6. Modificateur encodeurl à l'action "set"

Usage : ":encodeurl"

Quand le script Sieve spécifie à la fois les capacités "variables" [RFC5229] et "enotify" dans le "require", un nouveau modificateur d'action "set" (voir la [RFC5229]) ":encodeurl" devient disponible pour les scripts Sieve. Ce modificateur effectue un codage en pourcentage de tout octet de la chaîne qui n'appartient pas à l'ensemble "unreserved" (voir la [RFC3986]). La procédure de codage en pourcentage est décrite dans la [RFC3986].

Le modificateur ":encodeurl" a la préséance 15.

Exemple 6 :

```
require ["enotify", "variables"];

set :encodeurl "body_param" "Safe body&evil=evilbody";

notify "mailto:tim@exemple.com?body=${body_param}";
```

7. Interactions avec d'autres actions Sieve

L'action "notify" est compatible avec toutes les autres actions, et n'affecte pas le fonctionnement des autres actions. En particulier, l'action "notify" NE DOIT PAS annuler la garde implicite.

Il est permis d'exécuter plusieurs actions "notify". Des méthodes de notification spécifiques PEUVENT permettre que de multiples notifications provenant du même script soient fusionnées en une seule.

8. Considérations sur la sécurité

Les considérations sur la sécurité sont discutées dans la [RFC5228]. De plus, les mises en œuvre doivent veiller à suivre les considérations sur la sécurité des méthodes de notification spécifiques.

L'action "notify" est potentiellement très dangereuse. Le chemin que prend la notification à travers le réseau peut n'être pas sûr. Une erreur dans la chaîne d'options peut causer la transmission du message à quelqu'un à qui il n'est pas destiné, ou peut exposer des informations à des espions.

Le fait qu'une notification soit reçue ne signifie pas qu'elle a été envoyée par la mise en œuvre de Sieve. Il serait possible de falsifier des notifications ou de modifier des parties de notifications valides avec certaines méthodes de notification.

La falsification de la valeur de ":importance" (par exemple, par une modification non autorisée du script) peut résulter en un ralentissement de la livraison de notification.

Noter que certains composants de notifications ne devraient pas être crus aveuglément. Par exemple, le champ Horodatage peut être facilement falsifié ou modifié quand certains transports de notification sont utilisés. Même si l'horodatage est

estimé être correct par l'expéditeur et n'est pas modifié dans le transit, il pourrait être trompeur pour le système receveur du fait des différences d'horloges.

Une organisation peut avoir une politique de transmission d'informations classifiées à des réseaux non classifiés. Sauf si la politique est aussi appliquée dans le module chargé de la génération (ou de l'envoi) des notifications, les utilisateurs peuvent utiliser l'extension définie dans le présent document pour extraire les informations classifiées et contourner la politique.

Les notifications peuvent résulter en boucles et rebonds. Aussi, permettre à un seul script de notifier plusieurs destinations peut être utilisé comme moyen d'amplifier le nombre de messages dans une attaque. De plus, si la détection de boucle n'est pas mise en œuvre de façon appropriée, il est possible d'établir des boucles de notification à croissance exponentielle. En conséquence, les méthodes de notification Sieve :

1. DOIVENT fournir des mécanismes pour éviter les boucles de notifications.
2. DOIVENT fournir des moyens pour que les administrateurs limitent la capacité des utilisateurs d'abuser des notify. En particulier, il DOIT être possible de limiter le nombre d'actions "notify" qu'un script peut effectuer. De plus, si il n'existe pas de cas d'utilisation de l'usage de "notify" avec plusieurs destinations, cette limite DEVRAIT être réglée à 1. Des limites supplémentaires, comme la capacité de restreindre "notify" aux utilisateurs locaux, PEUVENT aussi être mises en œuvre.
3. DOIVENT fournir des facilités d'enregistrer l'utilisation de "notify" afin de faciliter le traçage des abus.
4. PEUVENT utiliser l'analyse de script pour déterminer si un certain script peut ou non être exécuté en toute sécurité. Bien que le langage Sieve soit suffisamment complexe pour que l'analyse complète de tous les scripts possibles soit computationnellement infaisable, la majorité des scripts sont en réalité ouverts à l'analyse. Par exemple, une mise en œuvre pourrait permettre que les scripts qu'elle a déterminés comme sûrs fonctionnent sans entrave, qu'elle bloque les scripts qui sont potentiellement problématiques, et qu'elle soumette les scripts inclassables à un examen et enregistrement supplémentaires.

Une action "notify" peut n'être pas appropriée du tout dans des situations où les scripts Sieve sont associés à des comptes de messagerie qui sont librement disponibles et/ou non traçables jusqu'à une personne qui puisse être tenue pour responsable de la création d'explosions de messages ou autres abus.

Les mises en œuvre qui construisent des URI en interne à partir de divers paramètres notify DOIVENT s'assurer que tous les composants de tels URI sont proprement codés en pourcentage (voir la [RFC3986]). En particulier, cela s'applique aux valeurs des arguments étiquetés "from" et "message" et peut s'appliquer aux valeurs "options".

Les essais d'en-tête/enveloppe [RFC5228], avec les variables Sieve, peuvent être utilisés pour extraire la liste des utilisateurs qui vont recevoir des notifications provenant du message entrant ou de son enveloppe. Cela est potentiellement assez dangereux, car cela peut être utilisé pour des attaques de déni de service sur les receveurs contrôlés par l'expéditeur du message. Pour cette raison, les mises en œuvre NE DEVRAIENT PAS pas permettre l'utilisation de variables contenant des valeurs extraites du message dans le paramètre "method" à l'action "notify". Noter qu'une violation de cela NE DEVRAIT PAS peut résulter en la création d'un relais ouvert, c'est-à-dire, où tout expéditeur serait capable de créer des messages spécialement étudiés qui résulteraient en notifications livrées aux receveurs sous le contrôle de l'expéditeur. Dans le pire des cas, il pourrait en résulter une perte financière pour l'utilisateur qui contrôle le script Sieve et/ou pour les receveurs de notifications (par exemple, si une notification est un message SMS).

Noter que ce dernier "NE DEVRAIENT PAS" n'est pas une interdiction générique de l'utilisation de variables dans l'action "notify", car le contrôle de la cible d'une notification en l'extrayant d'un magasin de données en possession de l'utilisateur (comme une entrée du protocole léger d'accès à un répertoire (LDAP, *Lightweight Directory Access Protocol*) de l'utilisateur) est considéré comme utile.

Il est impératif que quelle que soit la méthode qu'utilisent les mises en œuvre pour mémoriser les scripts de filtrage définis par l'utilisateur pour se protéger de modifications non autorisées, elle préserve l'intégrité du système de messagerie. Un attaquant qui peut modifier un script peut causer l'élimination du message, son rejet ou sa transmission à un receveur non autorisé. De plus, il est possible que les scripts Sieve exposent des informations privées, comme des noms de boîte aux lettres ou des adresses de messagerie de correspondants favorisé (ou défavorisés). À cause de cela, les scripts DEVRAIENT aussi être protégés contre une restitution non autorisée.

9. Considérations relatives à l'IANA

9.1 Enregistrement de l'extension Sieve

Pour : iana@iana.org

Objet : enregistrement d'une nouvelle extension à Sieve

Nom de capacité : `enotify`

Description : ajoute l'action "notify" pour notifier à l'utilisateur la réception du message. Elle fournit aussi deux nouveaux essais : `valid_notify_method` vérifie la validité des URI de notification ; `notify_method_capability` peut vérifier les capacités des receveurs.

RFC publiée : RFC 5435

Adresse de contact : liste de diffusion Sieve <ietf-mta-filters@imc.org>

Ces informations ont été ajoutées à la liste des extensions à Sieve disponibles sur <http://www.iana.org/>.

9.2 Nouveau registre pour les mécanismes de notification Sieve

L'IANA a créé un nouveau registre pour les mécanismes de notification Sieve. Ce registre contient les noms des mécanismes de notification contrôlés par le vendeur (commençant par "vnd.") et les noms des mécanismes de notification contrôlés par l'IETF. Les noms des mécanismes de notification contrôlés par le vendeur ont le format défini dans le paragraphe suivant et peuvent être enregistrés selon le principe du "premier arrivé, premier servi" [RFC5226], sur demande à l'IANA avec le formulaire spécifié plus loin. L'enregistrement des mécanismes de notification qui ne commencent pas par "vnd." se fait en utilisant la politique de "spécification requise" [RFC5226].

Les noms des mécanismes de notification contrôlés par le vendeur DOIVENT avoir la forme "vnd.<nom de vendeur>.<nom de mécanisme>", où <nom de vendeur> est comme spécifié dans le registre de sous arborescence de fabricant du protocole d'accès à la configuration d'application (ACAP, *Application Configuration Access Protocol*) [RFC2244].

Voici la définition du gabarit d'un nouveau registre pour les mécanismes de notification Sieve, qui a été créé et est disponible sur <http://www.iana.org/>. Il n'y a pas d'entrées initiales pour ce registre.

Pour : iana@iana.org

Objet : enregistrement d'un nouveau mécanisme de notification Sieve

Nom du mécanisme : [le nom du mécanisme]

URI de mécanisme : [numéro de RFC du document qui définit l'URI utilisé par ce mécanisme. Des mécanismes différents DOIVENT utiliser un schéma d'URI différent.]

Options spécifiques du mécanisme : [nom de toutes les options Sieve notify (comme utilisées dans le paramètre ":options") qui sont spécifiques de ce mécanisme, ou "aucune"]

Référence permanente et directement disponible : [le numéro de RFC ou un URL du document qui définit ce mécanisme de notification].

Personne et adresse de messagerie à contacter pour plus d'informations: [le nom et l'adresse de messagerie du contact technique pour des informations sur ce mécanisme]

9.3 Nouveau registre pour les paramètres de capacité de notification

L'IANA a créé un nouveau registre pour les paramètres notification-capability de l'essai `notify_method_capability`. Ce registre contient les valeurs de capacité de notification contrôlées par le vendeur (commençant par "vnd.") et les valeurs de capacité de notification contrôlées par l'IETF. Les valeurs de capacité de notification contrôlées par le vendeur qui ont le format défini dans l'alinéa suivant peuvent être enregistrées selon le principe du "premier arrivé premier servi" [RFC5226], sur demande à l'IANA avec le formulaire spécifié plus loin. L'enregistrement des valeurs de capacité de notification qui ne commencent pas par "vnd." est effectué en utilisant la politique de "spécification requise" [RFC5226].

Les valeurs de capacité de notification contrôlées par le vendeur DOIVENT avoir la forme "vnd.<nom de vendeur>.<nom de capacité>", où <nom de vendeur> est comme spécifié dans le registre Sous arborescence de vendeur ACAP [RFC2244].

Le gabarit suivant doit être utilisé pour enregistrer les paramètres de capacité de notification :

Pour : iana@iana.org

Objet : enregistrement d'un nouveau paramètre notification-capability

Nom de capacité : [nom de la capacité de notification]

Description: [une explication de l'objet de la capacité de notification]

Syntaxe : [définition formelle des valeurs permises et de leur syntaxe]

Référence permanente et directement disponible : [numéro de RFC ou URL du document qui définit ce mécanisme de notification]

Informations de contact : [le nom et l'adresse de messagerie du contact technique pour des informations sur ce mécanisme]

Voici le formulaire d'enregistrement pour la capacité de notification "online" :

Pour : iana@iana.org

Objet : enregistrement d'un nouveau paramètre notification-capability

Nom de capacité : online

Description : retourne si l'entité identifiée par le paramètre notification-uri à l'essai `notify_method_capability` peut recevoir immédiatement une notification `notify`.

Syntaxe : peut contenir une des trois valeurs : "oui", "non", et "peut-être". les valeurs DOIVENT être en minuscules.

Référence permanente et directement disponible : RFC 5435

Informations de contact : liste de diffusion Sieve à <ietf-mta-filters@imc.org>

10. Remerciements

Merci à Larry Greenfield, Sarah Robeson, Tim Showalter, Cyrus Daboo, Nigel Swinson, Kjetil Torgrim Homme, Michael Haardt, Mark E. Mallett, Ned Freed, Lisa Dusseault, Dilyan Palauzov, Arnt Gulbrandsen, Peter Saint-André, Sean Turner, Cullen Jennings, et Pasi Eronen pour leur aide à ce document.

11. Références

11.1 Références normatives

- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC3986] T. Berners-Lee, R. Fielding et L. Masinter, "[Identifiant de ressource uniforme](#) (URI) : Syntaxe générique", STD 66, janvier 2005. (P.S. ; MàJ par [RFC8820](#))
- [RFC5228] P. Guenther et autres, "[Sieve : un langage de filtrage](#) de messagerie électronique", janvier 2008. (P.S. ; Remplace [RFC3028](#), MàJ par [RFC5229](#), [5429](#) [9042](#))
- [RFC5229] K. Homme, "[Filtrage de messagerie Sieve](#) : extension Variables", janvier 2008. (MàJ par [RFC5173](#)) (P.S.)
- [RFC5231] W. Segmuller, B. Leiba, "[Filtrage de messagerie Sieve](#) : extension Relational", janvier 2008. (Remplace [RFC3431](#))(P.S.)
- [RFC5234] D. Crocker, P. Overell, "[BNF augmenté pour les spécifications de syntaxe](#) : ABNF", janvier 2008. ([STD0068](#))
- [RFC5436] B. Leiba, M. Haardt, "[Mécanisme de notification Sieve](#) : mailto", janvier 2009. (MàJ [RFC3834](#)) (P.S.)

11.2 Références pour information

- [RFC2244] C. Newman, J. G. Myers, "[ACAP – Protocole d'accès à la configuration d'application](#)", novembre 1997. (P.S.)
- [RFC3920] P. Saint-Andre, éd., "[Protocole extensible de messagerie](#) et de présence (XMPP) : éléments centraux", octobre 2004. (P.S.) (Remplacée par [RFC6120](#))
- [RFC3966] H. Schulzrinne, "[L'URI tel pour les numéros de téléphone](#)", décembre 2004. (MàJ par [RFC5341](#)) (P.S.)

- [RFC5122] P. Saint-André, "[Identifiants de ressource internationalisée \(IRI\)](#) et identifiants de ressource uniformes (URI) pour le protocole extensible de messagerie et de présence (XMPP)", février 2008. (Remplace [RFC4622](#)) (P.S.)
- [RFC5226] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, mai 2008. (Remplace [RFC2434](#) ; remplacée par [RFC8126](#))

Adresse des auteurs

Alexey Melnikov
Isode Limited
5 Castle Business Village
36 Station Road
Hampton, Middlesex TW12 2BX
UK
mél : Alexey.Melnikov@isode.com

Wolfgang Segmuller
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532
US
téléphone : +1 914 784 7408
mél : werewolf@us.ibm.com

Barry Leiba
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532
US
téléphone : +1 914 784 7941
mél : leiba@watson.ibm.com

Tim Martin
Endless Crossword
672 Haight st.
San Francisco, CA 94117
US
téléphone : +1 510 260-4175
mél : timmartin@alumni.cmu.edu