

Configurer sendmail avec le kit jussieu

© Mathieu DECORE

12 avril 2001

Table des matières

1	Pourquoi configurer un serveur de méls ?	2
2	Pourquoi choisir sendmail ?	3
3	Pourquoi ne pas choisir sendmail ?	3
4	Utiliser le Kit jussieu	4
4.1	Configurer le mél en local uniquement	4
4.2	Configurer le mél pour la maison	8
4.3	Configurer le mél pour un réseau local	10
4.4	Réécrire les logins en Prenom.Nom	11
4.5	Accepter et regiriger les méls adressés à Prenom.Nom	12
4.6	Configurer des hôtes virtuels	13
4.7	La lutte anti-spam	14

On dit souvent que celui qui n'a jamais édité un fichier standard send-mail.cf n'est pas un véritable administrateur UNIX. La légende dit aussi qu'il ne faut pas le faire deux fois, sous peine de devenir fou.

Olaf KIRCH

1 Pourquoi configurer un serveur de méls ?

Pour lire ses méls chez soi, ou faire en sorte que les utilisateurs d'un réseau local puissent lire leur méls, il y a plusieurs solutions, plus ou moins faciles à mettre en oeuvre :

- utiliser un programme, tel que `kmail`, qui va se connecter à un autre serveur de méls pour envoyer ou récupérer les courriers (c'est en général le serveur de méls du fournisseur d'accès qu'on utilise) ;
- utiliser le serveur de méls du réseau local qui va se débrouiller pour transmettre les courriers, soit en passant par un serveur de méls relai, soit en contactant directement le serveur de méls de la personne à qui on souhaite envoyer un courrier.

La deuxième solution nécessite l'installation et la configuration d'un serveur DNS et d'un serveur de méls, ce qui est loin d'être trivial. De plus, ces deux nouveaux services sont sources de failles de sécurité.

Alors, pourquoi vouloir installer son propre serveur de méls ? Plusieurs raisons peuvent motiver un utilisateur à le faire :

- si l'on dispose de plusieurs ordinateurs, on peut faire en sorte que ceux-ci centralisent les courriers sur la seule machine du réseau connectée à l'Internet ;
- si on veut envoyer des courriers avec un client mél comme `mutt` ou des programmes tels que PHP, Perl, etc. ;
- si le serveur de méls du fournisseur d'accès est jugé trop lent, on gagnera du temps en contactant directement le serveur local de méls, qui lui-même contactera le serveur de méls du correspondant ;
- c'est formateur et culturel.

2 Pourquoi choisir sendmail ?

sendmail est LE serveur de méls : il a été conçu en 1982, il est présent et utilisé sur la plupart des serveurs (on estime que un courrier transitant à travers l'Internet passe par au moins un serveur de méls **sendmail**), il est souple et puissant. Dans le monde fluctuant de la messagerie naissante, il a évolué depuis 20 ans ; cela prouve que le principe de conception est bon.

sendmail a mauvaise réputation. Pendant de nombreuses années, il n'y avait pas d'alternative aussi puissante, et l'écriture du fichier de configuration était le cauchemard des administrateurs systèmes. Aujourd'hui avec l'utilisation du kit jussieu ou des macros **m4**, c'est devenu un jeu d'enfant de le configurer.

Par ailleurs, il y a eu quelque problèmes de sécurité. Ceux-ci s'expliquent aisément : **sendmail** est un programme lancé par n'importe quel utilisateur, mais il est *root set-uid* (l'utilisateur lançant **sendmail** le fait avec les droits du **root**). En plus, comme c'est le serveur de méls le plus utilisé, la moindre faille se sait rapidement, et les conséquences sont plus importantes que si on utilise un serveur de méls quelconque.

sendmail peut gérer un nombre important de connexions simultanément, et il le fait bien. La plupart des "gros" serveurs dans le monde l'utilisent d'ailleurs, et sa fiabilité n'est plus à prouver.

3 Pourquoi ne pas choisir sendmail ?

sendmail est un peu une "usine à gaz", gourmand en ressources CPU. Si on a un petit site ne recevant pas beaucoup de courriers, ou des serveurs modestes, il faudrait peut-être mieux se tourner vers d'autres serveurs de méls, comme **qmail** ou **postfix** voir **exim**. De plus, **sendmail** reste un des serveurs de méls les plus difficiles à configurer.

Dans la suite, deux méthodes sont présentées : l'une en utilisant le kit jussieu, disponible à l'adresse <ftp://ftp.jussieu.fr/jussieu/sendmail/kit/> [1]. L'autre utilise les macros **m4**, livrées avec la plupart des distributions.

Comme le disaient JC Pollman et Bill Mote [2] :

*De prime abord, cela semble un projet assez simple, et pourtant relativement ambitieux : si vous n'êtes pas en terrain connu, dites adieu à toute votre petite famille pour pas loin d'une semaine.
Soyez cependant assurés que le jeu en vaut la chandelle.*

4 Utiliser le Kit jussieu

Configurer **sendmail** en utilisant le kit jussieu est relativement simple : il y a peu de variables à préciser, la plupart ont des valeurs par défaut qui sont bonnes en général. De plus, la documentation est très bien faite (et en français, s'il vous plaît !). Des exemples types sont détaillés dans la documentation [3], et livrés avec le kit.

En fait, dans les cas simples auxquels on va se limiter (qui sont des cas courants), nous n'avons que cinq variables à positionner :

Host nom d'hôte de la station (celui donné par la commande **hostname**) ;

Domaine nom complet du domaine courant (celui donné par la commande **hostname -d**) ;

ListeDomaines noms de domaines à reconnaître comme étant le domaine courant (les méls adressés à **user@domaine** doivent être reçus par **sendmail** si cette variable contient **domaine**) ;

AdressesLocales spécifie quelles sont les adresses à reconnaître comme locales, c'est-à-dire pour lesquelles les courriers restent sur cette machine. On verra dans chaque cas quel mot-cléf mettre ;

AdressesInternes spécifie quelles sont les adresses reconnues comme étant dans notre juridiction, c'est-à-dire pour lesquelles cette machine agit comme "redistributeur" de courriers ;

4.1 Configurer le mél en local uniquement

On cherche à faire fonctionner le serveur de méls pour une machine unique **machine.domaine.local** reliée à aucune autre machine. On veut donc que tous les courriers destinés à notre domaine restent sur cette machine.

Pour configurer les méls en local, il faut positionner les variables du kit comme suit :

ListeDomaines les méls adressés au domaine local doivent être acceptés par **sendmail**. Cette variable contient donc le nom du domaine, contenu dans la variable **Domaine** définie précédemment (attention, il faut mettre des doubles quotes pour que le \$ soit interprété) ;

AdressesLocales seuls les courriers destinés à l'hôte local doivent rester sur cette machine. Cette variable est donc initialisée à *HOST* ;

AdressesInternes aucun courrier n'est interne. Tout courrier non local est considéré comme externe. Cette variable est donc initialisée à *RIEN* ;

Les autres variables n'ont pas besoin d'être modifiées en principe. Ces variables sont :

MailerLocal pour acheminer les méls reçus par **sendmail** à l'utilisateur, il faut utiliser un mailer local. Ce mailer peut-être **mail** ou **procmail**. Ce dernier permet de trier les méls, ce qui peut-être intéressant à utiliser. Pour connaître les options à passer, chercher dans le fichier */etc/sendmail.cf* livré par défaut la chaîne *Mlocal* :

```
$ grep -A2 Mlocal /etc/sendmail.cf
Mlocal,          Path=/usr/bin/procmail, Flags=lsDFMAw5:/|wA9q,
                  Sender=10, Recipient=20, Argv=procmail -Y -a $h -d $u
```

\$

Aliases localisation du fichier des aliases, permettant d'accepter les méls à destination de **Prenom.Nom@domaine** et de les redistribuer à l'utilisateur associé (voir 4.5). Comme précédemment, pour être sûr de la localisation de ce fichier, chercher la chaîne *OA* :

```
$ grep OA /etc/sendmail.cf
OA/etc/aliases
$
```

Ce fichier peut se trouver où on veut du moment qu'on le précise. Cependant si un fichier par défaut */etc/mail/aliases* existe déjà avec des alias prédéfinis, il faut mieux éviter les doublons en éditant par la suite un fichier */etc/sendmail.cf*...

SendmailSt localisation du fichier d'aide de **sendmail**. Ce fichier est utilisé dans deux occasions : sur requête **HELP** dans une session **SMTP**, ou dans le mode de test des règles de réécriture de **sendmail** (option *-bt*) ;

Mqueue localisation du répertoire servant de file d'attente. C'est dans ce répertoire que seront stockés les méls avant de partir ;

Voici donc le fichier `local.config` à écrire :

```
#
# Nom d'hôte de la machine locale
#

Host='machine'

#
# Nom de domaine de la machine locale
#

Domaine='domaine.local'

#
# Les mels adresses au domaine local doivent etre acceptes par sendmail
#

ListeDomaines="$Domaine"

#
# Seuls les courriers destines a machine.domaine.local doivent rester sur
# cette machine
#

AdressesLocales='HOST'

#
# Aucun courrier n'est interne. Tout courrier non local est considere
# comme externe
#

AdressesInternes='RIEN'

#
# Ce qui suit n'a pas besoin d'etre modifie, normalement ;- )
#

#
# Parametres du mailer local. Si procmail n'est pas installe, mettre :
#
```

```

# MailerLocal='/bin/mail DFMPImnrs mail -d $u'
#

MailerLocal='/usr/bin/procmail lsSDFMhPfn procmail -Y -a $h -d $u'

#
# Localisation du fichier des aliases
#

Aliases='/etc/aliases'

#
# Localisation du fichier de compatibilite de sendmail
#

SendmailSt='/etc/sendmail.st'

#
# Localisation du fichier d'aide de sendmail
#

SendmailHf='/usr/lib/sendmail.hf'

#
# Localisation du repertoire servant de file d'attente
#

Mqueue='/var/spool/mqueue'

```

Comme on ne précise pas ce qu'il faut faire avec les courriers externes, `sendmail` va chercher à résoudre le nom de domaine du destinataire et, n'y parvenant pas, enverra un mél d'erreur à l'expéditeur. Comme on n'a pas de connexion vers l'extérieur, c'est bien ce qu'on attend de notre serveur de méls.

Pour générer le fichier `sendmail.cf`, taper :

```

$ configurateur vide/regles.vide local.config > sendmail.cf
$

```

Il ne reste alors plus qu'à copier ce fichier dans le répertoire `/etc` et de relancer `sendmail` pour tester :

```

# /etc/init.d/sendmail restart

```



```
Stopping mail transport agent: sendmail.  
Starting mail transport agent: sendmail.  
#
```

On va maintenant pouvoir tester notre nouveau serveur de méls. Pour cela, voir la semaine prochaine.

4.2 Configurer le mél pour la maison

On cherche à envoyer des courriers vers l'Internet depuis notre unique machine **machine.domaine.local**. On passe par un fournisseur d'accès à Internet, qui possède un serveur de méls **smtp.fai.fr** qui peut nous relayer. De plus, on veut que les courriers partant aient une adresse d'expéditeur de la forme **login@fai.fr**, pour qu'une réponse à nos courriers se fasse sur une adresse valide (une adresse du style **login@domaine.local** n'étant pas valide).

Voici donc ce que doit faire notre serveur de méls :

- **machine** envoie et reçoit des méls en local. Tous les autres méls doivent transiter par le serveur de méls du fournisseur d'accès **smtp.fai.fr** ;
- un mél envoyé depuis **machine.domaine.local** doit avoir comme nom d'hôte émetteur **fai.fr** et non **domaine.local** (masquage du nom d'hôte).

```
#  
# Nom d'hote de la machine locale  
#  
  
Host='machine'  
  
#  
# Nom de domaine du fournisseur d'accès  
#  
  
Domaine='fai.fr'  
  
#  
# Seuls les mels adresses au domaine local doivent etre acceptes par sendmail  
#  
  
ListeDomaines="$Domaine"  
  
#
```

```
# Seuls les courriers destines a machine.domaine.local doivent rester sur  
# cette machine
```

```
#
```

```
AdressesLocales='HOST'
```

```
#
```

```
# Aucun courrier n'est interne. Tout courrier non local est considere  
# comme externe
```

```
#
```

```
AdressesInternes='RIEN'
```

```
#
```

```
# Signature des courriers émis localement. Les mels envoyes auront pour  
# champ From: login@fai.fr au lieu de From: login@machine.domaine.local
```

```
#
```

```
ReecritureAdressesLocales=$Domaine
```

```
#
```

```
# Tout le courrier non local et non interne doit etre envoye (sans tenir  
# compte des MX) a l'adresse specifiee (le serveur de mels du fournisseur  
# d'accès)
```

```
#
```

```
RelaisExterieur=smtp.$Domaine
```

```
#
```

```
# Ce qui suit n'a pas besoin d'etre modifie, normalement ;-)
```

```
#
```

```
#
```

```
# Parametres du mailer local. Si procmail n'est pas installe, mettre :
```

```
#
```

```
# MailerLocal='/bin/mail DFMPImnrs mail -d $u'
```

```
#
```

```
MailerLocal='/usr/bin/procmail lsSDFMhPfn procmail -Y -a $h -d $u'
```

```
#
# Localisation du fichier des aliases
#

Aliases='/etc/aliases'

#
# Localisation du fichier de compatibilite de sendmail
#

SendmailSt='/var/log/sendmail.st'

#
# Localisation du fichier d'aide de sendmail
#

SendmailHf='/usr/share/misc/sendmail.hf'

#
# Localisation du repertoire servant de file d'attente
#

Mqueue='/var/spool/mqueue'
```

4.3 Configurer le mél pour un réseau local

Maintenant, on veut que seuls les courriers extérieurs à notre réseau local **domaine.local** soient routés vers l'extérieur, les autres devant rester dans notre domaine. On voudrait aussi que les courriers adressés à l'extérieur ne passent pas par un serveur de méls relai comme précédemment, mais qu'ils soient envoyés directement au serveur destinataire.

Pour cela, il faut d'abord installer un serveur de noms (DNS). Ensuite, il suffit de ne pas préciser de serveur relai, **sendmail** s'occupera de contacter le serveur destinataire directement en utilisant le protocole SMTP (voir <http://www.toolinux.com/linutile/reseau/mail/smtp/index.htm>).

```
Host='machine'
Domaine='domaine.local'
ListeDomaines="$Domaine"

#
```

```
# Seuls les courriers destines a machine.domaine.local et a domaine.local
# doivent rester sur cette machine
#

AdressesLocales='DOMAINE'

#
# Tous les courriers adresses a *.domaine.local doivent etre distribues
#

AdressesInternes='ETOILE_DOMAINE'

#
# Signature des courriers emis localement
#

ReecritureAdressesLocales=$Domaine

MailerLocal='/usr/bin/procmail lsSDFMhPfn procmail -Y -a $h -d $u'
Aliases='/etc/aliases'
SendmailSt='/etc/sendmail.st'
SendmailHf='/usr/lib/sendmail.hf'
Mqueue='/var/spool/mqueue'
```

4.4 Réécrire les logins en Prenom.Nom

Pour réécrire les logins des utilisateurs en leur Prenom.Nom, il faut une base de données faisant la correspondance. Cette base de données peut être générée automatiquement à partir du fichier `/etc/passwd`, voir...

La première étape est d'écrire un fichier qu'on peut appeler `genericstable`, à placer dans le répertoire `/etc` ou `/etc/mail`. Ce fichier existe sans doutes déjà :

```
# locate genericstable
/etc/mail/genericstable
/etc/mail/genericstable.db
#
```

On voit ici que le fichier existe bien déjà. On constate aussi que la base de données associée, `genericstable.db`, se trouve dans ce même répertoire.

Ce fichier `genericstable` peut ressembler à ceci :

```
mdecore:          mdecore@linux-france.org
```

Il faut maintenant générer la base de données à l'aide de **sendmail** à partir du répertoire où se trouve le fichier **genericstable** (**/etc** ou **/etc/mail**, en général) :

```
# sendmail -bi -oA/etc/genericstable
#
```

Une autre solution qui revient au même consiste à écrire un fichier ayant un format légèrement différent (sans les “:”) et d'appeller **makemap**.

Voici le fichier **genericstable** :

```
jbon  Jean.Bon
jdeuf  John.Deuf
```

Et voici comment générer la base de données, toujours à partir du répertoire où se trouve le fichier **genericstable** :

```
# makemap hash /etc/mail/genericstable < /etc/mail/genericstable
#
```

Il ne reste plus qu'à rajouter la ligne suivante dans le fichier à passer au configurateur du kit jussieu :

```
RevAliases='hash -N /etc/mail/genericstable'
```

Noter que certaines distributions livrent un fichier **Makefile** permettant d'automatiser la compilation de ce fichier, ainsi que celle d'autres fichiers du même genre.

4.5 Accepter et regiriger les méls adressés à **Prenom.Nom**

Pour faire en sorte que les méls adressés à **Prénom.Nom@domaine** soient redirigés vers la boîte aux lettre du bon utilisateur, il faut éditer le fichier **/etc/aliases** :

```
Jean.Bon@domaine: jbon
John.Deuf@domaine: jdeuf
```

```
# Tous les mels adresses a root sont redistribues a sysadm
root: sysadm
```

```
# Tous les mels adresses a sysadm sont redistribues a admin et moi
```

```
sysadm: admin moi
```

```
# Une liste de discussion
```

```

admin: jbon, Harry.Cover@labas.com

# Pour moi, on execute le script /home/moi/monfiltre

moi: | /home/moi/monfiltre

# Ceci est une liste de diffusion geree par un utilisateur. La liste des
# gens y etant abonnées est stockee dans le fichier /home/mdecore/foot

foot::include:/home/mdecore/foot

```

Encore une fois, ce fichier existe sans doute déjà, avec des alias prédéfinis (comme **postmaster** qui est plus que conseillé par la RFC...). Il faut donc sans doute éditer le fichier existant. Si un alias commence par “|”, c’est un script qui sera lancé.

Lancer la commande **newaliases** (équivalente à **sendmail -bi**) pour mettre à jour la base de données (**/etc/aliases.db**).

Et voilà, ce n’est pas plus compliqué que ça ! Il ne reste plus qu’à recompiler notre fichier du kit jussieu, à relancer **sendmail** comme précédemment et à rajouter la ligne suivante dans le fichier à passer au configurateur du kit jussieu :

```
Aliases='/etc/mail/aliases'
```

4.6 Configurer des hôtes virtuels

Configurer des hôtes virtuels avec le kit jussieu est particulièrement simple : il suffit de rajouter dans le fichier **/etc/mail/ableroutages** le nom des hôtes devant accepter des courriers comme suit :

```

domaine1.com: local.MACHIN
domaine2.com: local.MACHIN
domaine3.com: local.MACHIN
domaine4.net: local.MACHIN

```

Le fichier est au format **domaine : agent.relais**, qui indique que tout courrier adressé au domaine *domaine* devra être routé vers le relai *relais* en utilisant l’agent *agent*. Dans notre cas, on indique que tous les méls adressés à un hôte du domaine **domaine2.com** doivent être envoyés au mailer local, qui va lire le fichier **/etc/mail/aliases**. Il doit donc y avoir dans ce fichier (ainsi que dans le fichier **/etc/mail/revalias**) une correspondance pour

chaque utilisateur.

Il faut alors compiler ce fichier :

```
/usr/lib/sendmail -bi -oA/usr/local/etc/mail/routages
```

et indiquer dans notre fichier de configuration à passer au configurateur l'emplacement de ce fichier :

```
TableRoutages='hash -N /etc/mail/ableroutages'
```

4.7 La lutte anti-spam

Pour la lutte anti-spam, il faut commencer par rajouter la ligne suivante dans le fichier à passer au configurateur :

```
ListeNoire="hash -N /usr/local/etc/liste-noire"
```

Le fichier `/usr/local/etc/liste-noire` doit alors contenir :

domaine	mot-clef
adresse IP	mot-clef
utilisateur@domaine	mot-clef

ou *mot-clef* est :

SPAM adresse connue pour être spammeuse ;

OK adresse connue pour ne pas être spammeuse, alors que l'adresse englobante est connue pour être spammeuse ;

LOCAL adresses pour le compte desquelles on accepte de relayer les courriers.

En pratique, cela donne :

mechant.com	SPAM
cops.mechant.com	OK
abrutit@fai.fr	SPAM
193.253.181.72	SPAM
eudora.domaine.com	LOCAL

Il ne reste plus qu'à compiler le fichier :

```
cd /usr/local/etc  
makemap -N hash liste-noire < liste-noire
```

Références

- [1] **Le Kit de configuration de sendmail de Jussieu** , Pierre David, Jacky Thibault et Sébastien Vautherot, 20 Janvier 2000.
<ftp://ftp.jussieu.fr/jussieu/sendmail/kit/>
<http://www.prism.uvsq.fr/~pda/kit-jussieu/>
- [2] **Du courrier électronique pour le réseau domestique**, JC Pollman et Bill Mote, Septembre 1999.
- [3] **Sendmail v8 Kit Jussieu**, Pierre David, Juillet 1999.
<http://www.prism.uvsq.fr/~pda/kit-jussieu/support/index.html>