

**Tuteurs informatique**  
**Nous contacter**

Utiliser ce site  
Actu et stages  
Nouveautés  
Docs à imprimer  
Plan du site  
Rechercher

Les tuteurs  
Internet  
Courrier  
électronique  
Spam  
Courrier  
collectif  
International  
Mutt  
Pine  
**Procmail**  
Vider sa  
messagerie

Serveur des élèves  
ENS  
SPI

# Filtrer son courrier électronique avec procmail

- À quoi sert procmail ?
- Ce que doit contenir un .procmailrc
- Comment ça marche ?
  - Les règles de filtrage
  - Expressions régulières
  - Rediriger certains mails vers une autre adresse électronique
  - Filtrer le spam
  - Envoyer directement certains mails dans un dossier particulier
- Conclusion

## À quoi sert procmail ?

Procmail est un programme très puissant qui sert à filtrer les courriers électroniques. Grâce à lui, vous pouvez rediriger votre mail, le trier ou encore vous protéger contre le spam.

Pour donner des instructions à procmail, il vous faut créer un fichier nommé .procmailrc (dans le monde Unix, les fichiers commençant par un point et terminant par « rc » sont souvent des fichiers de configuration).



**Attention !**

Lisez bien l'**intégralité** de ce tutorial, surtout les avertissements, avant d'installer un .procmailrc. Procmail est un outil puissant et dangereux. Vous risquez de perdre des courriers importants en recopiant n'importe quoi à tort et à travers. Mieux vaut passer 5 minutes de plus à lire tout plutôt que devoir dire coucou au SPI parce que vous aurez perdu du courrier...

## Ce que doit contenir un .procmailrc

Commencez par créer un fichier nommé .procmailrc avec votre éditeur préféré, puis copiez-collez les instructions qui suivent (nous vous expliquons ci-dessous ce qu'elles veulent dire) :

```
PATH=/bin:/usr/bin
MAILDIR=$HOME/Mail
SPAMBOX=spam
```

Explication ligne par ligne :

**PATH=/bin:/usr/bin**

On indique à procmail le chemin des programmes qu'il va utiliser.

**MAILDIR=\$HOME/Mail**

On indique à procmail où se trouve votre répertoire de mail.

**SPAMBOX=spam**

Cette ligne crée une « boîte à spam » (le répertoire \$HOME/mail/spam) dans votre boîte aux lettres. C'est là que partiront les courriers suspects. Voir [infra](#).

Ces instructions de début sont la base pour faire marcher correctement procmail, mais en soi, elles ne suffisent pas. Il vous faudra spécifier, dans la suite du fichier, ce que vous voulez filtrer à l'aide de règles de filtrage. C'est ce que nous vous expliquons ci-dessous.

Vous pouvez aussi ajouter la ligne suivante :

```
LOGFILE=$MAILDIR/procmail.log
```

Ceci demande à Procmail de créer un journal (*log* en anglais) de ce qu'il fait. Ce journal est très précieux pour faire des tests, mais quand on l'oublie il prend très rapidement beaucoup de place. Songez à l'effacer régulièrement quand vous en faites un.

**Syntaxe** : les retours à la ligne sont importants. Ne recopiez pas tout à la suite. Les lignes commençant par # sont des commentaires. De manière générale, ne prenez pas d'initiative pour changer les recettes présentées ici si vous ne savez pas ce que vous faites.

## Comment ça marche ?

### Les règles de filtrage

Les « recettes » (*recipes*) sont les éléments de base du .procmailrc. Leur principe est le suivant : on commence par spécifier des conditions (une condition dit à procmail ce qu'il doit regarder dans le courrier, par exemple l'adresse de l'expéditeur, ou encore le champ « Sujet ») puis on indique l'action à accomplir si les conditions sont remplies (par exemple, rediriger le courrier, ou l'envoyer à la poubelle).

Une recette a toujours le format suivant (le # introduit un commentaire pour expliquer le code, vous n'avez pas à recopier la partie après le #) :

```
:0 [drapeau(x)] [: [fichier verrou] ]  
# Les conditions commencent chacune par *  
  * condition(s)  
  action à accomplir
```

Chaque recette s'ajoute dans le fichier .procmailrc, à la suite du bloc de début (voir [ci-dessus](#)).

### Expressions régulières

Pour spécifier les conditions, on se sert d'expressions régulières. Une expression régulière est une notation permettant de définir des schémas de recherche. Pour cela, on se sert de métacaractères, aussi appelés jokers.

Joker	Signification
^	Début de ligne
\$	Fin de ligne
.	Un caractère quelconque
[xy]	N'importe quel caractère dans l'ensemble spécifié

[^xy]	N'importe quel caractère hors de l'ensemble spécifié
foo bar	foo ou bar
c*	Un nombre quelconque (même 0) de répétitions du caractère c

Quelques exemples :

- ^From: : cette condition vise toutes les chaînes de caractère From: au début d'une ligne. On dirait une en-tête de courrier électronique, non ? :-)
- ^From:.\*choupi.\* : cette condition vise toutes les chaînes de caractère From: au début d'une ligne, suivies d'un ou plusieurs caractères, puis de choupi, puis d'un ou plusieurs autres caractères. Cette fois, on a défini l'ensemble des mails provenant d'une adresse contenant l'expression choupi : de cette façon, on vise tous les mails provenant de Choupi, votre meilleur ami.
- ^(From|Cc|To).\*choupi.\* cette fois, on vise tous les mails qui viennent de Choupi, ceux où Choupi est en Cc:, et ceux adressés à Choupi.

## Rediriger certains mails vers une autre adresse électronique

### Redirection de base

Pour rediriger votre courrier électronique vers une autre adresse, c'est simple. Ici, on redirige le courrier vers l'adresse toto@polenord.pn. Il n'y pas de condition (cela concerne tous les mails). Le point d'exclamation (!) sert à dire qu'il faut faire une redirection. Ajoutez donc à votre .procmailrc la règle suivante :

```
:0
! toto@polenord.pn
```

### Redirection avec conditions

Toujours plus fort, maintenant. Imaginons que vous soyez un fan de Scoubidou. Votre page web porte sur Scoubidou. Choupi (votre meilleur ami, souvenez-vous), vous a aidé à le réaliser. Vous souhaitez relayer les courriers qui vous arrivent et qui parlent de Scoubidou à Choupi. Avec procmail, vous pouvez le faire.

```
:0
# Concernés : tous les mails dont le sujet contient « Scoubidou »
* ^Subject:.*Scoubidou.*
# On forward à Choupi
! choupi@scoubidou-fan.com
```

Comme vous pourrez le constater, les conditions peuvent porter sur tous les champs d'un courrier électronique : From, To, CC, Bcc, etc.

Ajoutons que dans cet exemple, vous souhaiteriez sans doute vous aussi lire les courriers « Scoubidou ». Avec la recette ci-dessus, ces courriers sont simplement reexpédiés vers Choupi, et vous ne les avez jamais dans votre boîte aux lettres. Nous allons utiliser un drapeau (*flag*) qui est une sorte d'option, qui modifie le comportement standard de procmail. Il suffit d'ajouter c après le :0, ce qui demande à procmail de copier dans le répertoire de mail les courriers qui arrivent. L'exemple complet est donc le suivant :

```
:0 c
# Concernés : tous les mails dont le sujet contient « Scoubidou »
* ^Subject:.*Scoubidou.*
# On forward à Choupi
! choupi@scoubidou-fan.com
```

## Filtrer le spam

On appelle « spam » le courrier électronique envoyé en masse, non sollicité,

généralement publicitaire. Un filtre tel que procmail vous permet de lutter contre ce phénomène, mais il est plus efficace d'avoir une stratégie d'ensemble combinant logiciel anti-spam, filtre et prévention. Référez-vous à notre [page anti-spam](#) pour en savoir plus.

### Filtrer suivant le sujet

La plupart des spams sont des mails provenant d'expéditeurs bidons et différents à chaque fois, et qui ont pour point commun de vous promettre de l'argent, de vous proposer du Viagra, des diplômes par cher, et autres joyeusetés. Procmail vous permet d'identifier plus ou moins ces mails en fonction de leur sujet, et de les envoyer directement à la poubelle. De même, le filtrage par sujet permet de filtrer les mails pyramidaux (chaînes de l'amitié, fausses alertes de virus, etc.), que vous pouvez être amené à recevoir plusieurs fois.

Dans l'exemple suivant, vous allez filtrer tous les mails dont le sujet contient le mot « Viagra » en les envoyant vers votre poubelle à spam, \$HOME/mail/spam. Voir [infra](#) pour plus d'explication.

```
:0
# Sont visés tous les mails dont le sujet contient Viagra
* ^Subject: .*Viagra.*
# Et on les envoie à la poubelle à spam
$SPAMBOX
```

### Filtrer suivant l'expéditeur

Dans d'autres cas, vous pouvez très bien connaître l'adresse électronique de ceux qui vous spamment. Ce peut être une liste de diffusion à laquelle vous avez été abonné, volontairement ou non, et à laquelle les administrateurs de la liste ne vous désabonnent pas. La recette qu'il vous faut est la suivante : tous les courriers provenant de l'adresse spammeur@tagada.com seront impitoyablement envoyés dans votre boîte à spam.

```
:0
* ^From: .*spammeur@tagada.com
$SPAMBOX
```

### Filtrer suivant l'encodage

Vous recevez beaucoup de mails en provenance de Corée, qui sont un charabia illisible. Vous voudriez les filtrer. On rappelle que procmail peut filtrer sur n'importe quel en-tête de message. Si vous regardez les en-têtes d'un spam coréen (appuyez sur H, que vous utilisiez mutt ou pine, pour y avoir accès), vous verrez un champ appelé Content-type. Dans ce champ est déclaré charset="ks\_c\_5601-1987". Il vous suffit de filtrer cet encodage pour être débarrassé de la plupart des spams coréens :

```
:0
* ^Content-Type: .*charset="ks_c_5601-1987"*
$SPAMBOX
```

De manière plus perverse, certains spams coréens ne déclarent pas leur encodage au bon endroit, pour éviter justement ce genre de filtres. L'encodage est alors annoncé dans le sujet. Pour y remédier, ajoutez simplement :

```
:0
* ^Subject: .*EUC-KR.*
$SPAMBOX

:0
* ^Subject: .*ks_c_5601-1987.*
$SPAMBOX
```

Si vous êtes sûr de ne pas vouloir de mails en encodage coréen, vous pouvez mettre ces mails à la poubelle au lieu de les mettre dans la boîte à spam. Sur UNIX, la poubelle est /dev/null. Attention, contrairement à la poubelle de Windows ou MacOS, il n'y a pas de filet de sécurité, on ne peut plus récupérer les données une fois poubellisées. Ceci donne :

```
:0
* ^Subject: .*EUC-KR.*
/dev/null
```

**Attention** : la présence d'un /dev/null dans votre .procmailrc est dangereuse : la moindre erreur de syntaxe, la moindre coquille est susceptible d'envoyer tout votre mail dans /dev/null, soit le vide intersidéral d'où vous ne pouvez récupérer aucun courrier. En cas d'accident, vous pouvez vous adresser au [SPI](#), mais faites particulièrement attention en éditant votre .procmailrc.

### Trop de filtres nuit

En installant des filtres anti-spam : vous prenez le risque d'écarter des courriers qui en fait n'en sont pas, mais qui répondent tout de même à vos règles. Ce sera le cas d'un prof coréen qui vous écrit si vous filtrez les encodages coréens, d'un courrier parlant de FreeBSD si vous filtrez le mot « free », d'un courrier dont le sujet contient « bissextile » si vous filtrez « sex », etc. Plus vos règles sont strictes, moins vous recevrez de spam, mais plus vous aurez de chances de perdre du courrier.

Pour cette raison, il vaut parfois mieux envoyer les courriers suspects dans un dossier spécial, appelé « spam », par exemple. Voir ci-dessous comment faire.

## Envoyer directement certains mails dans un dossier particulier

Cette recette est particulièrement utile pour les listes de diffusion (*mailing-lists*) : vous pouvez envoyer tous les courriers provenant de la liste directement dans un dossier spécial, qui contiendra seulement les courriers de ce genre.

Admettons que vous apparteniez à la liste des amis de Scoubidou. Cette liste a pour adresse `scoubidou@zoinx.foo`, et tous les courriers de la liste ont pour champ expéditeur :

```
To: Les Amis de Scoubidou <scoubidou@zoinx.foo>
```

Vous voulez regrouper les courriers de la liste dans un dossier appelé « scoubidou ». La recette qu'il vous faut est la suivant :

```
# Le dernier ":" demande l'utilisation d'un fichier verrou
:0:
# La condition : tous les fichiers contenant 'scoubidou@zoinx.foo'
# dans le champ 'To: '
* ^To.*scoubidou@zoinx.foo
# Action : placer tous ces mails dans le dossier « Scoubidou »
$HOME/Mail/scoubidou
```

À quoi sert le fichier verrou ? Imaginons que deux courriers de la liste arrivent en même temps : deux procmails essaieront d'envoyer les deux messages dans le dossier en même temps. Si vous utilisez un fichier verrou, le premier procmail verrouillera le dossier pendant qu'il écrira son message dans le dossier, empêchant ainsi le second message d'y accéder. Une fois que le premier procmail aura fait son travail, le second prendra le relais en toute sécurité.

**À noter** : si vous classez ainsi vos mails, vous perdrez les avertissements « You have new mail. » Pour remédier à ce problème, il va s'agir d'utiliser une fonctionnalité du shell `zsh`...

Il suffit d'ajouter dans votre `.zshrc` la ligne suivante :

```
MAILPATH="$MAIL:$HOME/mail/Scoubidou?You have new mail in Scoubidou."
```

Si vous voulez appliquer la même technique à plusieurs dossiers, il faut les séparer par des symboles « : » de sorte que si vous êtes aussi fan des Schtroumpfs, vous pourrez, en plus de la recette procmail adaptée, mettre les lignes suivantes dans votre .zshrc :

```
MAILPATH="$MAIL:$HOME/mail/Scoubidou?You have new mail in Scoubidou.\
:$HOME/mail/Schtroumpfs?You have new mail in Schtroumpfs."
```

### Une « boîte à spam »

Une application intéressante de classement concerne le spam. Pour des raisons évoquées plus haut, il n'est parfois pas prudent d'envoyer le courrier suspect directement vers /dev/null/ : vous risquez de perdre du vrai courrier. Il vaut mieux parfois rerouter le courrier vers une boîte à spam.

L'avantage d'une boîte à spam est que vous n'êtes plus dérangé pour rien par les avertissements « You have new mail ». Il vous suffit de consulter la boîte de temps à autre, toutes les semaines disons, et de faire un tri rapide de vrais spams et des faux.

Prenons un exemple. Beaucoup de spams ont un Content-type en text/html, c'est-à-dire du HTML seulement sans copie texte. Inversement, il n'y a guère que les spams pour être en text/html : les gens normaux avec un logiciel mal configuré n'envoient guère que du HTML+ texte (donc un Content-type en multipart/alternative). Vous voulez donc filtrer le text/html. Mais l'expérience prouve que certains logiciels particulièrement malfaisants envoient des courriers parfaitement normaux en text/html. Solution : envoyer tous ces mails dans la boîte à spam.

Application :

```
:0
* ^Content-Type: text/html.*
$HOME/Mail/spam
```

### Recettes imbriquées

Un dernier exemple pour la route ! Encore plus fort, maintenant ! Nous allons reprendre l'exemple du courrier à relayer à votre ami Choupi. Dans l'état où nous vous avons laissé, vous saviez lui relayer le courrier contenant « Scoubidou » dans le sujet et en garder une copie chez vous. Maintenant nous allons vous montrer comment faire ceci, mais en envoyant vos copies à vous dans le dossier « Scoubi. » Il s'agit donc d'une recette constituée de deux recettes imbriquées (chaque recette commence par un :0, souvenez-vous) :

```
:0:
* ^Subject:.*Scoubidou.*
{
  :0 c
  # On fait suivre vers Choupi
  ! choupi@scoubidou-fan.com
  # et en plus de ça...
  :0
  # ... on dépose la copie dans le dossier « Scoubi »
  $HOME/Mail/scoubi
}
```

## Conclusion

Ceci ne montre qu'une partie des possibilités de procmail. Vous pouvez faire beaucoup de choses plus compliquées mais tout aussi utiles : ajouter systématiquement un champ (Reply-To, par exemple), éviter les boucles... Pour davantage de renseignements, consultez les pages de man (si vous ne savez pas ce qu'est une page de man, consultez notre [page sur l'aide en ligne d'Unix](#)) :

- man procmail est une description de procmail lui-même. Vous y trouverez les options de procmail.
- man procmailrc est très complet mais indubitablement compliqué.
- man procmailsc est à réserver aux experts de procmail.

- `man procmail` propose des exemples prêts à utilisation. C'est sans doute la page de `man` la plus utile pour celui qui veut se familiariser davantage avec `procmail`.

Auteur Marie-Lan Nguyen. Dernière modification : le 9 mai 2005.