

Configurer postfix

© Mathieu DECORE

12 juillet 2001

Table des matières

1	Présentation de postfix	2
2	Installation de postfix	2
3	Configurer postfix	2
3.1	Configuration basique	3
3.2	Configuration pour une machine isolée	4
3.3	Post-installation	4
4	Tester notre configuration	5
5	Station connectée à l'Internet	5
5.1	Utiliser un relai	6
5.2	Gérer les alias	6
5.3	Réécrire les adresses du champ <i>From</i> :	6
5.4	Gérer les domaines virtuels	7
5.4.1	Utiliser transport	7
5.4.2	Utiliser virtual	8
5.4.3	La lutte anti spam	8
6	Passer de sendmail à postfix	9
6.1	A partir de m4	9
6.2	A partir du kit jussieu	9

1 Présentation de postfix

`postfix` a pour ambition de remplacer `sendmail`. Pour cela, l'auteur (Wietse Venema, développeur de TCP wrappers et co-créateur de SATAN) a décidé de le rendre compatible à 100 % avec les fichiers de configurations de `sendmail`, à l'exception du fameux fichier `/etc/sendmail.cf` qui fût pendant longtemps le cauchemar des administrateurs systèmes. Cela permet un passage en douceur pour les administrateurs de `sendmail` (l'administration reste la même, et des outils supplémentaires leurs sont fournis), tout en ayant un programme modulaire et plus sécurisé.

2 Installation de postfix

L'installation de `postfix` se fait souvent à partir des packages pré compilés (RPM ou DEB). On peut néanmoins, si on désire utiliser une version récente, compiler `postfix` à partir des sources.

Pour installer à partir des packages, il faut récupérer les fichiers `postfix.rpm` et `pcrc-3.2-1.i386.rpm` et les installer. Pour installer à partir des sources, il faut récupérer l'archive, la décompresser et compiler comme on le fait d'habitude (`make && make install` devrait suffire). Il faut répondre à quelques questions, mais dans la plupart des cas on peut se contenter d'appuyer sur `Entree`.

3 Configurer postfix

Dans un premier temps, il faut lancer `postfix` si cela n'est déjà fait :

```
# postfix start
postfix-script: starting the Postfix mail system
#
```

On peut vérifier dans le fichier de logs (défini par `syslog`, voir <http://www.toolinux.com/linutile/reseau/mail/sendmail/sendmail-test/index4.htm>) que l'opération a réussi :

```
# tail /var/log/mail.info
[...]
Apr 22 17:20:39 zecastor postfix-script: starting the Postfix mail system
Apr 22 17:20:39 zecastor postfix/master[997]: daemon started
#
```

La configuration de **postfix** se fait presque entièrement à partir du fichier `/etc/postfix/main.cf`. Après chaque modification de ce fichier, il faut indiquer à **postfix** qu'il faut qu'il relise son fichier de configuration :

```
# /etc/init.d/postfix reload
#
```

Le fichier `/etc/init.d/postfix` peut également se trouver dans les répertoires `/etc/rc.d/init.d` ou encore `/sbin/init.d` suivant la distribution utilisée.

Dans la suite, les modifications portent, sauf mention contraire, sur le fichier `/etc/postfix/main.cf`.

3.1 Configuration basique

Si la commande **hostname** ne retourne rien (ou par de nom de domaine pleinement qualifié, c'est à dire de la forme **machine.domaine.fr**), il faut modifier les variables *myhostname* et *mydomain* :

```
myhostname = tarsier.chez.moi
mydomain = chez.moi
```

postfix faisant appel à plusieurs programmes (modularité signifiant sécurité), chaque processus est lancé selon la description dans le fichier `/etc/postfix/master.cf`. En principe, on ne devrait pas avoir à modifier ce fichier, mais c'est toujours bon à savoir qu'il existe.

Les adresses ne contenant que le nom d'utilisateur peuvent être réécrites en ajoutant le nom du domaine (de la forme **user@\$mydomain**)

```
append_at_myorigin = yes
```

Cela permet de transformer une adresse d'expéditeur non valide (**user**) en adresse valide en tout point de l'internet (**user@domaine**).

Pour les mêmes raisons, on doit pouvoir transformer le nom de machine en nom pleinement qualifié (de la forme **user@machine.domaine**) :

```
append_dot_mydomain = yes
```

Pour utiliser **procmail** comme agent de délivrance local de courrier, décommenter la ligne suivante :

```
mailbox_command = /usr/local/bin/procmail
```

3.2 Configuration pour une machine isolée

On peut commencer par modifier la variable *myorigin*, qui contient le nom du domaine tel qu'il apparaîtra dans le champ *From* : des méls. Par défaut c'est le contenu de la variable *myhostname*, c'est à dire **tarsier.chez.moi** dans notre cas. On peut vouloir mettre le nom du domaine seul si on a plusieurs machines dans notre réseau local : *myorigin* = *\$mydomain*.

Enfin, la variable *mydestination* contient la liste des domaines que notre machine considère comme destination finale (en dehors des domaines virtuels). Ici comme on a des adresses d'expéditeurs de la forme **user@domaine.fr**, il faut accepter les méls adressés à ce type d'adresse (en cas de réponse à un mél, par exemple). Il faut donc rajouter *\$mydomain* à la variable *mydestination* pour que la liste soit complète :

```
mydestination = $myhostname, localhost.$mydomain $mydomain
```

Pour plus de détails sur toute ces options, voir `trivial-rewrite(8)`.

3.3 Post-installation

Il existe un programme très pratique, qui permet de visualiser et gérer la configuration de postfix : il s'agit de `postconf` (voir `postconf(8)`). Pour connaître les valeurs courantes des paramètres, taper simplement `postconf`. Pour avoir les valeurs par défaut taper `postconf -d` et pour avoir la liste des paramètres modifiés `postconf -n` :

```
tarsier:~ # postconf -n
alias_database = hash:/etc/aliases
alias_maps = hash:/etc/aliases
command_directory = /usr/sbin
daemon_directory = /usr/lib/postfix
debug_peer_level = 2
default_destination_concurrency_limit = 10
local_destination_concurrency_limit = 2
mail_owner = postfix
mydestination = $myhostname, localhost.$mydomain $mydomain
mydomain = chez.moi
myorigin = $mydomain
queue_directory = /var/spool/postfix
tarsier:~ #
```

4 Tester notre configuration

Pour commencer, on peut essayer d'envoyer un mél à un utilisateur :

```
mdecore@zecastor:~$ mail -v mdecore
Subject: Test Postfix
Ben voilà, quoi...
.
Cc:
postdrop: open maildrop/873921704B
postdrop: print string: 873921704B
send-mail: mail_scan_any: read string: 873921704B
postdrop: print int: 0
send-mail: mail_scan_any: read integer: 0
mdecore@zecastor:~$
```

Voici ce qu'indique alors le fichier de log :

```
zecastor:~# tail /var/log/mail.info
Apr 22 17:20:39 zecastor postfix/master[997]: daemon started
Apr 22 17:31:22 zecastor postfix/postdrop[1127]: open maildrop/873921704B
Apr 22 17:31:22 zecastor postfix/postdrop[1127]: print string: 873921704B
Apr 22 17:31:22 zecastor postfix/sendmail[1126]: mail_scan_any: read string:
873921704B
Apr 22 17:31:22 zecastor postfix/postdrop[1127]: print int: 0
Apr 22 17:31:22 zecastor postfix/sendmail[1126]: mail_scan_any: read integer: 0
Apr 22 17:31:22 zecastor postfix/pickup[998]: D68021704D: uid=1000
from=<mdecore>
Apr 22 17:31:23 zecastor postfix/cleanup[1128]: D68021704D:
message-id=<20010422153122.D68021704D@zecastor.chez.moi>
Apr 22 17:31:23 zecastor postfix/qmgr[999]: D68021704D:
from=<mdecore@zecastor.chez.moi>, size=347 (queue active)
Apr 22 17:31:23 zecastor postfix/local[1130]: D68021704D:
to=<mdecore@zecastor.chez.moi>, relay=local, delay=1, status=sent (mailbox)
zecastor:~#
```

5 Station connectée à l'Internet

Notre machine est connectée de façon intermitante à l'Internet, on doit donc utiliser un serveur pour relayer nos méls. De plus, notre nom de domaine n'est pas officiel, il va donc falloir réécrire notre adresse d'expéditeur.

5.1 Utiliser un relai

On indique le nom de notre serveur relai (sinon on utilise une connexion directe SMTP) :

```
relayhost = smtp.free.fr
```

Pour ne pas utiliser de service DNS lors du dépôt du courrier (si offline) :

```
disable_dns_lookups = yes
```

Pour différer l'envoi de courrier jusqu'à l'établissement d'une connexion SMTP :

```
defer_transports = smtp
```

5.2 Gérer les alias

Il y a deux façons de faire :

- utiliser `alias_maps` pour indiquer le fichier d'alias et son format :

```
alias_maps = dbm:/etc/aliases
et compiler ce fichier avec postalias :
# postalias -v dbm:/etc/aliases
#
```

- utiliser `alias_database` :

```
alias_database = hash:/etc/aliases
et compiler avec newaliases (comme avec sendmail) :
# newaliases
#
```

Pour plus de détails, voir `aliases(5)`.

5.3 Réécrire les adresses du champ *From* :

Dans le fichier `/etc/postfix/scanonical`, entrer les correspondances entre les adresses locales et les adresses réécrites :

```
mdecore@chez.moi mdecore@free.fr
piou@chez.moi    piaf@cuicui.com
```

Compiler ce fichier :

```
# postmap -v btree:/etc/postfix/scanonical
#
```

et indiquer qu'on doit utiliser cette base de données pour réécrire chaque mél sortant :

```
sender_canonical_maps = btree:/etc/postfix/scanonical
```

`mailq` (ou `sendmail -bp`) liste la file d'attente et `sendmail -q` vide la file.

Noter que `masquerade_domains = domaine.fr` réécrit tout `user@machine.domaine.fr` en `user@domaine.fr` et que `masquerade_exceptions = root` permet de spécifier que `root` n'est pas réécrit (comportement par défaut de `sendmail`).

Pour plus de détails, voir `canonical(5)`.

5.4 Gérer les domaines virtuels

On peut gérer les hôtes virtuels de deux façons : en utilisant le fichier `transport` (voir `transport(5)`) ou le fichier `virtual` (voir `virtual(5)`).

5.4.1 Utiliser `transport`

Dans le fichier `transport`, mettre la liste des domaines pour lesquels on accepte du courrier :

```
tarsier.domaine.org local:
localhost local:
autre.autredomaine local:
```

puis compiler ce fichier :

```
# postmap /usr/local/etc/postfix/transport
#
```

et indiquer dans le fichier `/etc/postfix/main.cf` dans quel fichier il faut lire la liste de ces domaines :

```
transport_maps = hash:/usr/local/etc/postfix/transport
```

Noter que `transport` outrepassse le contenu de la variable *`mydestination`*.

Maintenant il faut préciser comment redistribuer le courrier au différents domaines. C'est le fichier `/etc/postfix/canonical` qui s'en charge comme indiqué dans le fichier `/etc/postfix/main.cf` :

```
canonical_maps = hash:/etc/postfix/canonical
```

Voici le contenu du fichier `/etc/postfix/canonical` :

```
user          autreuser
user@domaine1.fr  autreutilisateur@domaine5.fr
@domaine1.fr    @domaine2.fr
@domaine3.fr    user@domaine4.fr
```

C'est en fait exactement le même format que le fichier `/etc/mail/virtusertable` présenté pour `sendmail` (voir ??).

5.4.2 Utiliser virtual

Le fichier `/etc/postfix/virtual` permet d'héberger plusieurs domaines sur la même machine sans réécrire les *headers* (ne réécrit que l'enveloppe) :

```
domaine0          nimportequoi
user1@domaine1     adresse1,adresse2
user2              adresse3,adresse4 # comme les alias
@domaine3.fr       adresse5,adresse6
@domaine4.fr       @domaine5.fr # pas de changement de la partie gauche
```

Il faut indiquer dans le fichier `/etc/postfix/main.cf` :

```
virtual_maps = hash:/etc/postfix/virtual
```

5.4.3 La lutte anti spam

Pour lutter contre le spam, il suffit de préciser dans le fichier `/etc/postfix/main.cf` que les règles pour filtrer les en-têtes des courriers sont dans le fichier `/etc/postfix/header_checks` et le contenu de chaque message dans le fichier `/etc/postfix/body_checks` :

```
header_checks = regexp:/etc/postfix/header_checks
body_checks = regexp:/etc/postfix/body_checks
smtpd_client_restrictions = reject_maps_rbl
maps_rbl_domains = rbl.maps.vix.com, dul.maps.vix.com
```

Voici un exemple de fichier `/etc/postfix/header_checks` :

```
/^from: .+@.*mechant.com/ REJECT
```

Pour renvoyer tout courrier venant du domaine **mechant.com**.

Voici pour terminer quelques options supplémentaires :

```
debugger_command
Nombre d'instances d'un module : default_process_limit = 10
Nombre de courriers adressés simultanément (entre 2 et 10) :
initial_destination_concurrency_limit = 2
default_destination_concurrency_limit = 10
local_destination_concurrency_limit = 2 (l'accès à la BAL est séquentiel)
smtp_destination_concurrency_limit = 4 (vers le FAI)

# Ne pas accepter les machines sans DNS inverse (stupide et dangereux)
# Spammers a mettre dehors à la connexion
smtpd_client_restrictions = reject_unknown_client,
hash:/etc/postfix/Access
```

6 Passer de sendmail à postfix

Cette section donne des éléments pour passer d'une configuration de sendmail à la configuration de postfix.

6.1 A partir de m4

m4	Postfix
FEATURE(mailertable)	transport
FEATURE(generics)	canonical_map
FEATURE(virtusertable)	virtual_maps
FEATURE(bestmx_is_local)	best_mx_transport = local

6.2 A partir du kit jussieu

Kit jussieu	Postfix
ListeDomaines et AdressesLocales	<i>mydestination</i>
AdressesInterne	relay_domains
RelaisExterieur	relayhost
TableRoutages	transport
ListeNoire	smtpd_sender_restrictions, mynetworks et relay_domains
RevAliases	sender_canonical_maps
SansCanonisation	disable_dns_lookups=yes
MailerLocal	mailbox_command

Références

- [1] **Poster et recevoir du courrier avec Postfix**, Éric Jacoboni, 2000.
<http://www.linux-france.org/article/mail/postfix-jaco/>
- [2] **Tutoriel Postfix**, Hervé Schauer Consultants, 1999.
<http://www.hsc.fr/ressources/cours/postfix/>