

Groupe de travail Réseau  
**Request for Comments : 5232**  
 Catégorie : Sur la voie de la normalisation

A. Melnikov, Isode Limited  
 janvier 2008  
 Traduction Claude Brière de L'Isle

## Filtrage de messagerie Sieve : extension Imap4flags

### Statut du présent mémoire

Le présent document spécifie un protocole 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 "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.

### Résumé

Des discussions récentes ont montré qu'il est souhaitable d'établir des fanions IMAP (RFC 3501) différents à la livraison du message. Cela peut être fait, par exemple, par un interpréteur Sieve qui fonctionne au titre d'un agent de livraison de messagerie.

Le présent document décrit une extension au langage de filtrage de messagerie Sieve pour établir des fanions IMAP. L'extension permet d'établir des fanions systèmes IMAP et des mots clés IMAP.

### Table des Matières

1. Introduction.....	1
1.1 Conventions utilisées.....	2
2. Exigences générales pour le traitement des fanions.....	2
3. Actions.....	2
3.1 Action setflag.....	3
3.2 Action addflag.....	3
3.3 Action removeflag.....	3
4. Essai hasflag.....	4
5. Argument étiqueté :flags.....	4
6. Interaction avec les autres actions Sieve.....	5
7. Considérations sur la sécurité.....	5
8. Considérations relatives à l'IANA.....	5
9. Exemple étendu.....	6
10. Remerciements.....	6
11. Références normatives.....	7
Adresse de l'auteur.....	7
Déclaration complète de droits de reproduction.....	7

## 1. Introduction

Cette extension au langage Sieve défini dans la [RFC5228] établit des fanions [RFC3501]. Elle ajoute un nouvel argument étiqueté à "keep" et "fileinto" qui décrit la liste des fanions qui doivent être établis quand le message est livré à la boîte aux lettres spécifiée. Elle ajoute aussi plusieurs actions pour aider à manipuler la liste des fanions et un essai pour vérifier si un fanion appartient à une liste.

Du point de vue de l'utilisateur, cette extension fournit plusieurs capacités. D'abord, elle permet la manipulation d'ensembles de fanions IMAP. Ensuite, elle donne la capacité d'associer un ensemble de fanions IMAP à un message qui est livré à un magasin de messages en utilisant les actions "keep/fileinto". Enfin, elle conserve une variable interne. La variable interne contient les fanions par défaut qui vont être associés à un message livré en utilisant les actions "keep", "implicit keep", ou "fileinto", quand l'argument étiqueté ":flags" n'est pas spécifié. Quand l'extension Sieve [RFC5229] est aussi prise en charge par le moteur Sieve, elle permet la mise en œuvre de plusieurs variables contenant des ensembles de fanions IMAP.

La chaîne "capability" associée à l'extension définie dans le présent document est "imap4flags". Toutes les mises en œuvre conformes DOIVENT mettre en œuvre toutes les actions Sieve (setflag, addflag, removeflag), l'essai "hasflag", et l'argument étiqueté ":flags" décrits dans le présent document.

L'extension "imap4flags" peut être utilisée avec ou sans l'extension "variables" de la [RFC5229]. Quand l'extension "variables" est activée dans un script utilisant <require "variables">, le script peut utiliser des noms de variable explicite dans les actions setflag/addflag/removeflag et l'essai "hasflag". Voir aussi la Section 3 pour les détails. Quand l'extension "variables" n'est pas activée, le paramètre Nom de variable explicite à setflag/addflag/removeflag/hasflag NE DOIT PAS être utilisé et DOIT causer une erreur conformément à la [RFC5228].

### 1.1 Conventions utilisées

Les conventions de notation sont comme au paragraphe 1.1 de la [RFC5228], incluant l'utilisation de l'étiquette "Usage:" pour la définition de l'action et la syntaxe d'arguments étiquetés.

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. Exigences générales pour le traitement des fanions

Les notes suivantes s'appliquent au traitement des actions addflag/removeflag/setflag, l'essai "hasflag" et l'argument étiqueté ":flags".

Un interpréteur Sieve DOIT ignorer les chaînes vides (c'est-à-dire, "") dans un paramètre "liste-de-fanions".

Une chaîne contenant une liste séparée par des espaces de noms de fanions est équivalente à une liste de chaînes de fanions. Cette exigence simplifie l'amalgame de plusieurs listes de fanions.

L'interpréteur Sieve DEVRAIT vérifier la validité de la liste des fanions comme décrit par l'ABNF de la [RFC3501]. En particulier, selon la [RFC3501], les caractères non ASCII ne sont pas permis dans les noms de fanions. Cependant, des espaces DOIVENT toujours être permises comme délimiteurs dans les chaînes qui représentent une liste de fanions. Dans ces chaînes, plusieurs espaces entre les noms de fanions DOIVENT être traitées comme un seul caractère espace, et les espaces en-tête et en queue DOIVENT être ignorées.

Si une vérification de validité de fanion échoue, le fanion DOIT être ignoré.

Noter qu'il n'est pas possible d'utiliser cette extension pour établir ou supprimer le fanion "\Recent" ou tout autre fanion spécial de système qui n'est pas réglable dans la [RFC3501]. Tous ces fanions DOIVENT être ignorés si ils sont inclus dans une liste de fanions.

## 3. Actions

Toutes les actions décrites dans la présente spécification (setflag, addflag, removeflag) opèrent sur des variables de chaîne qui contiennent un ensemble de fanions [RFC3501]. À une substitution de variable, un ensemble de fanions est représenté comme une chaîne contenant une liste de noms de fanions séparés par des espaces.

Toute action setflag/addflag/removeflag PEUT altérer la liste de fanions de toute façon qui laisse sa sémantique comme un ensemble de mots inchangés insensibles à la casse. Par exemple, elle peut réarranger les fanions, altérer la casse des lettres qu'ils contiennent, ou ajouter ou supprimer des espaces dupliqués ou supplémentaires. Les scripts NE DOIVENT PAS faire d'hypothèses sur l'ordre des fanions dans les listes ou sur la préservation de leur casse.

Noter que le paramètre qui spécifie un nom de variable pour les actions setflag/addflag/removeflag de l'essai hasflag est facultatif. Si le paramètre n'est pas spécifié, les actions opèrent sur la variable interne, qui a la valeur vide quand le script commence l'exécution. Si l'interpréteur SIEVE ne prend pas en charge l'extension "variables" [RFC5229], la présence du paramètre Nom de variable va causer une erreur au moment de l'exécution [RFC5228].

L'action "addflag" ajoute des fanions à un ensemble existant. L'action "removeflag" supprime des fanions d'un ensemble existant. L'action "setflag" remplace un ensemble existant de fanions par un nouvel ensemble. L'action "set" définie dans la [RFC5229] peut aussi être utilisée pour remplacer un ensemble existant de fanions par un nouvel ensemble. Cependant, on devrait noter que l'action "set" ne peut pas effectuer de réarrangement de fanions, d'élimination de dupliqués, etc.

L'argument étiqueté :flags aux actions "keep" et "fileinto" est utilisé pour associer un ensemble de fanions au message en cours. Si l'argument étiqueté :flags n'est pas spécifié avec ces deux actions, la valeur courante de la variable interne est utilisée à la place. La valeur de la variable interne s'applique aussi au "keep" implicite.

Noter que quand keep/fileinto est utilisé plusieurs fois dans un script et que l'élimination de message dupliqué est effectuée (comme décrit au paragraphe 2.10.3 de la [RFC5228]) la dernière valeur de liste de fanions DOIT l'emporter.

### 3.1 Action setflag

Usage : setflag [<variablename: chaîne>] <liste de fanions: liste-de-chaînes>

Setflag est utilisé pour régler les fanions ou mots clés du système [RFC3501]. Setflag remplace tous les ensembles de fanions précédents.

Exemple :

```
si taille supérieure à 500K {
    setflag "\supprimé";
}
```

Un exemple plus substantiel est :

Exemple :

```
si en-tête :contient "from" "boss@frobnitzm.exemple.edu" {
    setflag "flagvar" "\\Flagged";
    fileinto :flags "${flagvar}" "INBOX.From Boss";
}
```

### 3.2 Action addflag

Usage : addflag [<variablename: string>] <liste-des-fanions : liste-de-chaînes>

Addflag est utilisé pour ajouter des fanions à une liste de fanions de la [RFC3501]. Il ne remplace aucun ensemble de fanions établi précédemment. Cela signifie que plusieurs occurrences de addflag sont traitées de façon additive.

Les exemples suivants montrent les exigences décrites au paragraphe 2.1. Les deux actions suivantes :

```
addflag "flagvar" "\\Deleted";
addflag "flagvar" "\\Answered";
```

produisent le même résultat que la seule action addflag "flagvar" ["\\Deleted", "\\Answered"];

ou

```
addflag "flagvar" "\\Deleted \\Answered";
```

ou

```
addflag "flagvar" "\\Answered \\Deleted";
```

### 3.3 Action removeflag

Usage : removeflag [<variablename: string>] <liste-des-fanions : liste-de-chaînes>

Removeflag est utilisé pour retirer des fanions d'une liste de fanions de la [RFC3501]. Removeflag supprime les fanions établis précédemment par "set"/"addflag". Invoquer removeflag avec un fanion qui n'était pas établi précédemment n'est pas une erreur et est ignoré. Noter que si une mise en œuvre n'effectue pas l'élimination automatique des dupliqués, elle DOIT supprimer toutes les occurrences des fanions spécifiés dans le second paramètre de removeflag. Les chaînes vides dans la liste des fanions DOIVENT être ignorées. Noter aussi que les noms de fanions sont insensibles à la casse, comme décrit dans la [RFC3501]. Plusieurs actions removeflag sont traitées de façon additive.

Exemple :

```
si l'en-tête :contient "Disposition-Notification-To"
"mel@exemple.com" {
    addflag "flagvar" "$MDNRequired";
}
si l'en-tête :contient "from" "imap@cac.washington.exemple.edu" {
```

```

removeflag "flagvar" "$MDNRequired";
fileinto :flags "${flagvar}" "INBOX.imap-list";
}

```

#### 4. Essai hasflag

Usage : hasflag [MATCH-TYPE] [COMPARATOR] [<variable-list: string-list>] <liste-des-fanions : liste de chaînes>

L'essai hasflag s'évalue à vrai si une des variables correspond à un des noms de fanion. Le type de correspondance par défaut est ":is". Si la liste des variables est omise, la valeur de la variable interne est utilisée à la place.

Le comparateur par défaut est "i;ascii-casemap", qui est la même comparaison insensible à la casse que définie pour les fanions IMAP par la [RFC3501].

L'extension "relational" de la [RFC5231] ajoute un type de correspondance appelé ":count". Le :count d'une variable retourne le nombre de fanions distincts qu'elle contient. Le compte d'une liste de variables est la somme des comptes des variables membres.

Exemple :

Si la variable interne a la valeur "A B", l'essai suivant :

```
hasflag :is "b A"
```

s'évalue à vrai. L'essai ci-dessus peut aussi s'écrire comme

```
hasflag ["b","A"]
```

Exemple :

Si la variable "MyVar" a la valeur "NonJunk Junk gnus-forward

\$Forwarded NotJunk JunkRecorded \$Junk \$NotJunk", les essais suivants s'évaluent à vrai :

```
hasflag :contains "MyVar" "Junk"
```

```
hasflag :contains "MyVar" "forward"
```

```
hasflag :contains "MyVar" ["label", "forward"]
```

```
hasflag :contains "MyVar" ["junk", "forward"]
```

Noter que le dernier de ces essais peut être écrit comme :

```
hasflag :contains "MyVar" "junk forward"
```

ou

```
hasflag :contains "MyVar" "forward junk"
```

Cependant, les deux dernières formes ne sont pas recommandées.

Et les essais suivants vont s'évaluer à faux :

```
hasflag :contains "MyVar" "label"
```

```
hasflag :contains "MyVar" ["label1", "label2"]
```

Exemple :

Si la variable "MyFlags" a la valeur de "A B", l'essai suivant :

```
hasflag :count "ge" :comparator "i;ascii-numeric" "MyFlags" "2"
```

s'évalue à vrai, car la variable contient deux fanions distincts.

#### 5. Argument étiqueté :flags

La présente spécification ajoute un nouvel argument étiqueté facultatif ":flags" qui altère le comportement des actions "keep" et "fileinto".

L'argument étiqueté :flags spécifie que les fanions fournis dans l'argument suivant devraient être établis quand fileinto/keep livre le message à la boîte aux lettres principale de la boîte aux lettres/utilisateur cible. Si l'argument étiqueté :flags n'est pas spécifié, "keep" ou "fileinto" va utiliser la valeur courante de la variable interne lors de la livraison du message à la boîte aux lettres cible.

Usage : ":flags" <liste-des-fanions : liste-de-chaînes>

La copie du message mis dans la boîte aux lettres va avoir seulement les fanions mentionnés après l'argument étiqueté :flags.

L'interpréteur Sieve DOIY ignorer tous les fanions qu'il ne peut pas mémoriser de façon permanente. Cela signifie que l'interpréteur NE DOIT PAS traiter l'échec d'une mémorisation de fanion comme un échec à exécuter le script Sieve au moment du démarrage. Par exemple, si la boîte aux lettres "INBOX.From Boss" ne peut mémoriser aucun fanion, alors :

```
fileinto :flags "\\Deleted" "INBOX.From Boss";  
et  
fileinto "INBOX.From Boss";
```

sont équivalents.

Le présent document n'impose pas la façon dont l'interpréteur Sieve va régler les fanions de la [RFC3501]. En particulier, l'interpréteur Sieve peut fonctionner comme un client IMAP ou peut avoir un accès direct au magasin de messages.

## 6. Interaction avec les autres actions Sieve

La présente extension fonctionne seulement sur le message qui est en cours de traitement par Sieve ; elle n'affecte aucun autre message généré par un effet collatéral d'une action ou de tout autre message déjà présent dans le magasin de messages.

L'extension décrite dans ce document ne change pas la garde implicite (voir le paragraphe 2.10.2 de la [RFC5228]).

## 7. Considérations sur la sécurité

Les considérations sur la sécurité sont discutées dans les [RFC3501], [RFC5228], et [RFC5229].

Cette extension ne permet pas, intentionnellement, l'établissement de fanions de la [RFC3501] sur un message arbitraire dans le magasin de messages de la [RFC3501].

On estime que la présente extension n'introduit aucun problème de sécurité supplémentaire.

## 8. Considérations relatives à l'IANA

Le gabarit suivant spécifie l'enregistrement par l'IANA des variables de l'extension Sieve spécifiée dans ce document :

```
To : iana@iana.org  
Sujet : enregistrement d'une nouvelle extension Sieve.  
Nom de capacité : imap4flags  
Description : Ajoute des actions et des essais pour manipuler les fanions IMAP.  
RFC publiée : RFC 5232  
Adresse de contact : liste de diffusion Sieve à <ietf-mta-filters@imc.org>
```

Ces informations ont été ajoutées à la liste des extensions Sieve donnée à <http://www.iana.org/assignments/sieve-extensions>.

## 9. Exemple étendu

```
# Exemple de filtre Sieve : déclarer toutes les caractéristiques facultatives ou extensions utilisées par le script. #
```

```
require ["fileinto", "imap4flags", "variables"];
```

```
# Déplacer les grands messages dans une boîte aux lettres spéciale. #
```

```

si taille :over 1M
{
  addflag "MyFlags" "Big";
  si l'en-tête :is "From" "boss@company.exemple.com"
  {
# Le message sera marqué comme "\Flagged Big" lors de son placement dans la boîte aux lettres "Big messages" #
    addflag "MyFlags" "\\Flagged";
  }
  fileinto :flags "${MyFlags}" "Big messages";
}

si en-tête :is "From" "grandma@exemple.net"
{
  addflag "MyFlags" ["\\Answered", "$MDNSent"];
# Si le message fait plus de 1 Mbits il sera marqué comme "Big \Answered $MDNSent" lors de son placement dans la boîte
aux lettres "grandma". Si le message fait moins de 1 Mbits, il sera marqué comme "\Answered $MDNSent" #
  fileinto :flags "${MyFlags}" "GrandMa";
}

# Traitement des messages provenant de listes de diffusion connues. Déplacer les messages provenant de listes de
discussion de filtre de l'IETF dans le dossier "filtre". #
si en-tête :is "Sender" "owner-ietf-mta-filters@exemple.org"
{
  régler "MyFlags" "\\Flagged $Work";
# Le message va avoir à la fois des fanions "\Flagged" et $Work. #
  garder :flags "${MyFlags}";
}

# Garder tous les messages de ou pour les gens de ma société. #
elsif anyof address :domain :is ["From", "To"] "company.exemple.com"
{
  garder :flags "${MyFlags}"; # garder dans le dossier "Inbox" #
}

# Essayer d'attraper les messages non sollicités. Si un message n'est pas pour moi, ou si il contient un sujet connu pour être
un pourriel, l'éliminer. #
elsif anyof (not address :all :contains
  ["To", "Cc"] "me@company.exemple.com",
  en-tête :matches "subject"
  ["*make*money*fast*", "*university*dipl*mas*"])
{
  supprimer "MyFlags" "\\Flagged";
  fileinto :flags "${MyFlags}" "spam";
}
autrement
{
  # Déplacer tous les autres messages externes dans le dossier "personnel". #
  fileinto :flags "${MyFlags}" "personnel";
}

```

## 10. Remerciements

Ce document a été révisé en partie sur la base des commentaires et discussions qui ont eu lieu sur la liste de diffusion Sieve.

On remercie de leur aide ceux qui ont pris le temps de relire le document et faire des suggestions, en particulier celles de Tim Showalter, Barry Leiba, Randall Gellens, Ken Murchison, Cyrus Daboo, Matthew Elvey, Jutta Degener, Ned Freed, Marc Mutz, Nigel Swinson, Kjetil Torgrim Homme, Mark E. Mallett, Dave Cridland, Arnt Gulbrandsen, Philip Guenther, Rob Austein, Sam Hartman, Eric Gray, et Cullen Jennings.

Un merci particulier à Tony Hansen et David Lamb de leur aide à mieux expliquer le concept.

## 11. 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](#))
- [RFC3501] M. Crispin, "Protocole d'[accès au message Internet - version 4rev1](#)", mars 2003. (P.S. ; MàJ par [RFC4466](#), [4469](#), [4551](#), [5032](#), [5182](#), [7817](#), [8314](#), [8437](#), [8474](#) ; remplacée par la [RFC9051](#))
- [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.)

## Adresse de l'auteur

Alexey Melnikov  
Isode Limited  
5 Castle Business Village  
Hampton, Middlesex  
United Kingdom, TW12 2BX

mél : [alexey.melnikov@isode.com](mailto:alexey.melnikov@isode.com)

## Déclaration complète de droits de reproduction

Copyright (C) The IETF Trust (2008).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à [www.rfc-editor.org](http://www.rfc-editor.org), et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

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

## Propriété intellectuelle

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

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

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