

Installer Qmail et sa périphérie

Avertissement

Qmail est l'un des serveurs de messagerie les plus performants, les plus sûrs et les plus stables, probablement supérieur à Postfix dans le cas d'un traitement de très gros flux de messages, mais c'est un outil plutôt difficile à installer et à maîtriser.

D.J Bernstein, l'auteur de Qmail, a mis son logiciel sous une licence qui n'autorise que la distribution des sources sans trop de restrictions, ce qui fait que l'on ne trouve généralement pas de paquets binaires dans les distributions GNU/Linux.

Debian, par exemple, contourne le problème en fournissant un paquetage source muni de scripts qui permettent de construire de façon totalement automatique un paquetage binaire de Qmail.

D'autres solutions, comme `qmail-toaster`¹, proposent des paquets sources au format rpm (RedHat, Fedora, Mandriva...).

Cependant, de telles solutions proposent des installations préfabriquées, qui ne correspondront pas forcément à notre attente et qui, de toutes façons, ne permettront que difficilement de comprendre convenablement le fonctionnement du système. Qmail et sa périphérie représentent une suite d'outils complexes qu'il est nécessaire de maîtriser si l'on souhaite l'utiliser convenablement.

Rien n'empêche, une fois acquise la maîtrise de Qmail et de ses satellites, de passer en toute connaissance de cause à des solutions comme celle offerte par les paquets de `qmail-toaster`.

Pour ces raisons, nous ferons ici une mise en oeuvre basée sur les sources, ce qui ne nous empêchera pas d'utiliser des paquets binaires chaque fois que ce sera possible.

¹ Qmail-toaster : <http://www.qmailtoaster.com/>

Plan du chapitre

Avertissement.....	1
Daemontools et ucspi-tcp.....	4
Présentation.....	4
Les daemontools.....	4
Présentation.....	4
Fonctionnement.....	4
ucspi-tcp.....	6
Recommandations.....	6
Qmail & co.....	7
Conditions d'expérimentation.....	7
Premier problème.....	7
Qmail.....	7
On "patche" ou pas ?.....	7
Un Qmail "amélioré".....	8
Netqmail 1.05.....	8
Qmail-toaster-0.8.1 patch.....	8
Où prendre les ressources.....	8
Installation.....	9
Préparation de netqmail.....	9
Installation des daemontools.....	9
Installation de UCSPI-TCP.....	10
Première installation de Qmail.....	10
Et sendmail ?.....	12
Les logs.....	12
Démarrage des services.....	12
qmail-send.....	13
qmail-smtpd.....	13
softlimit.....	13
tcpserver.....	13
qmail-pop3d.....	14
qmail-pop3ds.....	14
Mise en place des scripts de gestion des logs.....	14
Conclusion.....	17
Vpopmail.....	18
Présentation rapide.....	18
Domaines virtuels.....	18
Authentification des utilisateurs.....	18
Installation.....	18
Travaux préliminaires.....	19
Configuration de la compilation.....	19
Un premier domaine virtuel.....	22
Qmail, installation définitive.....	23
TCPSEVER.....	23
Réinstallation de Qmail.....	24
spfbehavior.....	24
Démarrage du système.....	24

Notes importantes.....	26
Ajoutons un utilisateur.....	27
Administration de Vpopmail.....	28
Installation de vqadmin.....	28
Configuration d'Apache.....	29
Un peu de mécanique.....	33
Essais divers.....	35
Le tour du propriétaire.....	35
Utilisation purement locale.....	35
Ne pas oublier les alias.....	35
Remarque à propos des alias.....	37
Et les logs ?.....	37
Tester qmail-smtpd.....	39
Le serveur IMAP : courier-imap.....	43
Installation du paquetage source "courier".....	43
Attention, cette méthode présente un risque !.....	45
Configuration de courier-authdaemon.....	45
Configuration de courier-imap.....	45
Essais.....	45
Ezmlm-idx.....	48
Des patches, des patches.....	48
La gestion des listes.....	48
Dans la salle des machines.....	48
qmailadmin.....	50
Autorespond.....	50
qmailadmin.....	50
Filtrage des mails.....	58
Discussion sur les stratégies de filtrage.....	58
Virus et vers.....	58
Spam.....	58
Où vaut-il mieux filtrer ?.....	58
Clamav.....	58
Spamassassin.....	59
Simscan.....	60
Contrôles.....	63
Livraison locale.....	64
Que faire des spams ?.....	64
Avec POP3.....	64
Avec IMAP.....	64
Maildrop.....	64
Conclusions.....	67
Améliorations possibles.....	67
Quand tout va mal.....	67

Daemontools et ucspi-tcp

Présentation

D.J.Bernstein a prévu quelques utilitaires ingénieux pour faire fonctionner son Qmail. Bien qu'il soit possible de se passer de ces utilitaires, il ne semble pas convenable de le faire, parce qu'ils apportent des avantages que l'on ne peut reproduire autrement.

Les daemontools

Présentation

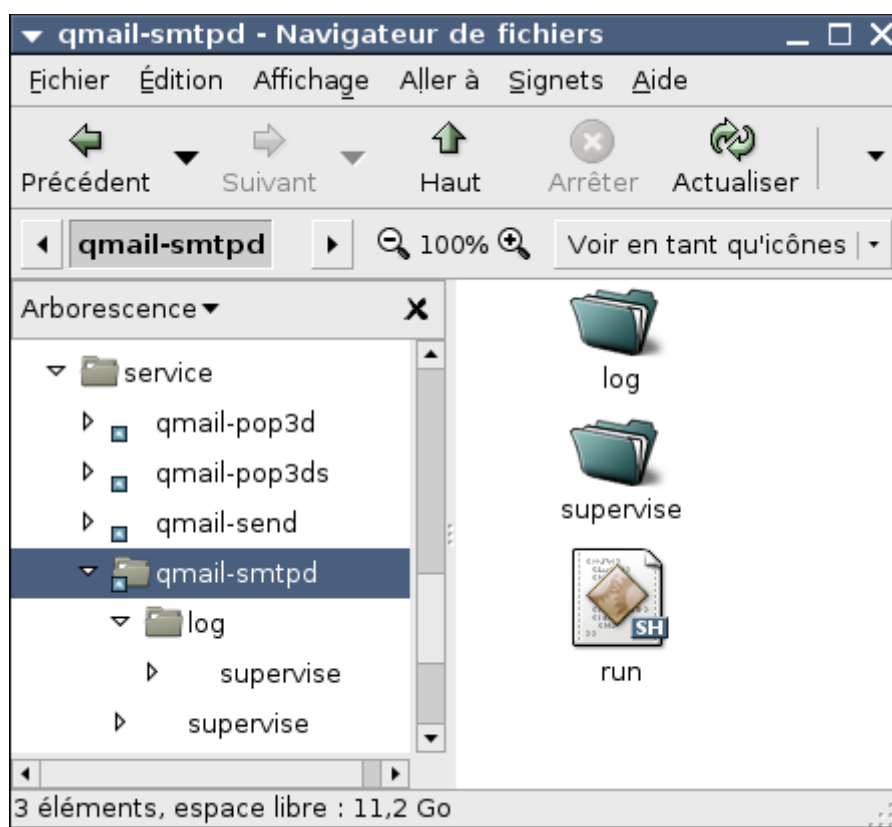
Les daemontools servent à superviser des services. Ils permettent de :

- démarrer un service,
- surveiller la vie de ce service,
- relancer un service qui meurt accidentellement,
- arrêter volontairement et redémarrer un service, si besoin est,
- gérer les logs de ces services.

Autrement dit, l'avantage par rapport à un script de démarrage classique (/etc/init.d/), c'est le redémarrage automatique d'un service qui meurt.

Fonctionnement

- Il doit exister un répertoire /service,
- tous les services à gérer disposent d'un répertoire dans /service (/service/mon_daemon, par exemple),
- dans ce répertoire, il doit exister un script nommé "run", qui définit le service à lancer (/service/mon_daemon/run),
- si l'on souhaite aussi gérer le log associé à ce service, il faut un sous-répertoire "log" contenant lui aussi un script "run" (/service/mon_daemon/log/run).



Les répertoires "supervise" seront créés automatiquement par les daemontools.

L'installation des daemontools ajoute une ligne au fichier "/etc/inittab", pour faire démarrer "svscanboot". Svscanboot démarre svscan. Svscan démarre un processus "supervise" pour chaque sous-répertoire contenu dans /service. C'est "supervise" qui exécute les scripts "run" et vérifie que les services sont toujours en vie.

Il y a bien d'autres outils dans les daemontools. Pour une liste complète, voyez <http://cr.yip.to/daemontools.html>.

Les logs sont gérés par "multilog".

Comme les services démarrés de cette façon deviennent quelque peu collants (un Kill n'aura qu'un effet très éphémère, le "supervise" qui y est attaché le relançant immédiatement), la commande "svc" sera d'un grand secours si l'on souhaite forcer l'arrêt d'un service et commander ensuite son redémarrage.

"Softlimit" permet de définir une limite de ressources de mémoire pour chaque processus lancé, afin d'éviter les "DoS".

"envdir" permet de définir un environnement particulier pour l'exécution d'un service.

"tai64nlocal" est un utilitaire dont je vous laisse la surprise. Son utilité vous apparaîtra certainement de façon évidente, lors de l'analyse des logs produits par multilog...

Nous aurons par la suite l'occasion de rencontrer ces composants des daemontools.

ucspi-tcp

Il s'agit principalement d'un "tcp server", à la manière de inetd ou de xinetd. Ses principaux avantages sont les suivants :

- une gestion fine des règles d'accès en fonction de l'adresse IP du client,
- le passage de "flags" au service considéré. Nous verrons que ceci est très utile pour définir le comportement de Qmail suivant l'adresse IP du client.

"tcpserver" a besoin d'un fichier au format .cdb (base de données). Ce fichier est construit à partir d'un simple fichier texte. L'outil qui permet de construire la base .cdb à partir du fichier texte est "tcprules".

Recommandations

Il est fort intéressant et utile de passer du temps à consulter les informations fournies par l'auteur² à propos de ces deux boîtes à outils.

² <http://cr.yp.to/>

Qmail & co.

Conditions d'expérimentation

Tout ce qui suit est essentiellement testé sur une Debian "testing" (Etch). Comme j'ai fait de multiples tests sur des machines différentes, il pourra se faire que dans le texte, apparaissent certaines incohérences concernant le nom des machines, ou quelques divergences mineures dans les copies d'écran, qui m'auront échappées. Je vous prie par avance de bien vouloir m'en excuser.

"La" machine de référence finale s'appelle "janus" et appartient à un domaine "virtuel" : maison.mrs

Premier problème

De très nombreux paquetages Debian font référence à un MTA. Comme il n'y en aura plus dans l'arbre des dépendances des paquetages (installation par les sources oblige), cela va causer pas mal de troubles. Aussi devons-nous d'abord nous arranger pour que l'arbre des dépendances sache qu'il y a tout de même un MTA d'installé.

Nous partons ici d'une installation tout à fait minimale, sans serveur http, sans base de données, sans FTP, juste avec Exim, puisqu'il est installé par défaut. Le paquetage "equivs" va nous venir en aide.

```
janus# apt-get install equivs
janus# cd /tmp
janus# cp /usr/share/doc/equivs/examples/mail-transport-agent.ctl .
janus# equivs-build mail-transport-agent.ctl
```

Ceci permet de créer un "faux paquetage" : mta-local_1.0_all.deb

```
janus# dpkg --ignore-depends=exim4 -r exim4
janus# dpkg --ignore-depends=exim4-daemon-light -r exim4-daemon-light
```

Ces deux lignes permettent de supprimer Exim sans enlever tous les éventuels paquetages dépendants.

```
janus# dpkg -i /tmp/mta-local_1.0_all.deb
```

Cette dernière ligne permet de placer notre faux paquetage qui fera croire au système apt qu'Exim est toujours présent. Ainsi, le reste de la distribution ne sera plus perturbé par le manque d'un paquetage de MTA, et les commandes apt fonctionneront normalement.

Qmail

On "patche" ou pas ?

L'objectif est de réaliser un système de messagerie complet avec la gestion :

- des domaines virtuels,
- des alias et redirections,
- des listes de diffusion,
- des virus et vers (suppression des messages en entrée du SMTP),
- des spams (marquage dans les en-têtes).

De plus, nous désirons que les clients puissent :

- accéder à leurs boîtes aux lettres via POP3, POP3S, IMAP4 et IMAP4S,
- utiliser le SMTP pour l'envoi de leurs mails (le serveur SMTP doit pouvoir authentifier le

client par son compte de messagerie).

Qmail/vpopmail/ezmlm dans leur forme initiale ne suffisent pas pour réaliser toutes ces opérations. Il nous faut donc patcher...

Un Qmail "amélioré"

Fort heureusement, il existe des ressources qui vont nous permettre de réaliser assez simplement la préparation de Qmail et son installation, mais aussi Vpopmail, pour la gestion des domaines virtuels, ezmlm, pour la gestion des listes de diffusions, et quelques autres accessoires que nous découvrirons en temps opportun.

Netqmail 1.05

Netqmail 1.05 se présente sous la forme d'une archive qui contient :

- qmail 1.03,
- un "patch" (netqmail-1.05.patch) qui permet principalement de connecter l'entrée de Qmail (qmail-queue) à des logiciels externes, ce qui nous servira pour la gestion des vers, virus et spams,
- quelques autres "patches" destinés à corriger le source pour qu'il soit compilable avec une glibc > 2.3,
- un script qui permet de simplifier la mise en place des sources "patchées".

Qmail-toaster-0.8.1.patch

Il s'agit d'un "patch" cumulatif qui permet, entre autres, d'utiliser TLS/SSL non seulement entre les clients et le serveur SMTP mais aussi entre SMTPs si c'est possible.

L'ensemble de ces ressources permettra de réaliser un serveur SMTP qui répondra complètement au cahier des charges.

Où prendre les ressources

Le moyen le plus simple est sans doute de consulter d'abord le site <http://shupp.org/toaster/> qui vous donnera des informations à jour.

Nous allons commencer, sous une session "root", par nous placer, par exemple, dans /usr/local/src. A l'heure où ces lignes sont écrites, la "liste des commissions" est la suivante :

```
janus# mkdir -p /usr/local/src/tar
janus# cd /usr/local/src/tar

janus# wget http://cr.yip.to/daemontools/daemontools-0.76.tar.gz
janus# wget http://cr.yip.to/ucspi-tcp/ucspi-tcp-0.88.tar.gz
janus# wget http://shupp.org/software/netqmail-1.05.tar.gz
janus# wget http://shupp.org/patches/qmail-toaster-0.8.1.patch.bz2
janus# wget http://shupp.org/software/vpopmail-5.4.13.tar.gz
janus# wget http://shupp.org/patches/vpopmail-5.4.13-cumulative-1.patch
janus# wget http://shupp.org/software/autorespond-2.0.4.tar.gz
janus# wget http://shupp.org/patches/autorespond-2.0.4-2.0.5.patch
janus# wget http://shupp.org/software/qmailadmin-1.2.9.tar.gz
janus# wget http://shupp.org/software/qmailadmin-help-1.0.8.tar.gz
janus# wget http://cr.yip.to/software/ezmlm-0.53.tar.gz
janus# wget http://shupp.org/software/ezmlm-idx-0.443.tar.gz
janus# wget http://shupp.org/software/courier-imap-4.0.6.tar.bz2
janus# wget http://shupp.org/software/courier-authlib-0.58.tar.bz2
```



```
janus# wget http://shupp.org/software/squirrelmail-1.4.5.tar.bz2
janus# wget http://shupp.org/software/quota_usage-1.3-1.2.7.tar.gz
janus# wget http://shupp.org/software/toaster-scripts-0.8.tar.gz
```

En réalité, tout ceci ne nous servira pas forcément. Dans cet exposé, nous utiliserons les paquetages sources de "courier" de la distribution (Debian Etch).

Installation

Assurez-vous que les outils nécessaires à la compilation sont installés :

```
janus# apt-get install libgdbm-dev gcc g++ patch make libc-dev stunnel libssl-dev apache2 php4 wget
bzip2
```

Préparation de netqmail

Extraction de l'archive, puis exécution du script "collate" qui va patcher Qmail :

```
janus# cd /usr/local/src
janus# tar -xzf tar/netqmail-1.05.tar.gz
janus# cd netqmail-1.05
janus# ./collate.sh
```

Installation des daemontools

Pour faire démarrer automatiquement un service, il est d'usage d'avoir recours aux scripts normalisés de /etc/init.d.

D.J.Bernstein a cependant créé un système, très différent, mais qui offre quelques avantages originaux, principalement la surveillance des services lancés (si un service s'arrête, pour une raison ou une autre, indépendante de l'administrateur, il sera automatiquement relancé).

De plus, une gestion assez fine des logs de ces services est assurée, sans avoir recours à l'habituel syslog.

L'ensemble de ces outils se trouve dans une archive nommée daemontools-0.76.tar.gz. Cette archive "officielle" est disponible sur le site de son concepteur : <http://cr.yp.to/daemontools/install.html>.

Il est bien entendu très fortement recommandé d'utiliser les daemontools

Les daemontools nécessitent, pour rester conforme aux spécifications de D.J.Bernstein, de créer un répertoire dans la racine, nommé "package" et d'y mettre les sources dedans. Nous en profiterons pour y appliquer le patch issu de netqmail :

```
janus# mkdir -p /package
janus# chmod 1755 /package
janus# cd /package
janus# tar -xpf /usr/local/src/tar/daemontools-0.76.tar.gz
janus# cd admin/daemontools-0.76
janus# patch -p1 < /usr/local/src/netqmail-1.05/other-patches/daemontools-0.76.errno.patch
```

Puis nous installons ces daemontools :

```
janus# ./package/install
```

Notez que l'installation va ajouter à votre fichier /etc/inittab la ligne :

```
SV:123456:respawn:/command/svscanboot
```

Cette ligne a pour but de maintenir éveillé un premier service : scanboot, qui lui-même va démarrer et maintenir un autre service : svscan.

Ce service va parcourir une arborescence que nous n'avons pas encore construite, à partir du répertoire /service.

Si tout c'est bien passé, une commande "*ps aux*" devrait donner ceci :

```
janus:/# ps aux | grep svscan
root 4477 0.0 0.4 2668 1264 ? Ss 09:49 0:00 /bin/sh /command/svscanboot
root 4494 0.0 0.1 1612 352 ? S 09:49 0:00 svscan /service
```

Le service "*svscan*", chaque fois qu'il trouvera ce qu'il attend (nous verrons quoi plus tard) dans l'arborescence */service*, lancera une instance de "*supervise*", destinée à maintenir en vie le service spécifié.

Installation de UCSPI-TCP

Qmail-smtpd, le service smtp de Qmail, n'est pas un serveur dit "standalone". Il s'appuie sur un système comme inetd ou xinetd. Ici encore, D.J.Bernstein a conçu un "tcp wrapper" sur mesure. Il est possible de faire fonctionner qmail-smtpd avec inetd ou xinetd, mais il est bien sûr fortement conseillé d'utiliser plutôt ucspi-tcp, d'autant qu'il va permettre une configuration beaucoup plus fine du fonctionnement de notre MTA, comme nous le verrons plus loin. En effet, ucspi-tcp va permettre :

- le contrôle des adresses IP des clients venant se connecter à qmail-smtps,
- le passage de "flags" à Qmail, suivant l'adresse IP du client. Le comportement de Qmail pourra se trouvé modifié suivant le client, grâce à ces flags.

ucspi-tcp est disponible également sur le site de leur concepteur : <http://cr.yip.to/ucspi-tcp/install.html>, mais encore une fois, nous avons déjà récupéré ces sources.

La procédure est assez similaire à celle des daemontools :

```
janus# cd /usr/local/src/
janus# tar -xzf tar/ucspi-tcp-0.88.tar.gz
janus# cd ucspi-tcp-0.88
janus# patch -p1 < /usr/local/src/netqmail-1.05/other-patches/ucspi-tcp-0.88.errno.patch
janus# make
janus# make setup check
```

Les binaires seront placés dans */usr/local/bin*. Ceux que nous manipulerons de façon explicite seront :

- tcpserver, qui est le "tcpwrapper",
- tcprules, qui permettra de mettre à jour les règles d'accès (ACL), à savoir, quelles adresses IP sont autorisées à se connecter à Qmail, et sous quelles conditions.

Pour mettre en place les règles, il faudra créer un fichier texte où on le souhaite, avec le nom qu'on souhaite. Nous ferons ceci plus tard.

Première installation de Qmail

Qmail a besoin de quelques utilisateurs sans privilèges, sous le nom desquels divers services s'exécuteront. Il est nécessaire de les créer avant d'installer Qmail. En effet, les UID et GID de ces utilisateurs seront inscrits "en dur" dans le code, de manière à éviter toute falsification ultérieure.

```
janus# groupadd nofiles
useradd -g nofiles -d /var/qmail/alias alias
useradd -g nofiles -d /var/qmail qmaild
useradd -g nofiles -d /var/qmail qmail1
useradd -g nofiles -d /var/qmail qmailp
janus# groupadd qmail
useradd -g qmail -d /var/qmail qmailq
useradd -g qmail -d /var/qmail qmailr
useradd -g qmail -d /var/qmail qmails
```

Pour respecter les désirs de D.J.B, Qmail sera installé dans le répertoire */var/qmail/* :

```
janus# mkdir /var/qmail
```

Enfin, nous installons une première fois Qmail, pour l'instant seulement "patché" par netqmail :

```
janus# cd /usr/local/src/netqmail-1.05/netqmail-1.05/
janus# make
janus# make setup check
```

Un petit moment après, /var/qmail sera peuplé de tout le code nécessaire :

```
drwxr-sr-x 2 alias qmail 6 2005-12-28 18:14 alias
drwxr-xr-x 2 root qmail 4096 2005-12-28 18:14 bin
drwxr-xr-x 2 root qmail 137 2005-12-28 18:14 boot
lrwxrwxrwx 1 root root 10 2005-12-28 17:37 control
drwxr-xr-x 2 root qmail 4096 2005-12-28 18:14 doc
drwxr-xr-x 10 root qmail 94 2005-12-28 18:14 man
drwxr-x--- 11 qmailq qmail 109 2005-12-28 18:14 queue
drwxr-xr-x 2 root qmail 6 2005-12-28 18:14 users
```

Un dernier script permettra de créer quelques points de configuration importants. Il s'agit d'indiquer à Qmail le nom pleinement qualifié de la machine hôte :

```
janus# /usr/local/src/netqmail-1.05/netqmail-1.05/config-fast janus.maison.mrs
```

(janus est dans le domaine fictif maison.mrs)

Ce script donne quelques informations sur ce qu'il fait :

```
Your fully qualified host name is janus.maison.mrs.
Putting janus.maison.mrs into control/me...
Putting bts.eme into control/defaultdomain...
Putting bts.eme into control/plusdomain...
Putting janus.maison.mrs into control/locals...
Putting janus.maison.mrs into control/rcpthosts...
Now qmail will refuse to accept SMTP messages except to janus.maison.mrs.
Make sure to change rcpthosts if you add hosts to locals or virtualdomains!
```

Pour le manuel (installé dans /var/qmail/man), ça ne va pas être bien pratique. Une solution sale mais efficace consiste à les copier dans l'arborescence de la distribution :

```
janus# for mandir in 1 5 7 8; do cp /var/qmail/man/man$mandir/* /usr/share/man/man$mandir/; done
```

Une solution nettement plus propre est d'agir sur le fichier /etc/manpath.config, en y ajoutant l'entrée :

```
MANDATORY MANPATH /var/qmail/man
```

(solution testée sur Debian Etch et sur Ubuntu 5.10)

A ce niveau, nous avons un Qmail qui devrait pouvoir fonctionner, à la condition de créer au moins un script, traditionnellement nommé :

```
/var/qmail/rc
```

Ce script permettra d'invoquer qmail-start. Qmail-start invoque qmail-send, qmail-lspawn, qmail-rspawn, et qmail-clean, sous leurs propres uids et gids. Ces quatre "daemons" coopèrent pour délivrer les messages depuis la file d'attente (queue).

Il y a plusieurs scripts proposés dans /var/qmail/boot, et dans les commissions que nous avons faites, il y en a un aussi :

```
janus# cd /usr/local/src
janus# tar -xzf tar/toaster-scripts-0.8.tar.gz
janus# cp toaster-scripts-0.8/rc /var/qmail/rc
```

Et rendre ce script exécutable :

```
janus# chmod 755 /var/qmail/rc
```

Ce script contient les lignes suivantes :

```
#!/bin/sh

# Using stdout for logging
# Using control/defaultdelivery from qmail-local to deliver messages by default

exec env - PATH="/var/qmail/bin:$PATH" \
qmail-start "`cat /var/qmail/control/defaultdelivery`"
```

Comme nous allons utiliser Vpopmail avec des boîtes aux lettres au format Maildir, nous allons

remplir le fichier `/var/qmail/control/defaultdelivery` comme ceci :

```
janus# echo ./Maildir/ >/var/qmail/control/defaultdelivery
```

Et sendmail ?

Beaucoup d'applications s'attendent à trouver un exécutable nommé "sendmail". Placé dans `/usr/sbin` et dans `/usr/lib` chez debian. Qmail en fournit un, mais dans `/var/qmail/bin`. Donc :

```
janus# ln -s /var/qmail/bin/sendmail /usr/sbin/sendmail
janus# ln -s /var/qmail/bin/sendmail /usr/lib/sendmail
```

Les logs

Nous allons utiliser qmail-log. Ce n'est pas une obligation, nous pourrions utiliser syslog, mais sur une installation classique de Qmail, nous nous attendons à trouver des logs gérés par qmail-log (vous verrez que ce n'est pas forcément un cadeau, mais avec un peu d'habitude, on s'y fait très bien). Qmail-log gère lui-même la rotation des logs, et il est d'usage de créer une arborescence dans `/var/log` comme ceci :

```
janus# mkdir -p /var/log/qmail/smtpd
janus# mkdir -p /var/log/qmail/pop3d
janus# mkdir -p /var/log/qmail/pop3ds
janus# chown -R qmail /var/log/qmail
```

Qmail-smtpd est le "daemon" qui reçoit les messages via SMTP. Il invoque qmail-queue pour les déposer dans la file d'attente. Ses logs iront dans `/var/log/qmail/smtpd`

A l'autre bout de la file d'attente, il y a qmail-send, qui transmettra les messages à qmail-lspawn ou à qmail-rspawn, suivant que le destinataire est local ou distant. Qmail-send génère des logs qui iront ici dans `/var/log/qmail`. Ce n'est peut-être pas très homogène, on préférerait éventuellement disposer d'un répertoire propre du genre `/var/log/qmail/send`. C'est faisable, mais en modifiant un script fourni dans nos commissions.

Qmail fournit qmail-pop3d et un patch permet d'utiliser aussi pop3ds. Ces deux serveurs pop enverront leurs logs dans leurs répertoires respectifs.

Démarrage des services

Nous allons devoir faire démarrer les divers services par l'entremise des daemontools. Nous avons déjà installé ces daemontools et le service svscan est déjà à l'affût.

Nous allons voir par l'exemple comment ceci fonctionne.

```
janus# mkdir -p /var/qmail/supervise/qmail-send/log
janus# mkdir -p /var/qmail/supervise/qmail-smtpd/log
janus# mkdir -p /var/qmail/supervise/qmail-pop3d/log
janus# mkdir -p /var/qmail/supervise/qmail-pop3ds/log
```

Nous commençons par créer un répertoire "supervise" dans `/var/qmail` qui, lui-même, contiendra un sous-répertoire par service à démarrer. Chacun de ces sous-répertoires contiendra un répertoire "log", pour le service des logs associé.

Ainsi, le répertoire `supervise/qmail-send` contiendra par la suite ce qu'il faut pour démarrer le service qmail-send et le répertoire `supervise/qmail-send/log` contiendra par la suite ce qu'il faut pour démarrer les logs associés à ce service.

L'élément déclencheur sera un script nommé "run". Les différents scripts "run" sont déjà présents dans nos commissions, il suffit de les recopier aux bons endroits :

```
janus# cp /usr/local/src/toaster-scripts-0.8/send.run /var/qmail/supervise/qmail-send/run
```

```
janus# cp /usr/local/src/toaster-scripts-0.8/smtpd.run /var/qmail/supervise/qmail-smtpd/run
janus# cp /usr/local/src/toaster-scripts-0.8/pop3d.run /var/qmail/supervise/qmail-pop3d/run
janus# cp /usr/local/src/toaster-scripts-0.8/pop3ds.run /var/qmail/supervise/qmail-pop3ds/run
```

Voyons le contenu de ces scripts :

qmail-send

```
janus# cat /var/qmail/supervise/qmail-send/run
```

```
#!/bin/sh
exec /var/qmail/rc
```

Un simple appel au script /var/qmail/rc que nous avons vu plus haut.

qmail-smtpd

```
janus# cat /var/qmail/supervise/qmail-smtpd/run
```

```
#!/bin/sh
QMAILDUID=`id -u vpopmail`
NOFILESGID=`id -g vpopmail`
MAXSMTPD=`cat /var/qmail/control/concurrencyincoming`
exec /usr/local/bin/softlimit -m 8000000 \
  /usr/local/bin/tcpserver -v -H -R -l 0 \
  -x /home/vpopmail/etc/tcp.smtp.cdb -c "$MAXSMTPD" \
  -u "$QMAILDUID" -g "$NOFILESGID" 0 smtp \
  /var/qmail/bin/qmail-smtpd \
  /home/vpopmail/bin/vchkpw /bin/true 2>&1
```

Ici, c'est franchement plus compliqué.

Déjà, nous voyons que nous aurons besoin de l'utilisateur et du groupe "vpopmail", que nous n'avons pas encore. Ce n'est pas une obligation pour l'instant. Nous pourrions utiliser qmaild, déjà créé, mais dans la configuration dont nous avons besoin pour répondre au cahier des charges initial, ça n'ira pas. Nous utilisons ici les services de vchkpw, qui ne peut être utilisé que par l'utilisateur vpopmail.

softlimit

Dont la documentation dit : "softlimit runs another program with new resource limits" (la doc officielle complète, comme seul D.J.B sait les faire, se trouve ici : <http://cr.yip.to/daemontools/softlimit.html>). En bref, il s'agit d'une protection destinée à éviter les "Deny of Service", en limitant les ressources allouées au processus appelé en cascade. Le processus en cascade n'est autre que...

tcpserver

Doc officielle : <http://cr.yip.to/ucspi-tcp/tcpserver.html> .

Vous constaterez à la lecture que vous n'avez pas vraiment le choix des options. Constatez que c'est ici que le fichier /home/vpopmail/etc/tcp.smtp.cdb est invoqué, de même que sont définis les utilisateur et groupe sous lesquels fonctionnera qmail-smtpd.

La variable MAXSMTPD prend sa valeur par la lecture du fichier /var/qmail/control/concurrencyincoming. Ce fichier n'est pas construit avec l'installation des daemontools ni de Qmail. Il faut donc le créer avant de l'oublier (on pourrait se passer de cette fonctionnalité, en écrivant une valeur "en dur" dans le script, mais ce qui est "en dur" n'est souvent pas très souple).

/var/qmail/control/concurrencyincoming n'existe pas encore, nous devons le créer :

```
janus# echo 20 > /var/qmail/control/concurrencyincoming
janus# chmod 644 /var/qmail/control/concurrencyincoming
```

Puis nous utiliserons "softlimit", "tcpserver", qui sont des exécutables fournis par ucspi-tcp, déjà installé, et enfin "vchkpw", qui fait partie de Vpopmail. Ce dernier assure l'authentification des clients, comme nous le verrons plus loin.

qmail-pop3d

```
janus# cat /var/qmail/supervise/qmail-pop3d/run
```

```
#!/bin/sh
VPOPMAILUID=`id -u vpopmail`
VPOPMAILGID=`id -g vpopmail`

exec /usr/local/bin/tcpserver -l 0 -R -H -v \
  -u"$VPOPMAILUID" -g"$VPOPMAILGID" 0 110 \
  /var/qmail/bin/qmail-popup `hostname` \
  /home/vpopmail/bin/vchkpw \
  /var/qmail/bin/qmail-pop3d Maildir 2>&1
```

Toujours "tcpserver", puis qmail-popup (qmail-popup lit un nom d'utilisateur et un mot de passe POP depuis le réseau), puis il exécute un sous-programme, qui est ici vchkpw, puis qmail-pop3d, le "daemon" POP3.

qmail-pop3ds

```
janus# cat /var/qmail/supervise/qmail-pop3ds/run
```

```
#!/bin/sh
VPOPMAILUID=`id -u vpopmail`
VPOPMAILGID=`id -g vpopmail`

exec /usr/local/bin/tcpserver -l 0 -R -H -v \
  -u"$VPOPMAILUID" -g"$VPOPMAILGID" 0 995 \
  /usr/sbin/stunnel -f -p /var/qmail/control/servercert.pem \
  -l /var/qmail/bin/qmail-popup -- qmail-popup `hostname` \
  /home/vpopmail/bin/vchkpw /var/qmail/bin/qmail-pop3d Maildir 2>&1
```

Assez semblable au précédent, si ce n'est qu'il y a ici "stunnel". Stunnel est un programme qui permet de chiffrer une connexion TCP quelconque dans SSL. Ici, une connexion POP3. Nous avons installé le paquetage stunnel de notre distribution au début de cette page.

Mise en place des scripts de gestion des logs

```
janus# cp /usr/local/src/toaster-scripts-0.8/send.log.run /var/qmail/supervise/qmail-send/log/run
janus# cp /usr/local/src/toaster-scripts-0.8/smtpd.log.run /var/qmail/supervise/qmail-smtpd/log/run
janus# cp /usr/local/src/toaster-scripts-0.8/pop3d.log.run /var/qmail/supervise/qmail-pop3d/log/run
janus# cp /usr/local/src/toaster-scripts-0.8/pop3ds.log.run /var/qmail/supervise/qmail-pop3ds/log/run
```

Ces quatre scripts ont à peu près la même forme :

```
janus# cat /var/qmail/supervise/qmail-send/log/run
```

```
#!/bin/sh
exec /usr/local/bin/setuidgid qmail /usr/local/bin/multilog t /var/log/qmail
```

Seul le répertoire qui doit contenir les logs change d'un script à l'autre.

Rendons tous ces scripts exécutables :

```

janus# chmod 755 /var/qmail/supervise/qmail-send/run
janus# chmod 755 /var/qmail/supervise/qmail-send/log/run
janus# chmod 755 /var/qmail/supervise/qmail-smtpd/run
janus# chmod 755 /var/qmail/supervise/qmail-smtpd/log/run
janus# chmod 755 /var/qmail/supervise/qmail-pop3d/run
janus# chmod 755 /var/qmail/supervise/qmail-pop3d/log/run
janus# chmod 755 /var/qmail/supervise/qmail-pop3ds/run
janus# chmod 755 /var/qmail/supervise/qmail-pop3ds/log/run

```

Presque tout est en place. En réalité, svscan observe la structure des services à superviser dans le répertoire /service, qui pour l'instant n'existe pas. Le moment venu, il suffira d'y créer des liens symboliques vers les répertoires qmail-* de /var/qmail/supervise.

Les daemontools gèrent les services de façon très "collante", c'est leur mission. Si un service meurt, il est automatiquement relancé. Comme il faut bien pouvoir arrêter ou démarrer un service si le besoin s'en fait sentir, la commande "svc" nous sera utile. Elle n'est pas très pratique à utiliser, aussi un script a été créé pour réaliser quelques actions courantes de façon simple. Ce script, qui s'appelle "qmailctl", fait aussi partie de nos commissions, nous allons le mettre en place :

```

janus# cp toaster-scripts-0.8/qmailctl /var/qmail/bin/
janus# chmod 755 /var/qmail/bin/qmailctl
janus# ln -s /var/qmail/bin/qmailctl /usr/bin

```

Voici ce script :

```

#!/bin/sh

# For Red Hat chkconfig
# chkconfig: - 30 80
# description: the qmail MTA

PATH=/var/qmail/bin:/bin:/usr/bin:/usr/local/bin:/usr/local/sbin
export PATH

QMAILDUID=`id -u qmaild`
NOFILESUID=`id -g qmaild`
svclist="qmail-send qmail-smtpd"

case "$1" in
  start)
    echo "Starting qmail"

    for svc in $svclist ; do
      if svok /service/$svc ; then
        svc -u /service/$svc
      else
        echo $svc service not running
      fi
    done

    if [ -d /var/lock/subsys ]; then
      touch /var/lock/subsys/qmail
    fi
    ;;
  stop)
    echo "Stopping qmail..."
    for svc in $svclist ; do
      echo " $svc"
      svc -d /service/$svc
    done
    if [ -f /var/lock/subsys/qmail ]; then
      rm /var/lock/subsys/qmail
    fi
    ;;
  stat)
    for svc in $svclist ; do
      svstat /service/$svc
      svstat /service/$svc/log
    done
    qmail-qstat
    ;;
)

```

```

doqueue|alarm|flush)
    echo "Sending ALRM signal to qmail-send."
    svc -a /service/qmail-send
    ;;
queue)
    qmail-qstat
    qmail-qread
    ;;
reload|hup)
    echo "Sending HUP signal to qmail-send."
    svc -h /service/qmail-send
    ;;
pause)
    for svc in $svclist ; do
        echo "Pausing $svc"
        svc -p /service/$svc
    done
    ;;
cont)
    for svc in $svclist ; do
        echo "Continuing $svc"
        svc -c /service/$svc
    done
    ;;
restart)
    echo "Restarting qmail:"
    for svc in $svclist ; do
        if [ "$svc" != "qmail-send" ] ; then
            echo "* Stopping $svc."
            svc -d /service/$svc
        fi
    done
    echo "* Sending qmail-send SIGTERM and restarting."
    svc -t /service/qmail-send
    for svc in $svclist ; do
        if [ "$svc" != "qmail-send" ] ; then
            echo "* Restarting $svc."
            svc -u /service/$svc
        fi
    done
    ;;
cdb)
    if ! grep '\#define POP_AUTH_OPEN_RELAY 1' ~vpopmail/include/config.h >/dev/null; then
        (cd ~vpopmail/etc ; cat tcp.smtp | tcprules tcp.smtp.cdb tcp.smtp.tmp)
        echo "Updated tcp.smtp.cdb."
    else
        ~vpopmail/bin/clearopensmtp
        echo "Ran clearopensmtp."
    fi
    ;;
help)
    cat <<HELP
    stop -- stops mail service (smtp connections refused, nothing goes out)
    start -- starts mail service (smtp connection accepted, mail can go out)
    pause -- temporarily stops mail service (connections accepted, nothing leaves)
    cont -- continues paused mail service
    stat -- displays status of mail service
    cdb -- rebuild the tcpserver cdb file for smtp
restart -- stops and restarts smtp, sends qmail-send a TERM & restarts it
doqueue -- sends qmail-send ALRM, scheduling queued messages for delivery
reload -- sends qmail-send HUP, rereading locals and virtualdomains
queue -- shows status of queue
alarm -- same as doqueue
flush -- same as doqueue
hup -- same as reload
HELP
    ;;
*)
    echo "Usage: $0 {start|stop|restart|doqueue|flush|reload|stat|pause|cont|cdb|queue|help}"
    exit 1
    ;;
esac

```



```
exit 0
```

Si vous avez eu le courage d'analyser en détail ce script, tant mieux. Sinon, nous aurons l'occasion de le voir à l'oeuvre plusieurs fois dans la suite de cet exposé.

Conclusion

Nous avons un Qmail en place et presque prêt à démarrer. Cependant, compte tenu du cahier des charges, nous devons encore installer Vpopmail, puis appliquer encore un patch à Qmail, et réinstaller ce dernier.

Vpopmail

Présentation rapide

Lorsque l'on met en place de gros serveurs de messagerie, les deux problèmes récurrents sont les suivants :

- comment gérer plusieurs domaines sur le même serveur ? (le cas des domaines virtuels est connu, les MTA savent généralement faire ce genre de chose, mais encore faut-il aussi disposer d'un moyen de stockage des messages),
- comment gérer des comptes d'utilisateurs de la messagerie, autrement qu'en créant autant de comptes "système" ? Un serveur, surtout public, moins il connaît d'utilisateurs, mieux il se porte, en général.

Vpopmail apporte à ces deux questions une solution élégante, efficace et "facile" à administrer. Bien qu'il soit possible d'utiliser Vpopmail avec postfix, il semble tout de même que le couple Qmail/vpopmail soit le plus harmonieux.

Vpopmail est maintenu par [Inter7³](http://www.inter7.com/), que nous aurons encore l'occasion de retrouver plus loin.

Domaines virtuels

Avec Vpopmail, une simple commande permet de créer un nouveau domaine virtuel, ainsi qu'un compte "postmaster" attaché à ce domaine. Une autre simple commande permet de créer un compte dans un domaine donné (et encore une autre permet de détruire un compte donné). Vpopmail va se greffer sur Qmail de façon tellement naturelle que l'ensemble va fonctionner presque "tout seul".

Authentification des utilisateurs

Vpopmail peut stocker les paramètres d'authentification des comptes de plusieurs manières :

- cdb (format de base de données déjà rencontré dans Qmail et ucspi-tcp),
- MySQL, Sybase ou Oracle, si l'on souhaite utiliser un moteur de base de données plus conséquent,
- LDAP, le service d'annuaire bien connu.

Sauf si vous avez à gérer plusieurs centaines de domaines, ou plusieurs centaines de comptes, le format cdb, qui ne fait pas intervenir un composant logiciel supplémentaire fera sans doute l'affaire, c'est ce cas que nous mettrons en pratique.

Nous l'avons vu lors de la première installation de Qmail, cette base de donnée des utilisateurs va servir non seulement pour POP3 et POP3s, mais aussi pour IMAP, et pour l'authentification des clients nomades qui voudront utiliser notre installation pour envoyer leurs messages.

Installation

Encore une fois, nous avons déjà ce qu'il faut dans nos commissions.

³ <http://www.inter7.com/>

Travaux préliminaires

Il y a quelques questions à se poser avant de se lancer dans la compilation et l'installation de Vpopmail. La première question est :

Où placer les domaines virtuels, avec leurs utilisateurs et leurs "Maildirs" ?

Habituellement, dans /home/vpopmail. Mais ce n'est absolument pas une obligation bien sûr. Dans cet exposé, c'est pendant ce choix qui est fait, c'est d'ailleurs plus ou moins la coutume.

Vpopmail nécessite la création d'un groupe et d'un utilisateur spécifiques, **avec des GID et UID spécifiques**. Assurez-vous que vous ne provoquerez pas de conflits, puis créez les :

```
janus# groupadd -g 89 vchkpw
janus# useradd -g vchkpw -u 89 -d /home/vpopmail vpopmail
```

```
janus# cd /usr/local/src
janus# tar -xzf tar/vpopmail-5.4.13.tar.gz
janus# cd vpopmail-5.4.13
janus# patch -p0 < ../tar/vpopmail-5.4.13-cumulative-1.patch
```

Configuration de la compilation

Une fois l'archive décompressée, allez dans le répertoire des sources de Vpopmail, ici :

/usr/local/src/vpopmail-5.4.13

Dans ce répertoire, il y a un certain nombre de README, qu'il ne peut qu'être conseillé de lire, et principalement, celui qui est relatif aux "roamingusers" (entendez par là, les utilisateurs nomades). Le problème évoqué ne concerne pas directement Vpopmail, mais plutôt Qmail, et se pose en ces termes :

un utilisateur "nomade" (ou même un sédentaire), change souvent d'adresse IP. Ce même utilisateur voudrait pouvoir disposer du serveur Qmail pour envoyer ses messages, mais nous savons que ceci est fixé par le "tcpserver". Comment faire alors pour permettre à un utilisateur dont l'adresse IP change, d'accéder à Qmail sans pour autant créer un relais ouvert ?

Une solution, qui n'est pas la meilleure, mais sans doute la plus simple pour ledit utilisateur, est d'avoir recours à ce que l'on appelle le "pop before smtp" (ou encore le "smtp after pop"). Le principe est simple, il suffit que l'utilisateur s'authentifie via POP3. Son adresse IP est récupérée, placée dans un fichier texte, puis le .cdb est reconstruit.

Il ne faut pas, bien entendu, que cette adresse IP reste éternellement dans la liste des adresses autorisées, aussi, une tâche "cron" devra faire le ménage régulièrement. Nous allons mettre en oeuvre cette méthode et nous verrons donc comment le faire en détail. Mais il faut déjà décider si l'on utilisera cette fonctionnalité avant de lancer la compilation.

L'autre solution, que nous adopterons ici, est que l'utilisateur s'authentifie auprès du serveur SMTP via TLS/SSL.

Il y a une grande quantité d'options de compilation, que l'on peut lister par la commande :

```
janus# ./configure --help
```

Les options surlignées :

- en vert nous intéressent, mais les valeurs par défaut correspondent à ce que l'on veut faire,
- en jaune nécessitent d'être données explicitement.

```
janus# cd /usr/local/src/vpopmail-5.4.13
janus# ./configure --help
```

```

'configure' configures vpopmail 5.4.13 to adapt to many kinds of systems.

Usage: ./configure [OPTION]... [VAR=VALUE]...

To assign environment variables (e.g., CC, CFLAGS...), specify them as
VAR=VALUE. See below for descriptions of some of the useful variables.

Defaults for the options are specified in brackets.

Configuration:
-h, --help                display this help and exit
  --help=short            display options specific to this package
  --help=recursive       display the short help of all the included packages
-V, --version            display version information and exit
-q, --quiet, --silent    do not print `checking...' messages
--cache-file=FILE       cache test results in FILE [disabled]
-C, --config-cache       alias for `--cache-file=config.cache'
-n, --no-create          do not create output files
  --srcdir=DIR           find the sources in DIR [configure dir or `..']

Installation directories:
--prefix=PREFIX          install architecture-independent files in PREFIX
                        [/usr/local]
--exec-prefix=EPREFIX    install architecture-dependent files in EPREFIX
                        [PREFIX]

By default, `make install' will install all the files in
`/usr/local/bin', `/usr/local/lib' etc. You can specify
an installation prefix other than `/usr/local' using `--prefix',
for instance `--prefix=$HOME'.

For better control, use the options below.

Fine tuning of the installation directories:
--bindir=DIR             user executables [EPREFIX/bin]
--sbindir=DIR           system admin executables [EPREFIX/sbin]
--libexecdir=DIR        program executables [EPREFIX/libexec]
--datadir=DIR           read-only architecture-independent data [PREFIX/share]
--sysconfdir=DIR        read-only single-machine data [PREFIX/etc]
--sharedstatedir=DIR    modifiable architecture-independent data [PREFIX/com]
--localstatedir=DIR     modifiable single-machine data [PREFIX/var]
--libdir=DIR            object code libraries [EPREFIX/lib]
--includedir=DIR        C header files [PREFIX/include]
--oldincludedir=DIR     C header files for non-gcc [/usr/include]
--infodir=DIR           info documentation [PREFIX/info]
--mandir=DIR            man documentation [PREFIX/man]

Program names:
--program-prefix=PREFIX prepend PREFIX to installed program names
--program-suffix=SUFFIX  append SUFFIX to installed program names
--program-transform-name=PROGRAM run sed PROGRAM on installed program names

System types:
--build=BUILD           configure for building on BUILD [guessed]
--host=HOST             cross-compile to build programs to run on HOST [BUILD]

Optional Features:
--disable-FEATURE      do not include FEATURE (same as --enable-FEATURE=no)
--enable-FEATURE[=ARG] include FEATURE [ARG=yes]
--disable-dependency-tracking Speeds up one-time builds
--enable-dependency-tracking Do not reject slow dependency extractors
--enable-qmaildir=DIR  Directory where qmail control and user directories are installed
                        /var/qmail.
--enable-qmail-newu=PATH Full path to qmail-newu program.
--enable-qmail-inject=PATH Full path to qmail-inject program.
--enable-qmail-newmrh=PATH Full path to qmail-newmrh program.
--enable-vpopuser=USER User name allocated to vpopmail vpopmail.
--enable-vpopgroup=GROUP Group name allocated to vpopmail vchkpw.
--enable-roaming-users Enable POP-before-SMTP functionality.
--enable-tcprules-prog=PATH Full path to tcprules program /usr/{local/}bin/tcprules.
--enable-tcpsrvr-file=PATH File where tcpserver -x relay information is stored
                        /home/vpopmail/etc/tcp.smtp.

```

```

--disable-rebuild-tcpserver-file  Disable rebuilding of tcpserver relay control file.
--enable-relay-clear-minutes=#    Expire time for roaming users after pop authentication [180].
--enable-learn-passwords         If no password is stored for a user, learn it the first time
                                they authenticate.
--disable-md5-passwords         Use DES crypt() instead of MD5 encryption for passwords.
--disable-file-locking           Don't use file locking.
--enable-file-sync               Enable file sync after each message is delivered.
--disable-make-seekable         Don't try to make input to vdelivermail seekable.
--disable-clear-passwd          Don't store a cleartext version of the password in addition to
                                the encrypted version.
--disable-users-big-dir         Disable hashing of user directories.
--enable-sqwebmail-pass        Store a copy of the user's password in the the user's maildir
                                for use
                                by pre-v3 sqwebmail.
--enable-qmail-ext              Enable qmail email address extension support.
--enable-ip-alias-domains       Enable mapping of default domain via reverse ip lookup table.
--enable-domains-dir=TEXT       Set domains directory name for ~vpopmail/domains/user/.
--enable-incdir=DIR             Your MySQL/Oracle 'include' directory.
--enable-libdir=DIR             Your MySQL/Oracle 'lib' directory.
--enable-auth-module=MOD        Nominate how to store the vpopmail account information (cdb
                                (default), mysql, pgsq, ldap, oracle, sybase, or activedir).
--enable-passwd                 Enable /etc/passwd (or shadow) accounts in addition to virtual
                                domains.
--enable-logging=OPT           Log to syslog: n=nothing, e=errors only (default),
                                y=all attempts, p=errors with passwords,
                                v=verbose (all attempts, with passwords).
--enable-log-name=TEXT         Set syslog name vpopmail.
--disable-auth-logging          Don't record time and ip of last auth attempt. Valid only for
                                CDB, MySQL, PGSQL, LDAP, ActiveDir.
--enable-sql-logging           Enable authentication logging to MySQL/Postgres.
--enable-mysql-limits          Use MySQL to store limits instead of .qmailadmin-limits files.
--enable-mysql-replication     Enable support for replicated MySQL auth servers.
--enable-valias                Store email aliases in MySQL.
--disable-many-domains         Creates a table for each virtual domain instead of storing all
                                users in a single table. Only valid for MySQL and PostgreSQL

Some influential environment variables:
CC          C compiler command
CFLAGS      C compiler flags
LDFLAGS     linker flags, e.g. -L<lib dir> if you have libraries in a
            nonstandard directory <lib dir>
CPPFLAGS    C/C++ preprocessor flags, e.g. -I<include dir> if you have
            headers in a nonstandard directory <include dir>
CPP         C preprocessor

Use these variables to override the choices made by `configure' or to help
it to find libraries and programs with nonstandard names/locations.

```

Comme ce sont des options de compilation, il faut bien réfléchir à ce que l'on veut faire et aussi à ce que l'on ne veut pas faire. En effet, un oubli ou une erreur se traduiront obligatoirement par une recompilation.

Voici la liste des options retenues dans cet exposé :

```
--prefix=/home/vpopmail --enable-logging=v
```

Allons-y :

```
janus# ./configure \
--prefix=/home/vpopmail \
--enable-logging=v
```

Si tout se passe bien, en fin de configuration, vous devez obtenir quelque chose de la forme :

```

vpopmail 5.4.13
Current settings
-----

vpopmail directory = /home/vpopmail
uid = 89
gid = 89
roaming users = OFF --disable-roaming-users (default)
password learning = OFF --disable-learn-passwords (default)

```

```

md5 passwords = ON --enable-md5-passwords (default)
file locking = ON --enable-file-locking (default)
vdelivermail fsync = OFF --disable-file-sync (default)
make seekable = ON --enable-make-seekable (default)
clear passwd = ON --enable-clear-passwd (default)
user dir hashing = ON --enable-users-big-dir (default)
address extensions = OFF --disable-qmail-ext (default)
ip alias = OFF --disable-ip-alias-domains (default)
auth module = cdb --enable-auth-module=cdb (default)
auth inc = -Icdb
auth lib =
system passwords = OFF --disable-passwd (default)
pop syslog = log success and errors including passwords
--enable-logging = v
auth logging = ON --enable-auth-logging (default)

```

Une fois la configuration terminée, il faut compiler et installer :

```

janus# make
janus# make install-strip

```

Un petit moment après, Vpopmail devrait être installé dans /home/vpopmail

```

janus# ls -l /home/vpopmail
total 4
drwxr-xr-x 2 vpopmail vchkpw 4096 2006-01-01 16:55 bin
drwxr-xr-x 4 vpopmail vchkpw 36 2006-01-01 16:55 doc
drwx----- 2 vpopmail vchkpw 6 2006-01-01 16:55 domains
drwxr-xr-x 2 vpopmail vchkpw 58 2006-01-01 16:55 etc
drwxr-xr-x 2 vpopmail vchkpw 92 2006-01-01 16:55 include
drwxr-xr-x 2 vpopmail vchkpw 26 2006-01-01 16:55 lib

```

Bon, les binaires ne seront pas dans le PATH. Ça peut s'arranger, mais nous verrons que ce n'est pas si grave. Les commandes de Vpopmail ne servent pas tous les jours et seront manipulées soit par l'administrateur, soit par un script cgi qu'il suffira de construire correctement.

Un premier domaine virtuel

Avant de toucher à quoi que ce soit, voyons le contenu de /var/qmail/control :

```

janus# ls -l /var/qmail/control/
total 28
-rw-r--r-- 1 root root 3 2006-02-25 16:00 concurrencyincoming
-rw-r--r-- 1 root root 11 2006-02-25 15:58 defaultdelivery
-rw-r--r-- 1 root root 11 2006-02-25 15:56 defaultdomain
-rw-r--r-- 1 root root 17 2006-02-25 15:56 locals
-rw-r--r-- 1 root root 17 2006-02-25 15:56 me
-rw-r--r-- 1 root root 11 2006-02-25 15:56 plusdomain
-rw-r--r-- 1 root root 17 2006-02-25 15:56 rcpthosts

```

Et créons un nouveau domaine :

```

janus# cd /home/vpopmail/bin
janus# ./vadddomain maison.mrs
Please enter password for postmaster:
enter password again:

```

La commande vadddomain permet de créer un nouveau domaine virtuel (ici maison.mrs). Dans la foulée, il est demandé un mot de passe pour le "postmaster" de ce domaine. Allons voir...

```

janus# cd /home/vpopmail/domains/
janus# ls
maison.mrs

janus# cd maison.mrs/
janus# ls
postmaster vpasswd vpasswd.cdb

```

Nous avons bien un répertoire "maison.mrs" et dedans, un répertoire "postmaster".

```

janus# cd postmaster/
janus# ls
lastauth Maildir

```

```
janus# cd Maildir/
janus# ls
cur new tmp
```

postmaster dispose d'un répertoire "Maildir" avec déjà tout ce qu'il faut dedans.

Voyons du côté de Qmail si des choses ont changé dans le répertoire de contrôle :

```
janus# cd /var/qmail/control
janus# ls -l
total 32
-rw-r--r-- 1 root root 3 2006-02-25 16:00 concurrencyincoming
-rw-r--r-- 1 root root 11 2006-02-25 15:58 defaultdelivery
-rw-r--r-- 1 root root 11 2006-02-25 15:56 defaultdomain
-rw-r--r-- 1 root root 17 2006-02-25 16:22 locals
-rw----- 1 root root 0 2006-02-25 16:22 locals.lock
-rw-r--r-- 1 root root 17 2006-02-25 15:56 me
-rw-r--r-- 1 root root 11 2006-02-25 15:56 plusdomain
-rw-r--r-- 1 root root 28 2006-02-25 16:22 rcpthosts
-rw----- 1 root root 0 2006-02-25 16:22 rcpthosts.lock
-rw-r--r-- 1 root root 22 2006-02-25 16:22 virtualdomains
-rw----- 1 root root 0 2006-02-25 16:22 virtualdomains.lock
```

Visiblement locals et rcpthosts ont été modifiés, virtualdomains a été ajouté. Plutôt bon signe...

```
janus# cat locals
janus.maison.mrs

janus# cat rcpthosts
janus.maison.mrs
maison.mrs

janus# cat virtualdomains
maison.mrs:maison.mrs
```

Rien de nouveau dans locals, mais le nouveau domaine apparaît dans les deux autres fichiers.

Qmail, installation définitive

Maintenant que Vpopmail est en place, nous allons pouvoir en finir avec Qmail.

TCPSEVER

Commençons par indiquer au tcpserver que la machine locale est autorisée à envoyer des messages :

```
janus# echo '127.0.0.1:allow,RELAYCLIENT=""' > ~vpopmail/etc/tcp.smtp
janus# echo ':deny' >> ~vpopmail/etc/tcp.smtp
```

En français, cela veut dire que l'adresse "locale" est autorisée à se connecter au serveur, et qu'elle dispose du "flag" RELAYCLIENT, c'est à dire que les messages issus de cette adresse peuvent être relayés, la seconde ligne indique que toutes les autres adresses IP sont refusées.

Il faudra y revenir par la suite, en fonction de ce que nous voulons autoriser ou non.

N'oublions pas que /home/vpopmail/etc/tcp.smtp n'est qu'un fichier texte qui doit servir à construire une base de données au format cdb. C'est le moment d'utiliser notre script qmailctl :

```
janus# qmailctl cdb
et vérifions le résultat :

janus# ls ~vpopmail/etc/
inc_deps lib_deps tcp.smtp tcp.smtp.cdb vlimits.default
```

Ça fonctionne.

Réinstallation de Qmail

Un dernier "patch" pour Qmail :

```
janus# cd /usr/local/src/netqmail-1.05/netqmail-1.05
janus# bunzip2 -c ../../tar/qmail-toaster-0.8.1.patch.bz2 | patch -p0
```

Puis une reconstruction et une réinstallation :

```
janus# make clean
janus# make
janus# qmailctl stop
janus# make setup check
```

SPF est quelque chose comme un système "reverse MX" basé sur DNS pour définir des auteurs de messages autorisés en fonction du nom de domaine inscrit dans leur adresse d'émetteur. L'objectif est d'interdire des "senders" dont l'adresse est falsifiée.

spfbehavior

Utilisé pour activer la recherche SPF. La valeur par défaut est 0 (off).

Une valeur entre 0 et 6 est possible :

- 0 : Inactif, ne crée pas d'en-tête Received-SPF
- 1 : Crée seulement l'en-tête Received-SPF, non bloquant.
- 2 : Utilise "erreur temporaire" quand il y a des problèmes de résolution DNS.
- 3 : Rejette les mails lorsque SPF résout comme *fail (deny)*
- 4 : Rejette les mails lorsque SPF résout comme *softfail*
- 5 : Rejette les mails lorsque SPF résout comme *neutral*
- 6 : Rejette les mails lorsque SPF ne résout pas comme *pass*

Les valeurs plus grandes que 3 sont fortement déconseillées, vous voudrez probablement fonctionner avec 2 or 3.

Important: Ce paramètre peut être surchargé en utilisant la variable d'environnement *SPFBEHAVIOR*, par exemple depuis les règles du tcpserver.

Note : Si *RELAYCLIENT* est mis, les recherches SPF sont désactivées.
(Ceci inclue aussi SMTP-AUTH et les patches similaires)

```
janus# echo 3 > /var/qmail/control/spfbehavior
```

Démarrage du système

Nous pouvons maintenant démarrer les services de Qmail :

```
janus# ln -s /var/qmail/supervise/qmail-send /service
```

Et vérifier que les services concernés ont bien démarré (il faut attendre quelques secondes, tout de même) :

```
janus# ps -aef | grep qmail
root  12319  5018  0 17:26 ? 00:00:00 supervise qmail-send
qmail1 12321 12320  0 17:26 ? 00:00:00 /usr/local/bin/multilog t /var/log/qmail
qmails 12322 12319  0 17:26 ? 00:00:00 qmail-send
root  12324 12322  0 17:26 ? 00:00:00 qmail-lspawn ./Maildir/
qmailr 12325 12322  0 17:26 ? 00:00:00 qmail-rspawn
qmailq 12326 12322  0 17:26 ? 00:00:00 qmail-clean
```


Ca fonctionne. Notez que le service de logs a démarré lui aussi. Nous pouvons ajouter les liens pour smtp, pop3 et pop3s :

```
janus# ln -s /var/qmail/supervise/qmail-smtpd /service
janus# ln -s /var/qmail/supervise/qmail-pop3d /service
janus# ln -s /var/qmail/supervise/qmail-pop3ds /service
```

Je vous laisse refaire un ps -aef pour vérifier, si vous êtes d'un naturel incrédule.

Voyons tout de même si les ports intéressants (smtp, pop3 et pop3s) sont bien à l'écoute :

```
janus# netqmail-1.05# netstat -a | grep :smtp
tcp        0      0 *:smtp          **          LISTEN

janus# netstat -a | grep :pop3
tcp        0      0 *:pop3s         **          LISTEN
tcp        0      0 *:pop3          **          LISTEN
```

Dans nos commissions, nous avons ramené un script pour gérer Vpopmail :

```
janus# cp /usr/local/src/toaster-scripts-0.8/vpopmailctl /var/qmail/bin/vpopmailctl
janus# chmod 755 /var/qmail/bin/vpopmailctl
janus# ln -s /var/qmail/bin/vpopmailctl /usr/bin
```

Mais à quoi sert ce script ?

```
janus# cat /usr/bin/vpopmailctl
```

```
#!/bin/sh

# For Red Hat chkconfig
# chkconfig: - 30 80
# description: the qmail-pop3d server

PATH=/var/qmail/bin:/bin:/usr/bin:/usr/local/bin:/usr/local/sbin
export PATH

QMAILDUID=`id -u vpopmail`
NOFILESGID=`id -g vpopmail`

case "$1" in
  start)
    echo "Starting qmail-pop3d"
    if svok /service/qmail-pop3d; then
      svc -u /service/qmail-pop3d
    else
      echo qmail-pop3d service not running
    fi
    echo "Starting qmail-pop3ds"
    if svok /service/qmail-pop3ds; then
      svc -u /service/qmail-pop3ds
    else
      echo qmail-pop3ds service not running
    fi
    if [ -d /var/lock/subsys ]; then
      touch /var/lock/subsys/vpopmail
    fi
    ;;
  stop)
    echo "Stopping qmail-pop3d..."
    svc -d /service/qmail-pop3d
    echo "Stopping qmail-pop3ds..."
    svc -d /service/qmail-pop3ds
    if [ -f /var/lock/subsys/vpopmail ]; then
      rm /var/lock/subsys/vpopmail
    fi
    ;;
  stat)
    svstat /service/qmail-pop3d
    svstat /service/qmail-pop3d/log
    svstat /service/qmail-pop3ds
    svstat /service/qmail-pop3ds/log
    ;;
  *)
    ;;
esac
```

```

pause)
  echo "Pausing qmail-pop3d"
  svc -p /service/qmail-pop3d
  echo "Pausing qmail-pop3ds"
  svc -p /service/qmail-pop3ds
  ;;
cont)
  echo "Continuing qmail-pop3d"
  svc -c /service/qmail-pop3d
  echo "Continuing qmail-pop3ds"
  svc -c /service/qmail-pop3ds
  ;;
restart)
  echo "Stopping qmail-pop3d:"
  echo "* Stopping qmail-pop3d."
  echo "Stopping qmail-pop3ds:"
  svc -d /service/qmail-pop3d
  svc -d /service/qmail-pop3ds
  echo "* Restarting qmail-pop3d."
  svc -u /service/qmail-pop3d
  echo "* Restarting qmail-pop3ds."
  svc -u /service/qmail-pop3ds
  ;;
help)
  cat <<HELP
  stop -- stops mail service (pop3 connections refused)
  start -- starts mail service (pop3 connection accepted)
  pause -- temporarily stops pop3 service
  cont -- continues paused pop3 service
  stat -- displays status of pop3 service
restart -- stops and restarts pop3d
HELP
  ;;
*)
  echo "Usage: $0 {start|stop|restart|stat|pause|cont|help}"
  exit 1
  ;;
esac

exit 0

```

Pas besoin d'être un expert en shell pour comprendre que vpopmailctl agit sur les serveurs pop3 et pop3s, à la manière de qmailctl, qui agit, entre autres, sur les services smtp.

```

janus# qmailctl stat
/service/qmail-send: up (pid 12322) 1450 seconds
/service/qmail-send/log: up (pid 12321) 1450 seconds
/service/qmail-smtpd: up (pid 12333) 1115 seconds
/service/qmail-smtpd/log: up (pid 12337) 1115 seconds
messages in queue: 0
messages in queue but not yet preprocessed: 0

janus# vpopmailctl stat
/service/qmail-pop3d: up (pid 12340) 1127 seconds
/service/qmail-pop3d/log: up (pid 12344) 1127 seconds
/service/qmail-pop3ds: up (pid 12348) 1122 seconds
/service/qmail-pop3ds/log: up (pid 12350) 1122 seconds

```

Notes importantes...

Il reste quelques détails qu'il ne faut surtout pas oublier de régler.

Dans cette configuration, nous avons changé, par rapport à l'installation "normale" l'utilisateur qui fait tourner qmail-smtpd. Ce n'est pas qmaild, mais Vpopmail. La réinstallation de Qmail nécessite cette modification :

```
janus# chown -R vpopmail:vchkpw /var/qmail/spam
```

De plus, nous utilisons TLS/SSL en divers endroits (stunnel pour pop3, authentification TLS sur qmail-smtpd...), mais TLS/SSL sans certificats, ça risque de moins bien fonctionner...

Le Makefile situé dans /usr/local/src/netqmail-1.05/netqmail-1.05 permet de le réaliser :

```
janus# cd /usr/local/src/netqmail-1.05/netqmail-1.05
janus# make cert

Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to '/var/qmail/control/servercert.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:France
Locality Name (eg, city) []:Marseille
Organization Name (eg, company) [Internet Widgits Pty Ltd]:maison
Organizational Unit Name (eg, section) []:r&d
Common Name (eg, YOUR name) []:janus.maison.mrs
Email Address []:postmaster@maison.mrs

make tmprsadh
echo "Creating new temporary RSA and DH parameters"
Creating new temporary RSA and DH parameters
./update_tmprsadh
Generating RSA private key, 512 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)

Generating DH parameters, 512 bit long safe prime, generator 2
This is going to take a long time
.....+.....+.....+...etc...

Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.....+.....+.....+...etc...
```

Bien entendu, l'idéal serait, en production, de disposer de "vrais" certificats (garantis par une autorité).

Enfin, prévoyez de mettre dans votre "crontab" la ligne suivante, pour régénérer cette clé temporaire :

```
01 01 * * * /var/qmail/bin/update_tmprsadh > /dev/null 2>&1
```

Ajoutons un utilisateur

```
janus# cd /home/vpopmail/bin/
janus# ./vadduser prof@maison.mrs
Please enter password for prof@maison.mrs:
enter password again:

janus# cd /home/vpopmail/domains/maison.mrs/
janus# ls
postmaster prof vpasswd vpasswd.cdb
```

Le répertoire du nouvel utilisateur est créé. Voyons vpasswd :

```
janus# cat /home/vpopmail/domains/maison.mr/vpasswd
postmaster:$1$uo.KFcKt$km3r1DxqSN6CZP016TPsu.:1:0:Postmaster:
/home/vpopmail/domains/maison.mrs/postmaster:NOQUOTA:password
prof:$1$IiZONksF$bV3K4FcNR0Zw/eF7b2/ZH0:1:0:prof:
/home/vpopmail/domains/maison.mrs/prof:NOQUOTA:password
```

Le "password" en fin de ligne, c'est le mot de passe du compte (peu original et ambigu, je l'avoue). Le mot de passe apparaît en clair, on a compilé Vpopmail avec cette option. On peut faire autrement, mais comme l'on sait que les utilisateurs peuvent oublier leur mot de passe...

Le lecteur attentif aura constaté qu'il n'y a pas de quota sur ces boîtes. Il est possible, bien sûr d'en fixer un, avec la commande :

vsetuserquota [email address] [quota in bytes]

```
janus# ./vsetuserquota prof@maison.mrs 1000000
janus# cat /home/vpopmail/domains/maison.mrs/vpasswd
postmaster:$1$uo.KFcKT$m3r1DxqSN6CZP016TPsu.:1:0:Postmaster:
/home/vpopmail/domains/maison.mrs/postmaster:NOQUOTA:password
prof:$1$IiZONKsF$bV3K4FcNR0Zw/eF7b2/ZH0:1:0:prof:
/home/vpopmail/domains/maison.mrs/prof:1000000S:password
```

Administration de Vpopmail

Sans se perdre dans les détails de toutes les options possibles, disons que basiquement, Vpopmail va vous permettre de :

- créer des domaines "virtuels" pour l'hébergement de comptes de messagerie,
- attribuer éventuellement un quota global à ce domaine (volume de données global attribué à l'ensemble des comptes de ce domaine),
- créer des comptes dans un domaine virtuel, avec éventuellement un quota pour chaque compte,
- créer des alias ou des redirections...

Bien entendu, chaque domaine virtuel devra disposer d'un enregistrement de type MX sur les DNS faisant autorité pour le domaine en question.

Bien sûr, il est possible d'administrer complètement Vpopmail à partir de la ligne de commande. Inter7 a tout de même créé un script CGI, vqadmin, qui permet de le faire avec plus d'ergonomie. Nous verrons plus loin qu'un autre script, qmailadmin, permet de gérer les comptes dans chaque domaine virtuel.

Il nous faudra un serveur apache. Il le faudra aussi pour qmailadmin et pour le webmail.

Installation de vqadmin

Toujours à partir des sources, que l'on peut trouver chez inter7⁴. En effet, ces sources là ne sont pas dans nos commissions. Après décompression dans /usr/local/src/vqadmin-2.3.2, nous pouvons consulter la liste des options de configuration :

```
janus# cd /usr/local/src/vqadmin-2.3.2
janus# ./configure --help
Usage: configure [options] [host]
Options: [defaults in brackets after descriptions]
Configuration:
  --cache-file=FILE      cache test results in FILE
  --help                 print this message
  --no-create            do not create output files
  --quiet, --silent     do not print `checking...' messages
  --version              print the version of autoconf that created configure
Directory and file names:
  --prefix=PREFIX       install architecture-independent files in PREFIX
                        [/usr/local]
  --exec-prefix=EPREFIX install architecture-dependent files in EPREFIX
                        [same as prefix]
  --bindir=DIR          user executables in DIR [EPREFIX/bin]
  --sbindir=DIR         system admin executables in DIR [EPREFIX/sbin]
  --libexecdir=DIR     program executables in DIR [EPREFIX/libexec]
  --datadir=DIR         read-only architecture-independent data in DIR
                        [PREFIX/share]
```

4 <http://www.inter7.com/>

```

--sysconfdir=DIR      read-only single-machine data in DIR [PREFIX/etc]
--sharedstatedir=DIR  modifiable architecture-independent data in DIR
                     [PREFIX/com]
--localstatedir=DIR   modifiable single-machine data in DIR [PREFIX/var]
--libdir=DIR          object code libraries in DIR [EPREFIX/lib]
--includedir=DIR      C header files in DIR [PREFIX/include]
--oldincludedir=DIR   C header files for non-gcc in DIR [/usr/include]
--infodir=DIR         info documentation in DIR [PREFIX/info]
--mandir=DIR          man documentation in DIR [PREFIX/man]
--srcdir=DIR          find the sources in DIR [configure dir or ..]
--program-prefix=PREFIX prepend PREFIX to installed program names
--program-suffix=SUFFIX append SUFFIX to installed program names
--program-transform-name=PROGRAM
                     run sed PROGRAM on installed program names

Host type:
--build=BUILD         configure for building on BUILD [BUILD=HOST]
--host=HOST           configure for HOST [guessed]
--target=TARGET       configure for TARGET [TARGET=HOST]

Features and packages:
--disable-FEATURE     do not include FEATURE (same as --enable-FEATURE=no)
--enable-FEATURE[=ARG] include FEATURE [ARG=yes]
--with-PACKAGE[=ARG]  use PACKAGE [ARG=yes]
--without-PACKAGE     do not use PACKAGE (same as --with-PACKAGE=no)
--x-includes=DIR      X include files are in DIR
--x-libraries=DIR     X library files are in DIR
--enable and --with options recognized:
--enable-qmaildir=dir directory where qmail is installed.
--enable-vpopuser=vpopmail user vchkpw was installed as.
--enable-vpopgroup=vchkpw group vchkpw was installed as.
--enable-cgibindir={dir} HTTP server's cgi-bin directory.

```

qmaildir est, dans notre installation, le chemin "normal", mais mieux vaut tout de même le spécifier. En ce qui concerne le chemin d'accès aux scripts cgi, tout dépend de votre distribution et de la configuration de votre apache. Sur la Debian avec apache2, il s'agit du répertoire /usr/lib/cgi-bin par défaut.

```

janus# ./configure \
--enable-cgibindir=/usr/lib/cgi-bin/ \
--enable-qmaildir=/var/qmail/

```

Après configuration, le petit résumé vous indique :

```

Current settings
-----
vpopmail directory = /home/vpopmail
      uid = 89
      gid = 89
      cgi-bin dir = /usr/lib/cgi-bin/
      vqadmin dir = /usr/lib/cgi-bin//vqadmin

```

Reste à compiler et à installer :

```

janus# make
janus# make install-strip

```

Configuration d'Apache

Ajoutez au fichier de configuration de votre apache un paragraphe de ce genre (à modifier éventuellement, en fonction de votre configuration) :

```

<Directory "/usr/lib/cgi-bin/vqadmin">
  deny from all
  Options ExecCGI
  AllowOverride AuthConfig
  Order deny,allow
</Directory>

```

Créez dans le répertoire cgi-bin/vqadmin un fichier .htaccess :

```

AuthType Basic
AuthUserFile /etc/apache2/vqadmin.passwd
AuthName vqadmin
require valid-user

```

```
satisfy any
```

A modifier, toujours éventuellement, suivant votre configuration. Le fichier des mots de passe devra se trouver dans un répertoire non accessible à la navigation. Le répertoire contenant la configuration d'apache semble un bon choix.

Créer le fichier de mots de passe /etc/apache2/vqadmin.passwd :

```
janus# htpasswd -c /etc/apache2/vqadmin.passwd postmaster
New password:
Re-type new password:
Adding password for user postmaster
```

vqadmin permet plusieurs niveaux de privilèges dans l'administration. Nous faisons ici quelque chose de très basique, où un seul administrateur (postmaster) pourra accéder à l'interface. Il est bien entendu possible d'en mettre plusieurs.

Enfin, ne pas oublier de modifier le fichier cgi-bin/vqadmin/vqadmin.acl pour accorder tous les droits à postmaster :

```
janus# cd /usr/lib/cgi-bin/vqadmin
janus# cat vqadmin.acl
```

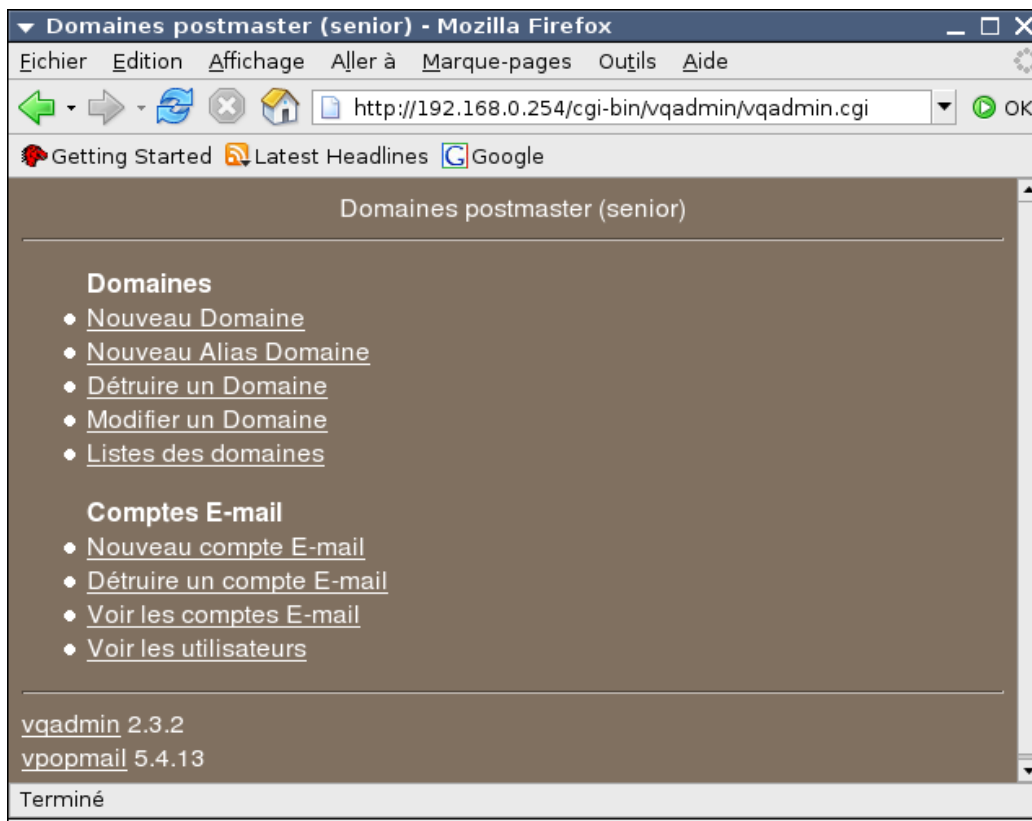
```
#
# Access List Definitions
# vol@inter7.com
#
#
# Default group contains permissions for all users
# not listed under any groups
#
# If the default group is not defined, users not
# listed under any other groups will have no
# permissions.
#
# Examples follow...
#
default - ...
#
# Access permissions:
#
# V View user information
# I View domain information
# M Modify user information
# U Modify domain information
# C Create user
# A Create domain
# D Delete user
# X Delete domain
#
# These features will still appear in the HTML templates
# if the user doesn't have access to them, however, they will
# get a permission denied error if they try to make use of
# them.
#
tech VI techluser
admin VIMUDCA adminluser
#
# An asterisk in the features field specifies that you
# want all users in this group to have access to
# all features.
#
senior * postmaster
```

Je vous laisse réfléchir, en vous aidant de la doc de vqadmin, pour exploiter au mieux, en fonction de vos besoins, les possibilités de privilèges accordées aux divers gestionnaires de vos domaines. Dans notre cas, "postmaster" est "senior", il a absolument tous les privilèges.

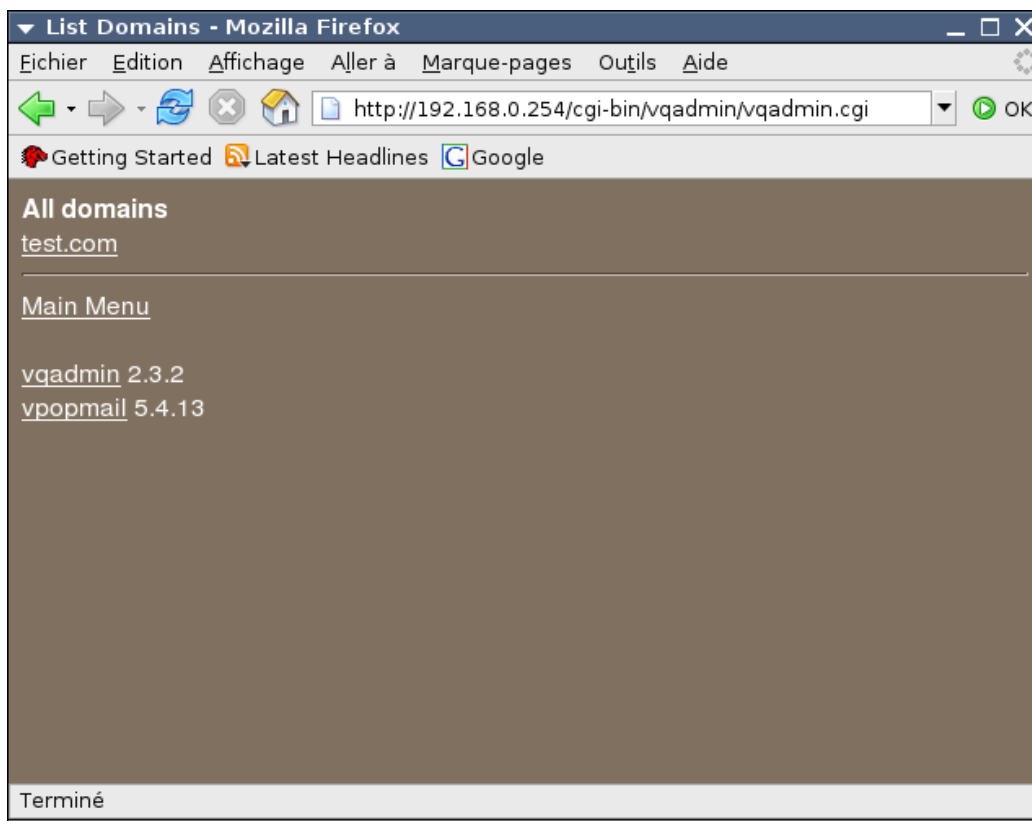
Osez installer qmail

© Christian CALECA
<http://christian.caleca.free.fr/qmail/>

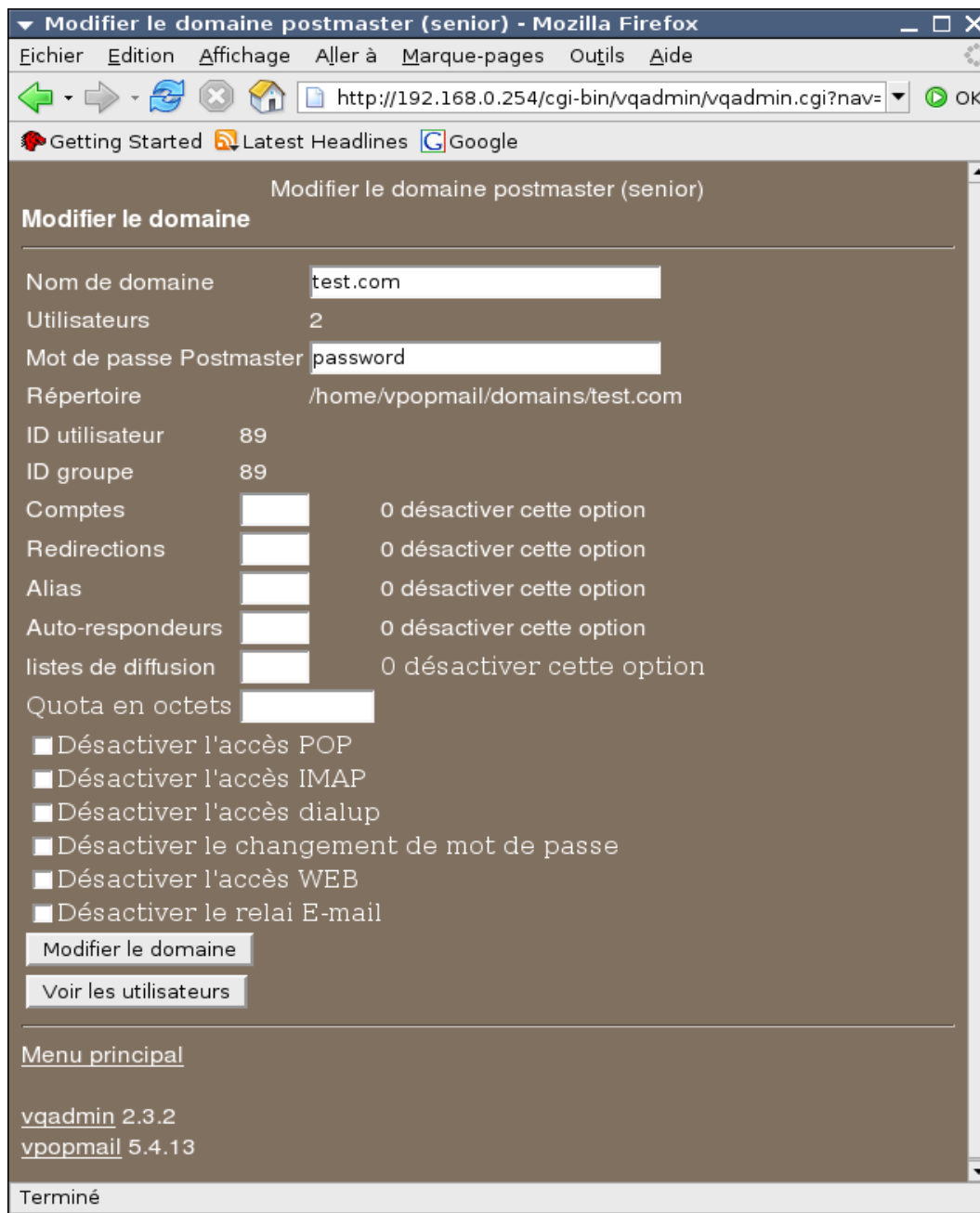
Il vous suffit maintenant d'invoquer le cgi depuis un navigateur :



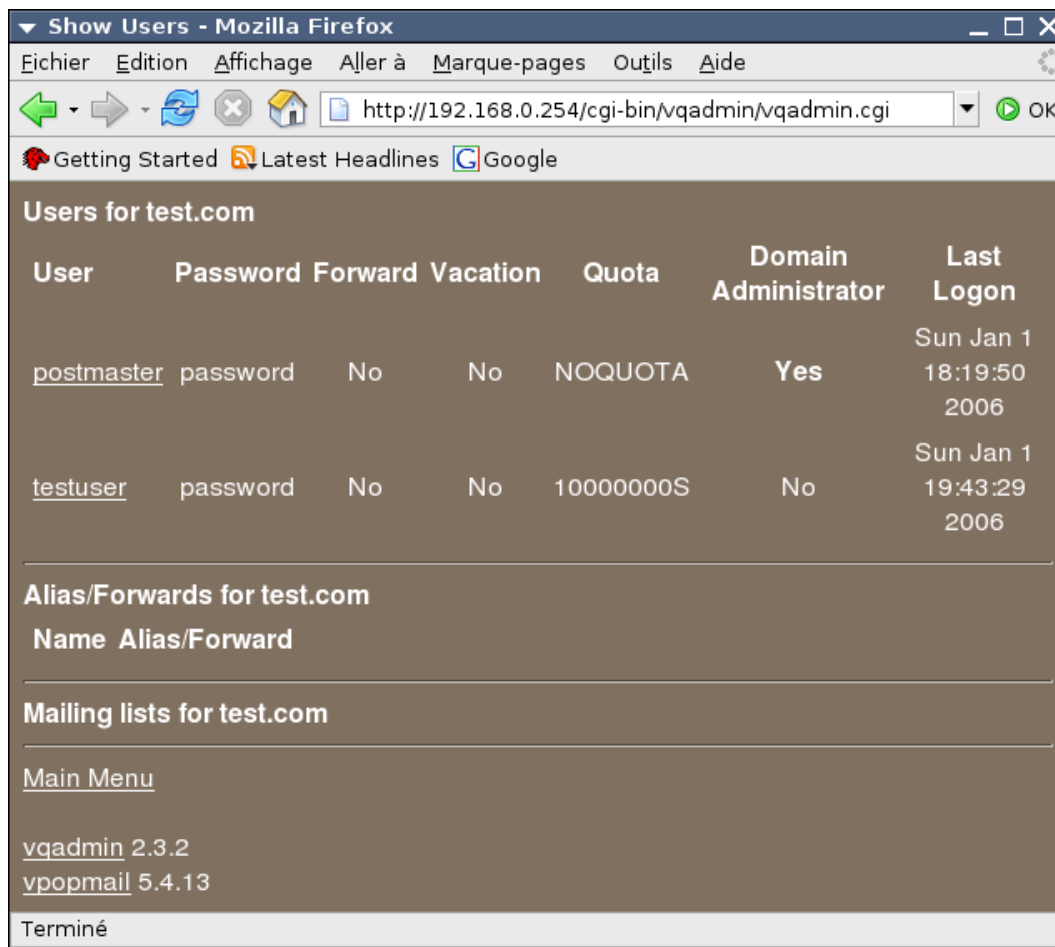
Nous pouvons lister tous les domaines existants (et en créer de nouveaux, si nécessaire) :



Modifier les caractéristiques d'un domaine :



Voir et modifier les caractéristiques des comptes dans un domaine donné :



Et d'autres choses encore. En réalité, à peu près tout ce qu'il est possible de faire avec les utilitaires Vpopmail est faisable depuis cette interface. Probablement qu'en production, vous préférerez éviter, ou tout au moins passer par https et restreindre l'accès à quelques adresses IP particulières. Tout ceci est du domaine d' Apache et sort du cadre de cet exposé.

Un peu de mécanique.

Dans chaque domaine virtuel créé, nous trouvons un fichier intitulé `.qmail-default`. Ce fichier contient normalement ceci :

```
janus# cat .qmail-default
```

```
| /home/vpopmail/bin/vdelivermail '' bounce-no-mailbox
```

Ce fichier est bien entendu fondamental. Lorsqu'un message arrive à destination du domaine en question, Qmail transmet à vdelivermail, qui placera le message dans la boîte du destinataire si elle existe (ou vers ce qui sera indiqué dans un fichier `.qmail-<user>` si il existe), ou retournera le message à l'expéditeur si le destinataire n'existe pas.

Il est par exemple possible de modifier ce fichier comme suit :

```
| /home/vpopmail/bin/vdelivermail '' delete
```

Dans ce cas, un message sans destinataire sera simplement détruit et non renvoyé à l'émetteur. Ce

n'est pas très correct. Malheureusement, avec la floraison de spams qui ciblent des destinataires forgés au hasard, et postés depuis un émetteur dont l'adresse est elle même falsifiée, ça évite une série de "bounces" inutiles.

Voyez <http://www.inter7.com/vpopmail/doc/vdelivermail.html> pour plus de détails sur vdelivermail.

Nous aurons l'occasion de revenir sur ces fichiers de type .qmail-xxx à propos des listes de diffusion et aussi de la gestion des spams.

Essais divers

Le tour du propriétaire

Voir toutes les subtilités de Qmail serait long, fastidieux et peut-être même inutile, du moins dans un premier temps. Voyons tout de même quelques points importants, qui aideront à une meilleure compréhension de notre installation, avant de passer aux autres accessoires, comme ezmlm, courier-imap etc.

Utilisation purement locale

Notre messagerie est opérationnelle. Vérifions ceci en envoyant un message au postmaster de maison.mrs :

```
janus# mail postmaster@maison.mrs
Subject: test qmail-vpopmail
Ca marche !
.
```

Et allons voir dans /home/vpopmail/domains/maison.mrs/postmaster/Maildir/new :

```
janus# ls /home/vpopmail/domains/maison.mrs/postmaster/Maildir/new/
1140885428.12364.janus,S=329

cat /home/vpopmail/domains/maison.mrs/postmaster/Maildir/new/1140885428.12364.janus\S=329
Return-Path: <root@janus.maison.mrs>
Delivered-To: postmaster@maison.mrs
Received: (qmail 12362 invoked by uid 0); 25 Feb 2006 16:37:08 -0000
Date: 25 Feb 2006 16:37:08 -0000
Message-ID: <20060225163708.12361.qmail@janus.maison.mrs>
From: root@janus.maison.mrs
To: postmaster@maison.mrs
Subject: test qmail-vpopmail

Ca marche !
```

Et voilà le travail. Bien sûr, le message est parti de janus pour arriver à janus. Pour que les messages puissent arriver de l'extérieur, il faudra que janus soit enregistré comme MX sur les DNS qui gèrent le domaine maison.mrs.

Ne pas oublier les alias

Il est très fortement conseillé de prévoir que "root", "postmaster" et "mailer-daemon" puissent recevoir localement des messages. Beaucoup d'applications utilisent la messagerie pour notifier l'administrateur de tel ou tel événement, et ces trois comptes de messagerie doivent être fonctionnels, pour éviter d'avoir des messages qui se perdent ("bounce").

Il est d'usage, pour résoudre ce problème, d'utiliser un unique compte, appartenant à un utilisateur référencé sur la machine, qui recevra les messages à destination de ces trois (parfois plus) utilisateurs particuliers. Le principe est de rediriger plusieurs adresses sur une seule. C'est le rôle du mécanisme des alias de réaliser cette opération.

Pour éviter de vilains "bounce", nous dirons que c'est "prof@maison.mrs" qui sera l'administrateur de tout ce système, que c'est donc lui qui devra recevoir les messages adressés à "root", "postmaster" et "mailer-daemon". Nous allons créer quelques alias en conséquence.

Les alias se définissent en créant dans /var/qmail/alias des fichiers dont le nom est de la forme :

```
.qmail-<nom du compte à rediriger>
```

et qui contienne le nom du compte réel. Il suffira par exemple de faire ceci :

```
janus# echo prof@maison.mrs > /var/qmail/alias/.qmail-root
janus# echo prof@maison.mrs > /var/qmail/alias/.qmail-mailer-daemon
janus# echo prof@maison.mrs > /var/qmail/alias/.qmail-postmaster
```

Enfin, attribuons des droits corrects à ces fichiers :

```
janus# chmod 644 /var/qmail/alias/.qmail-root /var/qmail/alias/.qmail-postmaster \
/var/qmail/alias/.qmail-mailer-daemon
```

Maintenant, tout message envoyé à l'un de ces trois personnages se retrouvera dans la boîte aux lettres de prof. Démonstration :

```
janus# mail root
Subject: test des alias
ça marche !
.
Cc: mailer-daemon postmaster
```

Voyons dans la boîte de "prof" :

```
janus# cd /home/vpopmail/domains/maison.mrs/prof/Maildir/new/
janus# ls
1140969372.6724.janus,S=490 1140969372.6726.janus,S=484 1140969372.6728.janus,S=493
```

3 nouveaux messages :

```
janus# cat 1140969372.6724.janus\S\=490

Return-Path: <root@janus.maison.mrs>
Delivered-To: prof@maison.mrs
Received: (qmail 6721 invoked by alias); 26 Feb 2006 15:56:12 -0000
Delivered-To: postmaster@janus.maison.mrs
Received: (qmail 6713 invoked by uid 0); 26 Feb 2006 15:56:12 -0000
Date: 26 Feb 2006 15:56:12 -0000
Message-ID: <20060226155612.6712.qmail@janus.maison.mrs>
From: root@janus.maison.mrs
To: root@janus.maison.mrs
Subject: test des alias
Cc: mailer-daemon@janus.maison.mrs, postmaster@janus.maison.mrs

ça marche !

cat 1140969372.6726.janus,S=484

Return-Path: <root@janus.maison.mrs>
Delivered-To: prof@maison.mrs
Received: (qmail 6722 invoked by alias); 26 Feb 2006 15:56:12 -0000
Delivered-To: root@janus.maison.mrs
Received: (qmail 6713 invoked by uid 0); 26 Feb 2006 15:56:12 -0000
Date: 26 Feb 2006 15:56:12 -0000
Message-ID: <20060226155612.6712.qmail@janus.maison.mrs>
From: root@janus.maison.mrs
To: root@janus.maison.mrs
Subject: test des alias
Cc: mailer-daemon@janus.maison.mrs, postmaster@janus.maison.mrs

ça marche !

cat 1140969372.6728.janus,S=493

Return-Path: <root@janus.maison.mrs>
Delivered-To: prof@maison.mrs
Received: (qmail 6720 invoked by alias); 26 Feb 2006 15:56:12 -0000
Delivered-To: mailer-daemon@janus.maison.mrs
Received: (qmail 6713 invoked by uid 0); 26 Feb 2006 15:56:12 -0000
Date: 26 Feb 2006 15:56:12 -0000
Message-ID: <20060226155612.6712.qmail@janus.maison.mrs>
From: root@janus.maison.mrs
To: root@janus.maison.mrs
Subject: test des alias
Cc: mailer-daemon@janus.maison.mrs, postmaster@janus.maison.mrs
```

ça marche !

Nous le retrouvons trois fois. Normal, à cause des "Cc:".

Remarque à propos des alias

Ce qui vient d'être fait peut paraître ambigu. En effet, nous avons manipulé des alias "système". Entendez par là que :

- root est implicitement root@janus.maison.mrs
- mailer-daemon est mailer-daemon@janus.maison.mrs
- etc.

Nous avons redirigé ces comptes sur des comptes du domaine "virtuel" maison.mrs. Nous aurions pu créer tout aussi bien un domaine virtuel "test.com" et rediriger les comptes système @janus.maison.mrs vers des comptes @test.com, ça aurait fonctionné de la même manière.

Il ne faut pas confondre avec les alias (ou redirections) que l'on pourra créer, par exemple avec qmailadmin, ou avec la commande valias de Vpopmail. Ces alias se présentent eux aussi sous la forme d'un fichier .qmail-<alias>, mais ceux-ci seront situés dans /home/vpopmail/domains/<domaine_concerné>

Et les logs ?

Voyons ce que donnent les logs à la mode multilog.

Envoi d'un message quelconque à "prof", puis lecture de /var/log/qmail/send/current :

```
janus# cd /var/log/qmail
janus# cat current
@400000004401b6a637424f14 new msg 4198711
@400000004401b6a637426e54 info msg 4198711: bytes 341 from <root@janus.maison.mrs> qp 6619 uid 1005
@400000004401b6a63794cc44 starting delivery 2: msg 4198711 to local maison.mrs-prof@maison.mrs
@400000004401b6a63794ef6c status: local 2/10 remote 0/20
@400000004401b6a63794fb24 delivery 1: success: did_0+1+0/qp_6619/
@400000004401b6a637c9d664 status: local 1/10 remote 0/20
@400000004401b6a637c9f1bc end msg 4198710
@400000004401b6a63aaafa0d4 delivery 2: success: did_0+0+1/
@400000004401b6a63aaafc3fc status: local 0/10 remote 0/20
@400000004401b6a63aaafd39c end msg 4198711
@400000004401cfa6053ae3dc new msg 4198710
@400000004401cfa6053b031c info msg 4198710: bytes 313 from <root@janus.maison.mrs> qp 6713 uid 0
@400000004401cfa605c2c11c starting delivery 3: msg 4198710 to local mailer-daemon@janus.maison.mrs
@400000004401cfa605c5709c status: local 1/10 remote 0/20
@400000004401cfa605d6cdec starting delivery 4: msg 4198710 to local postmaster@janus.maison.mrs
@400000004401cfa605d965fc status: local 2/10 remote 0/20
@400000004401cfa605e9a624 starting delivery 5: msg 4198710 to local root@janus.maison.mrs
@400000004401cfa605ec421c status: local 3/10 remote 0/20
@400000004401cfa607d07724 new msg 4198711
@400000004401cfa607d09a4c info msg 4198711: bytes 423 from <root@janus.maison.mrs> qp 6721 uid 1005
@400000004401cfa608a9ef04 starting delivery 6: msg 4198711 to local maison.mrs-prof@maison.mrs
@400000004401cfa608aa122c status: local 4/10 remote 0/20
@400000004401cfa608aa21cc delivery 4: success: did_0+1+0/qp_6721/
@400000004401cfa608aa316c status: local 3/10 remote 0/20
@400000004401cfa608aa410c delivery 5: success: did_0+1+0/qp_6722/
@400000004401cfa608aa4cc4 status: local 2/10 remote 0/20
@400000004401cfa608aa5c64 delivery 3: success: did_0+1+0/qp_6720/
@400000004401cfa608aa6c04 status: local 1/10 remote 0/20
@400000004401cfa608aad194 end msg 4198710
@400000004401cfa608c06504 new msg 4198712
@400000004401cfa608c0788c info msg 4198712: bytes 417 from <root@janus.maison.mrs> qp 6722 uid 1005
@400000004401cfa60928fa74 starting delivery 7: msg 4198712 to local maison.mrs-prof@maison.mrs
@400000004401cfa60929256c status: local 2/10 remote 0/20
@400000004401cfa60929350c new msg 4198713
```

```
@400000004401cfa6092940c4 info msg 4198713: bytes 426 from <root@janus.maison.mrs> qp 6720 uid 1005
@400000004401cfa6097a9514 starting delivery 8: msg 4198713 to local maison.mrs-prof@maison.mrs
@400000004401cfa6097ab83c status: local 3/10 remote 0/20
@400000004401cfa60baab79c delivery 6: success: did_0+0+1/
@400000004401cfa60bb4b634 status: local 2/10 remote 0/20
@400000004401cfa60bba673c end msg 4198711
@400000004401cfa60bc7675c delivery 7: success: did_0+0+1/
@400000004401cfa60bcea2ec status: local 1/10 remote 0/20
@400000004401cfa60bd5617c end msg 4198712
@400000004401cfa60c25851c delivery 8: success: did_0+0+1/
@400000004401cfa60c2d8fb4 status: local 0/10 remote 0/20
@400000004401cfa60c332d34 end msg 4198713
```

Ce sont des logs bien détaillés, mais que représentent ces choses, dans la première colonne ?

Ce sont des "timestamp" donnés dans le format TAI (Temps Atomique International). Si vous voulez absolument savoir ce que cela représente exactement, consultez la page : <http://cr.yip.to/libtai/tai64.html>.

On aurait espéré pouvoir lire la date de façon compréhensible... Heureusement tout de même qu'il y a dans les daemontools l'utilitaire tai64nlocal qui va nous aider :

```
janus# cat current | tai64nlocal
2006-02-26 16:56:12.087745500 new msg 4198710
2006-02-26 16:56:12.087753500 info msg 4198710: bytes 313 from <root@janus.maison.mrs> qp 6713 uid 0
2006-02-26 16:56:12.096649500 starting delivery 3: msg 4198710 to local mailer-daemon@janus.maison.mrs
2006-02-26 16:56:12.096825500 status: local 1/10 remote 0/20
2006-02-26 16:56:12.097963500 starting delivery 4: msg 4198710 to local postmaster@janus.maison.mrs
2006-02-26 16:56:12.098133500 status: local 2/10 remote 0/20
2006-02-26 16:56:12.099198500 starting delivery 5: msg 4198710 to local root@janus.maison.mrs
2006-02-26 16:56:12.099369500 status: local 3/10 remote 0/20
2006-02-26 16:56:12.131102500 new msg 4198711
2006-02-26 16:56:12.131111500 info msg 4198711: bytes 423 from <root@janus.maison.mrs> qp 6721 uid 100
5
2006-02-26 16:56:12.145354500 starting delivery 6: msg 4198711 to local maison.mrs-prof@maison.mrs
2006-02-26 16:56:12.145363500 status: local 4/10 remote 0/20
2006-02-26 16:56:12.145367500 delivery 4: success: did_0+1+0/qp_6721/
2006-02-26 16:56:12.145371500 status: local 3/10 remote 0/20
2006-02-26 16:56:12.145375500 delivery 5: success: did_0+1+0/qp_6722/
2006-02-26 16:56:12.145378500 status: local 2/10 remote 0/20
2006-02-26 16:56:12.145382500 delivery 3: success: did_0+1+0/qp_6720/
2006-02-26 16:56:12.145386500 status: local 1/10 remote 0/20
2006-02-26 16:56:12.145412500 end msg 4198710
2006-02-26 16:56:12.146826500 new msg 4198712
2006-02-26 16:56:12.146831500 info msg 4198712: bytes 417 from <root@janus.maison.mrs> qp 6722 uid 100
5
2006-02-26 16:56:12.153680500 starting delivery 7: msg 4198712 to local maison.mrs-prof@maison.mrs
2006-02-26 16:56:12.153691500 status: local 2/10 remote 0/20
2006-02-26 16:56:12.153695500 new msg 4198713
2006-02-26 16:56:12.153698500 info msg 4198713: bytes 426 from <root@janus.maison.mrs> qp 6720 uid 100
5
2006-02-26 16:56:12.159028500 starting delivery 8: msg 4198713 to local maison.mrs-prof@maison.mrs
2006-02-26 16:56:12.159037500 status: local 3/10 remote 0/20
2006-02-26 16:56:12.195737500 delivery 6: success: did_0+0+1/
2006-02-26 16:56:12.196392500 status: local 2/10 remote 0/20
2006-02-26 16:56:12.196765500 end msg 4198711
2006-02-26 16:56:12.197617500 delivery 7: success: did_0+0+1/
2006-02-26 16:56:12.198091500 status: local 1/10 remote 0/20
2006-02-26 16:56:12.198533500 end msg 4198712
2006-02-26 16:56:12.203785500 delivery 8: success: did_0+0+1/
2006-02-26 16:56:12.204312500 status: local 0/10 remote 0/20
2006-02-26 16:56:12.204680500 end msg 4198713
```

Bien qu'au format US, c'est tout de même bien plus lisible. Notez la précision mais ne riez pas, Qmail travaille très vite et une grande précision dans le "timestamp" peut s'avérer utile pour le dépouillement des logs.

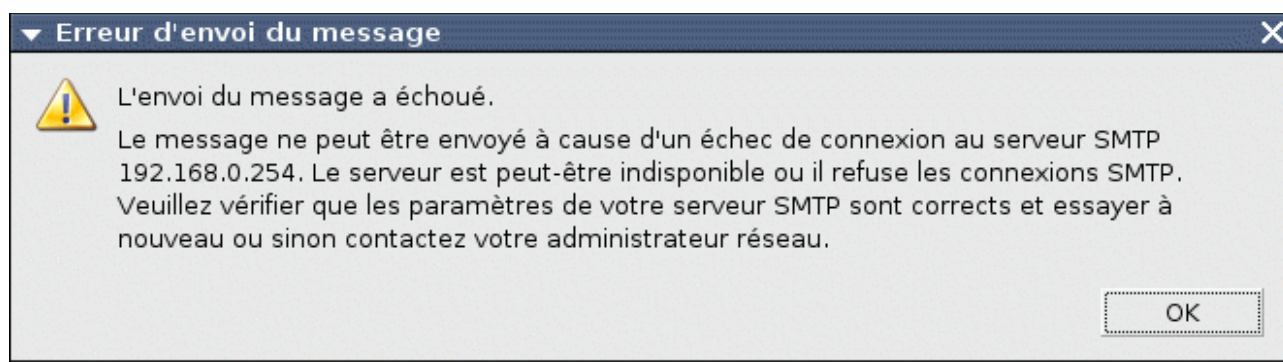
Tester qmail-smtpd

Pour ceci, il faut s'adresser au serveur par le biais, par exemple, d'un client de messagerie (MUA) traditionnel, un Thunderbird, par exemple.

Nous n'avons pas ceci sur janus (c'est une machine sans interface graphique), mais nous avons cela sur d'autres machines du réseau local. Il suffit, sur l'une d'entre elles, de configurer notre oiseau du tonnerre pour qu'il envoie les messages par janus.maison.mrs.

Premier test : envoi d'un message à prof@maison.mrs.

Résultat :



Indisponible ? Non, nous avons vu avec netstat que le port smtp est à l'écoute.

Il refuse les connexions ? Oui. N'oublions pas que qmail-smtpd passe par tcpserver, qui lui-même n'accorde le passage qu'en fonction de ce qui est écrit dans /etc/tcp.smtp. Hors :

```
janus# cat /home/vpopmail/etc/tcp.smtp
127.0.0.1:allow,RELAYCLIENT=""
:deny
```

En toute logique, il n'y a que localhost qui passe. Il faut donc modifier les règles de la façon suivante :

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.:allow
:deny
```

La seconde ligne veut dire que n'importe qui du réseau local (192.168.0.0/24 dans l'exemple) peut entrer sur le serveur SMTP, mais sans le flag "RELAYCLIENT". Les hôtes du réseau local ne peuvent donc pas faire relayer leurs messages vers l'extérieur.

N'oublions pas qu'il faut mettre à jour le fichier /home/vpopmail/etc/tcp.smtp.cdb. Comme nous avons installé le script qmailctl, autant l'utiliser :

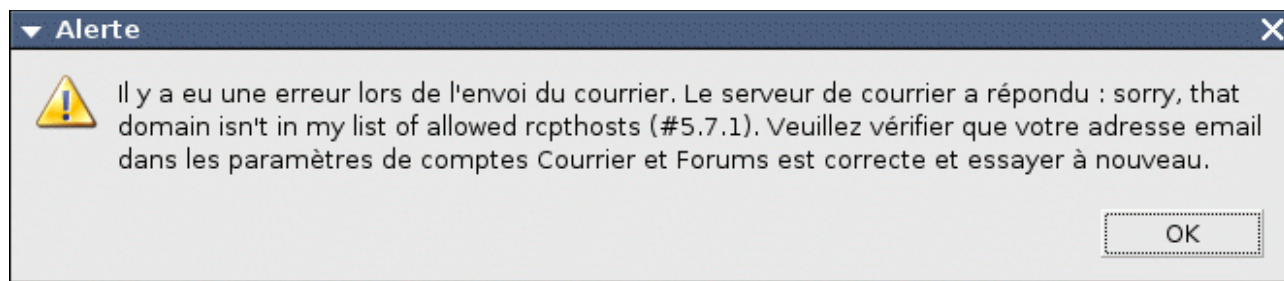
```
janus# qmailctl cdb
```

Second test : renvoi du même message. Il passe et on le retrouve bien dans la boîte du destinataire :

```
Return-Path: <postmaster@grenouille.com>
Delivered-To: prof@maison.mrs
Received: (qmail 4656 invoked from network); 30 Dec 2005 15:35:09 -0000
Received: from unknown (HELO ?192.168.0.15?) (192.168.0.15)
by janus.maison.mrs with SMTP; 30 Dec 2005 15:35:09 -0000
Message-ID: <43B55389.6090601@grenouille.com>
Date: Fri, 30 Dec 2005 16:34:33 +0100
From: Christian Caleca <postmaster@maison.mrs>
User-Agent: Mozilla Thunderbird 1.0.7 (X11/20051017)
X-Accept-Language: fr, en
MIME-Version: 1.0
To: prof@maison.mrs
Subject: test qmail local
Content-Type: text/plain; charset=ISO-8859-1; format=flowed
Content-Transfer-Encoding: 8bit
```


ça passe ?

Oui, mais sommes-nous bien certains que les hôtes du réseau local ne peuvent pas faire relayer leurs messages vers l'extérieur ? Oui, parce que le drapeau RELAYCLIENT n'est pas mis ! Pour nous en convaincre, essayons d'envoyer un message, mais cette fois-ci à l'extérieur, par exemple à prof@grenouille.com :



Ici, le message a été accepté en entrée par ucspi-tcp, mais Qmail n'a pas accepté le message car il ne satisfait pas l'une ou l'autre des conditions suivantes :

- il n'y a pas le "flag" de relais inconditionnel "RELAYCLIENT",
- le destinataire n'est pas dans un domaine référencé dans /etc/qmail/rcpthosts.

Pour l'instant, ce fichier ne contient que deux entrées :

```
janus# cat /var/qmail/control/rcpthosts
janus.maison.mrs
maison.mrs
```

Ces entrées ont été mises en place lorsque nous avons utilisé le script "config-fast", au moment de l'installation de Qmail (janus.maison.mrs), puis lorsque nous avons créé notre domaine virtuel (vaddomain maison.mrs).

Si nous ajoutons l'entrée "grenouille.com" à ce fichier, alors le message passera. Mais un autre message à destination d'un autre domaine ne passera toujours pas. Nous voudrions que notre SMTP relaye tous les messages sortants, mais seulement pour les hôtes de notre réseau local.

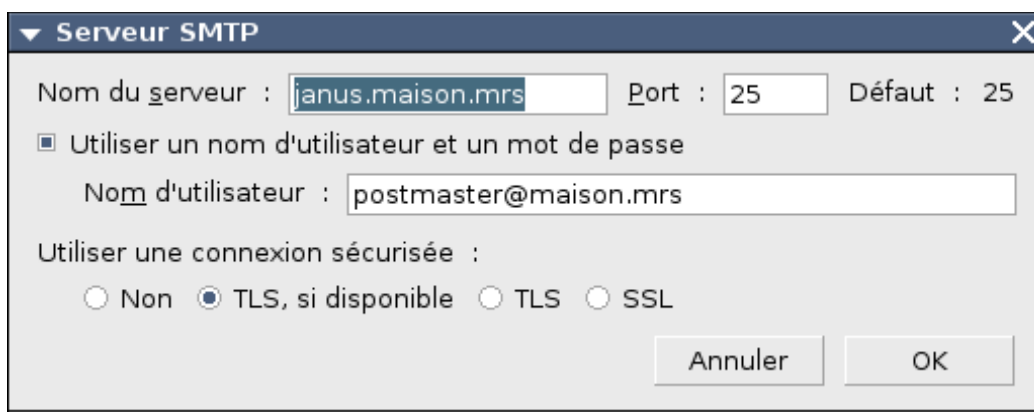
Mais avant de modifier notre tcp.smtp, expérimentons les bienfaits du "patch" qui permet d'implémenter TLS/SSL dans Qmail (qmail-toaster-0.8.1.patch). Ce patch implémente la fonctionnalité "starttls"⁵. Cette fonctionnalité permet l'identification et le chiffrement des données :

- de serveur à serveur,
- de client à serveur.

Lorsqu'un client arrive à réaliser une authentification avec un serveur SMTP, il est alors considéré comme étant local (au sens localhost). Ici, nous allons donc pouvoir tester cette fonctionnalité.

Les clients (MUA) modernes savent faire. Par exemple, avec thunderbird :

⁵ <http://www.ietf.org/rfc/rfc2487.txt>



Et réessayons d'envoyer le message. Dans les logs, nous voyons :

```
2006-02-26 18:39:59.549659500 CHKUSER relaying rcpt: from
<postmaster@maison.mrs:postmaster@maison.mrs:>
remote <[192.168.0.15]:unknown:192.168.0.15>
rcpt <prof@grenouille.com> : client allowed to relay
```

Récupérons ce message :

```
Received: (qmail 4709 invoked by uid 7797); 26 Feb 2006 17:35:14 -0000
Received: by simscan 1.1.0 ppid: 4684, pid: 4706, t: 0.1856s
scanners: clamav: 0.87.1/m:34/d:1162 spam: 3.1.0
X-Spam-Checker-Version: SpamAssassin 3.1.0 (2005-09-13) on ns2023.ovh.net
X-Spam-Level:
X-Spam-Status: No, score=-2.3 required=5.0 tests=BAYES_00 autolearn=ham
version=3.1.0
Received: from unknown (HELO janus.maison.mrs) (85.68.119.205)
by ns4223.ovh.net with (DHE-RSA-AES256-SHA encrypted) SMTP; 26 Feb 2006 17:35:13 -0000
Received: (qmail 6911 invoked by uid 89); 26 Feb 2006 17:39:59 -0000
Received: from unknown (HELO ?192.168.0.15?) (postmaster@maison.mrs@192.168.0.15)
by 0 with ESMTPA; 26 Feb 2006 17:39:59 -0000
Message-ID: <4401E6CF.6010602@maison.mrs>
Date: Sun, 26 Feb 2006 18:35:11 +0100
From: Postmaster <postmaster@maison.mrs>
User-Agent: Mozilla Thunderbird 1.0.7 (X11/20051013)
X-Accept-Language: fr, en
MIME-Version: 1.0
To: prof@grenouille.com
Subject: test starttls
Content-Type: text/plain; charset=ISO-8859-1; format=flowed
Content-Transfer-Encoding: 8bit
```

ça marche !

Non seulement le message est passé, mais il est passé chiffré, puisque le Qmail destinataire sait aussi utiliser TLS/SSL.

Cette méthode, très élégante, remplace avantageusement le système "pop before smtp" pour résoudre le problème des utilisateurs ne disposant pas d'une adresse IP fixe, mais qui ont besoin d'un SMTP pour relayer leurs messages sortants.

Pour en revenir à notre réseau local, dont les messages vers l'extérieur devraient pouvoir être relayés, sans forcément nécessiter une authentification TLS, il faut agir sur /home/vpopmail/etc/tcp.smtp comme suit :

```
janus# cat /home/vpopmail/etc/tcp.smtp
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.:allow,RELAYCLIENT=""
:allow
```

Et effectuer la mise à jour du cdb.

Il ne serait pas mauvais ici de tester nos serveurs POP3 et POP3s. Je vous laisse faire par vous-même.

Nous savons que les serveurs sont bien présents :

```
janus# ps -aef | grep pop3
root      4420  4413  0 10:08 ?        00:00:00 supervise qmail-pop3d
qmail    4422  4421  0 10:08 ?        00:00:00 multilog t /var/log/qmail/pop3d
root      4423  4413  0 10:08 ?        00:00:00 supervise qmail-pop3ds
qmail    4425  4424  0 10:08 ?        00:00:00 multilog t /var/log/qmail/pop3ds
vpopmail 4431  4420  0 10:09 ?        00:00:00 /usr/local/bin/tcpserver -l 0 -R -H -v -u89 -g89 0 110
          /var/qmail/bin/qmail-popup janus
          /home/vpopmail/bin/vchkpw
          /var/qmail/bin/qmail-pop3d Maildir
vpopmail 4432  4423  0 10:09 ?        00:00:00 /usr/local/bin/tcpserver -l 0 -R -H -v -u89 -g89 0 995
          /usr/sbin/stunnel -f -p
          /var/qmail/control/servercert.pem -l
          /var/qmail/bin/qmail-popup -- qmail-popup janus
          /home/vpopmail/bin/vchkpw
          /var/qmail/bin/qmail-pop3d Maildir
```

Et aussi :

```
janus# netstat -a | grep pop3
tcp        0      0 *:pop3s                :::*          LISTEN
tcp        0      0 *:pop3                  :::*          LISTEN
```

Le serveur IMAP : courier-imap

Passons maintenant au serveur IMAP. Il existe plusieurs serveurs qui permettent de gérer les boîtes aux lettres au format Maildir. Courier-IMAP semble cependant le plus approprié au couple qmail/vpopmail.

Courier-imap existe dans les distributions Debian, nous faisons un apt-get install courier-imap, qui installera aussi courier-authdaemon et courier-base.

Malheureusement, en lisant le fichier /usr/share/doc/courier-base/README.Debian :

```
janus# cat /usr/share/doc/courier-base/README.Debian
...
Vpopmail
*****

There is no support for vpopmail in the Debian packages, because
this would cause Courier to be removed from the main distribution.
See Debian bugs #182040 and #248319 for more information.
...
```

Vpopmail dépend intimement de Qmail, lequel ne peut être distribué de façon binaire sans l'accord de l'auteur. Vpopmail, bien qu'existant dans la "sid" n'est pas placé dans "main" mais dans "contrib". Si les paquetages courier supportent Vpopmail, il faudra les placer eux aussi dans "contrib" et non plus dans "main". Courier fonctionne avec tout MTA et les responsables Debian ne veulent pas, à cause de Vpopmail, sortir les paquetages courier de "main".

Bref, nous avons un courier-imap installé, mais qui ne fonctionnera pas avec notre configuration. Nous avons donc deux solutions possibles :

- installer une fois de plus à partir des sources,
- recompiler un paquetage binaire debian à partir du paquetage source, et l'installer ensuite avec dpkg.

Nous allons choisir la seconde solution. Il faudra prendre quelques précautions, les paquetages binaires ne se laissant pas construire sans quelques difficultés. En effet, sur la Debian Etch, certaines étapes de la compilation ne devront pas se faire en étant root et d'autres oui.

Installation du paquetage source "courier"

Dans la suite, c'est un utilisateur "normal" (chris) qui va effectuer l'opération :

```
janus:/home/chris$ mkdir courier
janus:/home/chris$ cd courier
janus:/home/chris/courier$ apt-get source courier
```

Normalement, l'installation a créé un répertoire "courier-0.47". Dans ce répertoire, il y a un répertoire "debian" et dedans, un fichier "rules".

Nous allons faire vite et sale. Editez ce fichier et recherchez dedans le paragraphe :

```
COMMON_CONF_OPTS=--prefix=/usr --mandir=\${prefix}/share/man \
--with-piddir=/var/run/courier \
--sysconfdir=/etc/courier \
--libexecdir=\${prefix}/lib/courier \
--datadir=\${prefix}/lib/courier \
--localstatedir=/var/lib/courier \
--sbindir=\${exec_prefix}/sbin \
--with-mailuser=daemon \
--with-mailgroup=daemon \
--without-socks \
--enable-workarounds-for-imap-client-bugs \
```

```

--with-authpam \
--without-authpwd \
--with-authmysql \
--with-mysql-includes=/usr/include/mysql \
--with-mysql-libs=/usr/lib \
--with-authpgsql \
--with-pgsql-includes=`pg_config --includedir` \
--with-pgsql-libs=/usr/lib \
--without-authshadow \
--without-authvchkpw \
--with-authdaemonvar=/var/run/courier/authdaemon \
--with-authldap \
--with-authcram \
--with-db=gdbm \
--without-fcgi \
--with-htlplibdir=/usr/share/sqwebmail \
--with-ispell=/usr/bin/ispell \
--enable-imageurl=/sqwebmail \
--with-mailer="/usr/sbin/sendmail" \
--enable-sendmail="/usr/sbin/sendmail" \
--with-cachedir=/var/cache/sqwebmail \
--with-calendardir=/var/run/courier/calendar \
--with-webadmindir=/usr/share/courier/webadmin \
--enable-userdb \
--enable-syslog=1 \
--enable-unicode \
--disable-root-check

```

Nous allons juste modifier la ligne surlignée en :

```
--with-authvchkpw \
```

Commençons par voir les dépendances nécessaires :

```

janus# cd /home/chris/courier/courier-0.47
janus# dpkg-checkbuilddeps
dpkg-checkbuilddeps: Unmet build dependencies: libmysqlclient-dev | libmysqlclient14-dev
| libpam0g-dev libgdbm-dev
| libgdbmgl-dev libperl-dev
| zlib1g-dev libldap-dev
| libsasl2-dev libsasl-dev expect
| mgetty-fax libfam-dev libssl-dev
| libpq-dev postgresql-dev
| automake1.8

dpkg-checkbuilddeps: Build conflicts: automake1.4

```

Il faut bien entendu installer tout ce qui est nécessaire pour ne plus avoir de messages générés par "dpkg-checkbuilddeps".

Puis, nous reconstruisons les paquetages binaires. C'est ici qu'il faut ruser, en utilisant les services de "fakeroot" :

```
janus# dpkg-buildpackage -rfakeroot -us -uc
```

L'opération est un peu longue, tous les paquetages courier-xxx vont être construits.

A la fin de l'opération, nous disposons de tous les paquetages courier recompilés, avec le support de Vpopmail.

```

janus# cd /home/chris/courier
janus# ls *.deb
courier-authdaemon_0.47-13_i386.deb      courier-mlm_0.47-13_i386.deb
courier-authmysql_0.47-13_i386.deb      courier-mta_0.47-13_i386.deb
courier-authpostgresql_0.47-13_i386.deb  courier-mta-ssl_0.47-13_i386.deb
courier-base_0.47-13_i386.deb           courier-pcp_0.47-13_i386.deb
courier-doc_0.47-13_all.deb             courier-pop_0.47-13_i386.deb
courier-faxmail_0.47-13_i386.deb        courier-pop-ssl_0.47-13_i386.deb
courier-imap_3.0.8-13_i386.deb          courier-ssl_0.47-13_i386.deb
courier-imap-ssl_3.0.8-13_i386.deb      courier-webadmin_0.47-13_i386.deb
courier-ldap_0.47-13_i386.deb          sqwebmail_0.47-13_i386.deb
courier-maildrop_0.47-13_i386.deb

```

En réalité, il nous suffit de remplacer le paquetage courier-authdaemon de la distribution, par celui que nous avons recompilé.

Pour réaliser cette opération, nous utilisons dpkg :

```
janus# dpkg -i ./courier-authdaemon_0.47-13_i386.deb
(Lecture de la base de données... 72898 fichiers et répertoires déjà installés.)
Préparation du remplacement de courier-authdaemon 0.47-13 (en utilisant .../courier-authdaemon_0.47-13_i386.deb) ...
Stopping Courier authdaemon: done.
Dépaquetage de la mise à jour de courier-authdaemon ...
Paramétrage de courier-authdaemon (0.47-13) ...
Starting Courier authdaemon: done.
```

Attention, cette méthode présente un risque !

A la prochaine mise à jour des paquetages, apt-get upgrade remplacera sans se poser de questions votre courier-authdaemon, et l'authentification ne fonctionnera plus...

La solution "propre" consiste à se créer un dépôt local, dans lequel nous mettrons tous les paquetages relatifs à courier, et que nous indiquerons en tête du fichier /etc/apt/sources.list.

<http://www.debianaddict.org/article31.html> explique très bien comment réaliser ce dépôt.

Configuration de courier-authdaemon

Dans le fichier /etc/courier/authdaemonrc, trouvez ces lignes, et modifiez le paramètre "authmodulelist" comme indiqué :

```
# The authentication modules that are linked into authdaemond. The
# default list is installed. You may selectively disable modules simply
# by removing them from the following list. The available modules you
# can use are: authcustom authcram authuserdb authvchkpw authldap authpgsql authmysql authpam
authmodulelist="authvchkpw"
```

Configuration de courier-imap

La configuration par défaut devrait être acceptable dans la plupart des cas.

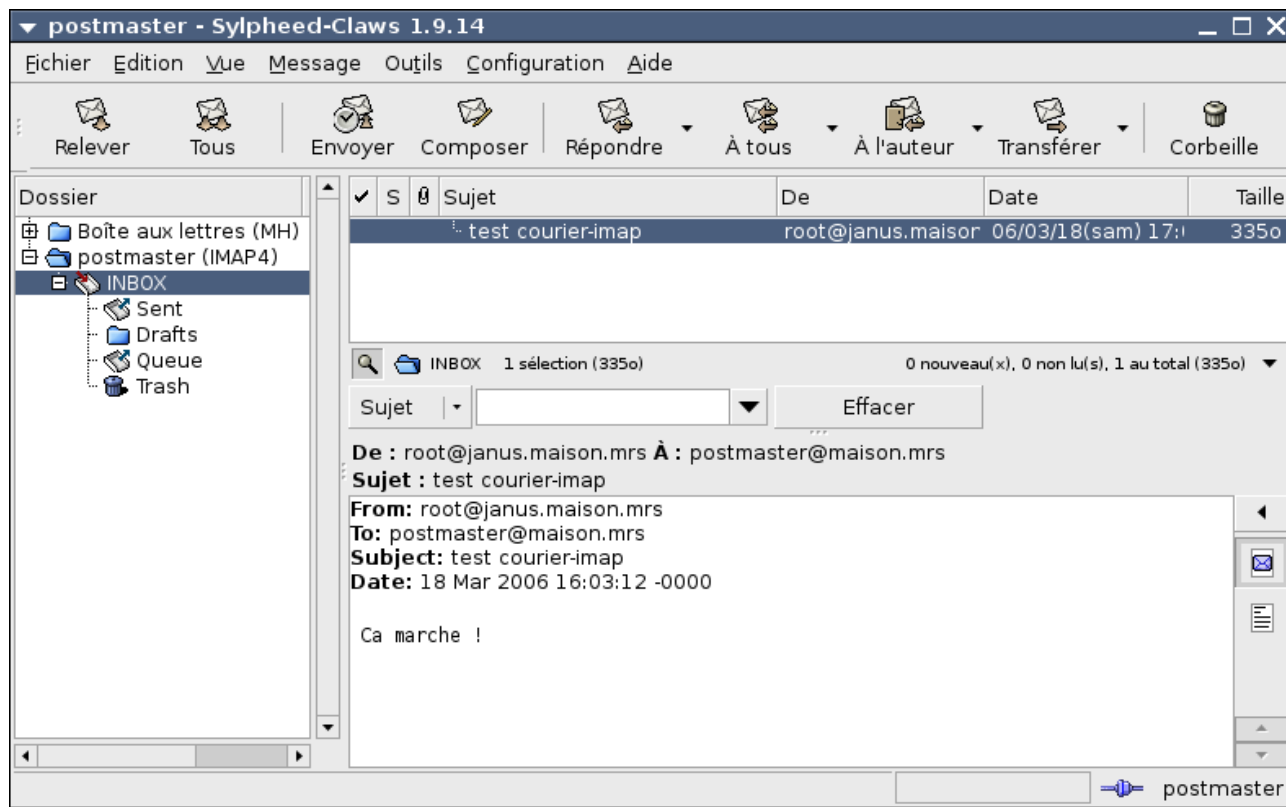
Essais

Et allons voir avec notre sylpheed-claws s'il est possible d'accéder en IMAP.

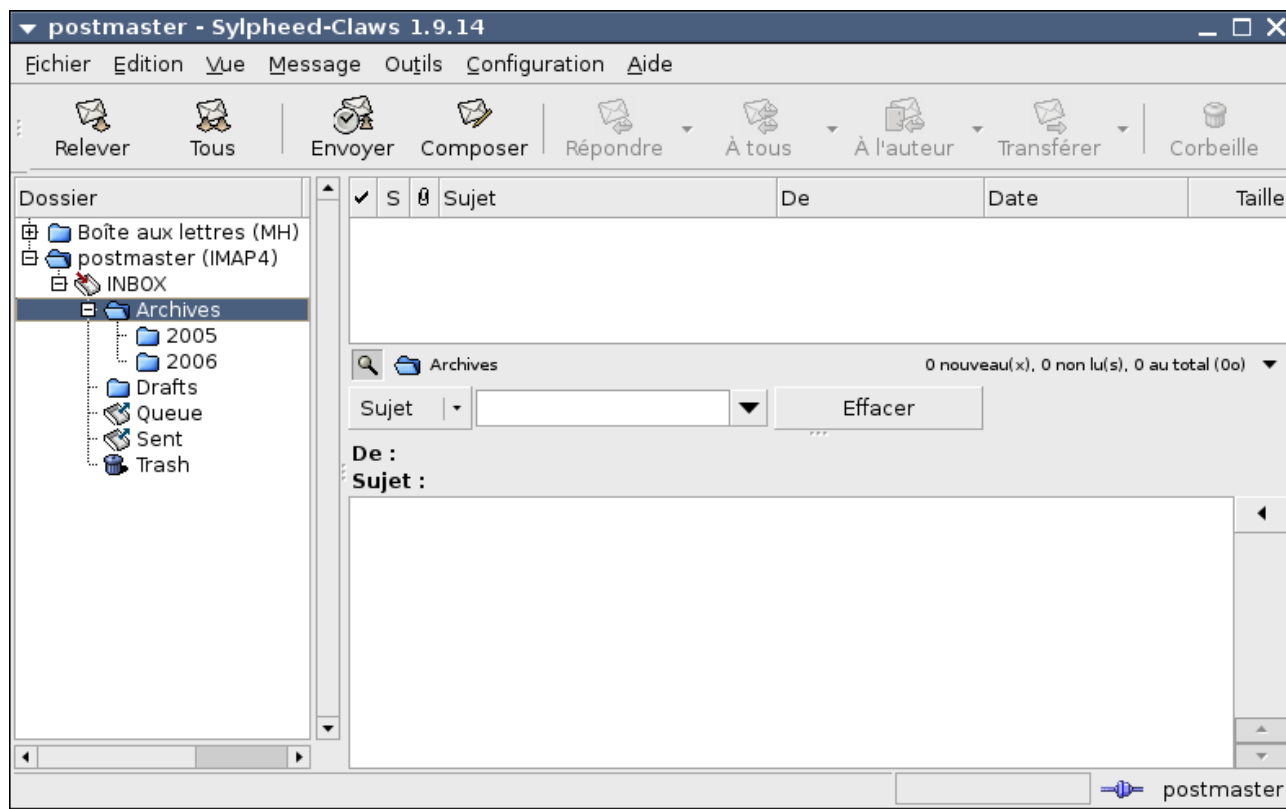
Un petit message, d'abord, pour qu'il y ait quelque chose dans la boîte :

```
janus:~# mail postmaster@maison.mrs
Subject: test courier-imap
Ca marche !
.
Cc:
```

Et dans sylpheed-claws :



Et en imap, nous pouvons créer sur les serveur des dossiers de classement de nos messages :



L'avantage du format Maildir, c'est qu'il est possible de créer plusieurs niveaux de sous-répertoires. Voyons ce que ceci a donné du côté de Vpopmail :

```
janus:/home/vpopmail/domains/maison.mrs/postmaster/Maildir# ls -l
total 12
-rw----- 1 vpopmail vchkpw 45 2006-03-02 14:58 1141307937.qw
drwx----- 2 vpopmail vchkpw 93 2006-03-18 17:10 courierimapkeywords
-rw-r--r-- 1 vpopmail vchkpw 23 2006-02-25 18:55 courierimapsubscribed
-rw-r--r-- 1 vpopmail vchkpw 45 2006-03-18 17:03 courierimapuidb
drwx----- 2 vpopmail vchkpw 44 2006-03-18 17:03 cur
drwx----- 2 vpopmail vchkpw 6 2006-03-18 17:03 new
drwx----- 2 vpopmail vchkpw 6 2006-03-18 17:10 tmp
```

Mais où sont les répertoires créés ?

```
janus:/home/vpopmail/domains/maison.mrs/postmaster/Maildir# ls -la
total 16
drwx----- 13 vpopmail vchkpw 4096 2006-03-18 17:10 .
drwx----- 3 vpopmail vchkpw 35 2006-02-25 16:22 ..
-rw----- 1 vpopmail vchkpw 45 2006-03-02 14:58 1141307937.qw
drwx----- 6 vpopmail vchkpw 103 2006-03-18 17:10 .Archives
drwx----- 5 vpopmail vchkpw 77 2006-03-18 17:10 .Archives.2005
drwx----- 5 vpopmail vchkpw 77 2006-03-18 17:10 .Archives.2006
drwx----- 2 vpopmail vchkpw 93 2006-03-18 17:10 courierimapkeywords
-rw-r--r-- 1 vpopmail vchkpw 23 2006-02-25 18:55 courierimapsubscribed
-rw-r--r-- 1 vpopmail vchkpw 45 2006-03-18 17:03 courierimapuidb
drwx----- 2 vpopmail vchkpw 44 2006-03-18 17:03 cur
drwx----- 6 vpopmail vchkpw 103 2006-03-18 17:00 .Drafts
drwx----- 2 vpopmail vchkpw 6 2006-03-18 17:03 new
drwx----- 6 vpopmail vchkpw 103 2006-03-18 17:00 .Queue
drwx----- 6 vpopmail vchkpw 126 2006-03-18 17:01 .Sent
drwx----- 2 vpopmail vchkpw 6 2006-03-18 17:10 tmp
drwx----- 6 vpopmail vchkpw 126 2006-03-18 17:01 .Trash
```

Ils sont cachés...

Nous avons bien avancé. Nous pouvons installer aussi courier-imap-ssl, à condition de gérer proprement les certificats nécessaires.

Ezmlm-idx

Des patches, des patches...

Ezmlm est un robot de listes de diffusions conçu par D.J.Bernstein. Ezmlm-idx est un "patch" qui accroît considérablement les possibilités de ezmlm. Il serait dommage de s'en priver. Voyez <http://www.ezmlm.org/readme.html> pour plus de détails.

Ezmlm et ezmlm-idx sont dans nos commissions. L'installation en est très simple, il n'y a rien de spécial à configurer :

```
janus# cd /usr/local/src
janus# tar -xzf tar/ezmlm-0.53.tar.gz
janus# tar -xzf tar/ezmlm-idx-0.443.tar.gz
janus# mv ezmlm-idx-0.443/* ezmlm-0.53/
janus# cd ezmlm-0.53
janus# patch -p0 < idx.patch
janus# make
janus# make fr # si vous voulez que le robot vous parle en français
janus# make setup
```

La gestion des listes

Autant ezmlm est simple à installer, autant c'est une chose dont la maîtrise est loin d'être évidente. Il est très probable que dans la plupart des cas, la gestion des listes depuis qmailadmin suffise.

Les listes peuvent être administrées sur le serveur au moyen des quelques commandes, et, suivant la configuration de ces listes, le propriétaire et les modérateurs peuvent administrer à distance par envoi de messages à des adresses spécifiques à chaque liste.

Entrer dans les détails nous mènerait beaucoup trop loin. Vous avez le "man ezmlm" dont la lecture est un vrai régal, de même que <http://cr.yip.to/ezmlm.html>. Enfin, pour vous achever, vous avez ceci : <http://www.ie.ezmlm.org/faq/FAQ.html>

Nous attendrons d'avoir installé qmailadmin pour tester ezmlm, ce sera tout de même plus simple.

Dans la salle des machines...

Les outils ezmlm se retrouvent dans /usr/local/bin/ezmlm.

Lorsqu'une liste est créée, nous retrouvons dans /home/vpopmail/domains/<le domaine concerné>:

- un répertoire au nom de la liste,
- une série de liens .qmail-xxx.

Nous savons que si un fichier .qmail-<utilisateur> existe, vdelivermail le consultera. Dans le cas d'une liste de diffusion, voyons ce que cela donne. Imaginons qu'une liste "demo" soit créée (ce qui sera fait dans le chapitre suivant, par l'intermédiaire de qmailadmin). Nous avons donc sur notre machine "janus", un répertoire : /home/vpopmail/domains/maison.mrs/demo et un fichier /home/vpopmail/domains/maison.mrs/.qmail-demo. En réalité, ce n'est pas un fichier mais un lien qui pointe sur /home/vpopmail/domains/maison.mrs/demo/editor.

Ce fichier contient :

```
|/usr/local/bin/ezmlm/ezmlm-reject '/home/vpopmail/domains/maison.mrs/demo'
|/usr/local/bin/ezmlm/ezmlm-issubn '/home/vpopmail/domains/maison.mrs/demo'
```



```
'/home/vpopmail/domains/maison.mrs/demo/digest'  
'/home/vpopmail/domains/maison.mrs/demo/allow'  
'/home/vpopmail/domains/maison.mrs/demo/mod'  
|| { echo "Seuls les abonnés peuvent poster des messages. Si vous  
        êtes abonné, faites suivre ce message a  
        demo-owner@maison.mrs pour que votre nouvelle adresse  
        soit prise en compte. (#5.7.2)";  
        exit 100 ;  
    }  
|/usr/local/bin/ezmlm/ezmlm-send '/home/vpopmail/domains/maison.mrs/demo'  
|/usr/local/bin/ezmlm/ezmlm-warn '/home/vpopmail/domains/maison.mrs/demo' || exit 0
```

Clairement, vdelivermail va transmettre le message à divers outils de ezmlm.

- ezmlm-reject est un outil qui permet de rejeter certains types de messages (man ezmlm-reject pour plus de détails). Si le message passe ce filtrage, il est envoyé à :
- ezmlm-issubn, qui vérifie si l'émetteur du message est inscrit sur la liste (dans le cas de listes où seuls les inscrits peuvent poster) (man ezmlm-issubn pour plus de détails). Si l'émetteur n'est pas inscrit, il reçoit une notification de non distribution. Sinon, le message est transmis à :
- ezmlm-send, qui va distribuer le message aux abonnés (man ezmlm-send...);
- ezmlm-warn, enfin, gèrera les messages qui sont revenus, à cause d'un problème de livraison à un ou plusieurs abonnés (man...).

Normalement, il n'est pas nécessaire, ni même souhaitable, de modifier ces choses manuellement. Dans les pires cas, utilisez ezmlm-sub (ou ezmlm-unsub) pour modifier les listes de ceux qui sont inscrits, ou simplement autorisés à poster.

Il est en effet possible, sur une liste où en principe, seuls les inscrits peuvent poster, d'autoriser cependant certains émetteurs, à poster, même s'ils ne sont pas inscrits. Cette démarche est utile, lorsqu'un automate doit envoyer des notifications à une liste. Le répertoire allow/senders semble conçu dans ce but, mais son contenu (de même que celui des divers "senders" répartis çà et là) ne peut être modifié manuellement. Désolé, mais "man ezmlm-sub" reste la meilleure source d'informations pour ce genre de manipulations.

Rassurez-vous, encore une fois, qmailadmin vous permettra dans la plupart des cas de gérer facilement vos listes. Il reste cependant nécessaire d'avoir une petite idée du fonctionnement du moteur, pour gérer les cas "hors norme".

qmailadmin

Autorespond

Il paraît à priori hors sujet de traiter autorespond ici, mais qmailadmin prend en charge ce répondeur automatique et la simplicité de son installation ne nécessite pas une page dédiée. Si vous n'en voulez pas, ne l'installez pas et pensez à désactiver sa gestion dans la compilation de qmailadmin.

Si vous souhaitez l'installer, il est lui aussi dans nos commissions :

```
janus# cd /usr/local/src
janus# tar -xzf tar/autorespond-2.0.4.tar.gz
janus# cd autorespond-2.0.4
janus# patch -p1 < ../tar/autorespond-2.0.4-2.0.5.patch
janus# make
janus# make install
```

qmailadmin

Qmailadmin est un script cgi à l'usage non seulement du postmaster, qui peut l'utiliser pour gérer les comptes, les listes et quasiment tout ce qu'il y a à gérer sur notre système, mais aussi à l'usage de tout possesseur d'un compte de messagerie. Bien entendu, dans ce cas, l'utilisateur n'aura accès qu'aux paramètres de son compte.

```
janus# cd /usr/local/src
janus# tar -xzf tar/qmailadmin-1.2.9.tar.gz
janus# cd qmailadmin-1.2.9
```

Il y a de nombreuses options de configuration sur qmailadmin :

```
janus# ./configure --help
`configure' configures this package to adapt to many kinds of systems.

Usage: ./configure [OPTION]... [VAR=VALUE]...

To assign environment variables (e.g., CC, CFLAGS...), specify them as
VAR=VALUE. See below for descriptions of some of the useful variables.

Defaults for the options are specified in brackets.

Configuration:
  -h, --help                display this help and exit
  --help=short              display options specific to this package
  --help=recursive          display the short help of all the included packages
  -V, --version             display version information and exit
  -q, --quiet, --silent    do not print `checking...' messages
  --cache-file=FILE        cache test results in FILE [disabled]
  -C, --config-cache        alias for `--cache-file=config.cache'
  -n, --no-create           do not create output files
  --srcdir=DIR              find the sources in DIR [configure dir or `..']

Installation directories:
  --prefix=PREFIX           install architecture-independent files in PREFIX
                           [/usr/local]
  --exec-prefix=EPREFIX    install architecture-dependent files in EPREFIX
                           [PREFIX]

By default, `make install' will install all the files in
`/usr/local/bin', `/usr/local/lib' etc. You can specify
an installation prefix other than `/usr/local' using `--prefix',
for instance `--prefix=$HOME'.

For better control, use the options below.
```

```

Fine tuning of the installation directories:
--bindir=DIR          user executables [EPREFIX/bin]
--sbindir=DIR         system admin executables [EPREFIX/sbin]
--libexecdir=DIR     program executables [EPREFIX/libexec]
--datadir=DIR        read-only architecture-independent data [PREFIX/share]
--sysconfdir=DIR     read-only single-machine data [PREFIX/etc]
--sharedstatedir=DIR modifiable architecture-independent data [PREFIX/com]
--localstatedir=DIR modifiable single-machine data [PREFIX/var]
--libdir=DIR         object code libraries [EPREFIX/lib]
--includedir=DIR     C header files [PREFIX/include]
--oldincludedir=DIR C header files for non-gcc [/usr/include]
--infodir=DIR        info documentation [PREFIX/info]
--mandir=DIR         man documentation [PREFIX/man]

Program names:
--program-prefix=PREFIX      prepend PREFIX to installed program names
--program-suffix=SUFFIX      append SUFFIX to installed program names
--program-transform-name=PROGRAM run sed PROGRAM on installed program names

System types:
--build=BUILD      configure for building on BUILD [guessed]
--host=HOST        cross-compile to build programs to run on HOST [BUILD]

Optional Features:
--disable-FEATURE do not include FEATURE (same as --enable-FEATURE=no)
--enable-FEATURE[=ARG] include FEATURE [ARG=yes]
--disable-dependency-tracking Speeds up one-time builds
--enable-dependency-tracking Do not reject slow dependency extractors
--enable-vpopmaildir=DIR Home directory of vpopmail.
--enable-htmldir=DIR HTTP server's base html dir.
--enable-cgibindir=DIR HTTP server's cgi-bin directory.
--enable-imageurl=URL URL to image files /images/qmailadmin.
--enable-imagedir=DIR Where to put the image files DocumentRoot/images/qmailadmin.
--enable-htmllibdir=DIR qmailadmin HTML library directory.
--enable-qmaildir=DIR directory where qmail is installed.
--enable-ezmlmdir=PATH Path to ezmlm binaries or =n to disable.
--enable-cgipath=PATH URL path for cgi /cgi-bin/qmailadmin.
--enable-autoresponder-path=PATH Path to autoresponder directory.
--enable-vpopuser=USER User vpopmail was installed as vpopmail.
--enable-vpopgroup=GROUP Group vpopmail was installed as vchkpw.
--disable-ezmlm-mysql Disable MySQL options for mailing lists.
--enable-maxusersperpage=# Number of users to display on a page [15].
--enable-maxaliasesperpage=# Number of aliases to display on a page [25].
--enable-no-cache Prohibit caching via http headers.
--disable-ipauth Disable IP address checks (NAT/proxy systems may not use the same IP for the whole session).
--disable-user-index Don't display user index on show users page.
--enable-modify-quota Allow domain admin (postmaster) to modify user quotas.
--enable-domain-autofill Autofill the domain on login page based on the hostname that accesses qmailadmin.
--enable-modify-spam Allow users to enable/disable spam checking.
--enable-spam-command=CMD Remember to use '|preline' at the beginning,
|preline /usr/local/bin/maildrop /etc/mailfilter.
--enable-help Display help links on login page.

Some influential environment variables:
CC C compiler command
CFLAGS C compiler flags
LDFLAGS linker flags, e.g. -L<lib dir> if you have libraries in a nonstandard directory <lib dir>
CPPFLAGS C/C++ preprocessor flags, e.g. -I<include dir> if you have headers in a nonstandard directory <include dir>
CPP C preprocessor

Use these variables to override the choices made by `configure' or to help it to find libraries and programs with nonstandard names/locations.

```

Nous allons configurer qmailadmin comme ceci :

```

janus# ./configure \
--enable-help \
--enable-cgibindir=/usr/lib/cgi-bin \
--enable-htmldir=/var/www \

```

```
--enable-maxusersperpage=25 \  
--enable-modify-quota \  
--disable-ezmlm-mysql
```

Ou, avec la possibilité d'aiguiller sur le tri antispam :

```
janus# ./configure \  
--enable-help \  
--enable-cgibindir=/usr/lib/cgi-bin \  
--enable-htmldir=/var/www \  
--enable-maxusersperpage=25 \  
--enable-modify-quota \  
--disable-ezmlm-mysql \  
--enable-modify-spam \  
--enable-spam-command="| /usr/bin/mailedrop ../mailfilter"
```

Ceci nous permettra, comme nous le verrons plus loin, d'activer ou non un tri des messages en fonction du marquage opéré par spamassassin. Les options surlignées permettront d'ajouter à l'interface une case à cocher qui permettra de créer ou non un fichier .qmail dans le répertoire virtuel de l'utilisateur.

Ce qui nous donne en fin de configuration, un résumé comme celui-ci :

```
qmailadmin 1.2.9  
Current settings  
-----  
cgi-bin dir = /usr/lib/cgi-bin  
html dir = /var/www  
image dir = /var/www/images/qmailadmin  
image URL = /images/qmailadmin  
template dir = /usr/local/share/qmailadmin  
qmail dir = /var/qmail  
vpopmail dir = /home/vpopmail  
autorespond dir = /usr/bin  
ezmlm dir = /usr/local/bin/ezmlm  
ezmlm idx = yes  
mysql for ezmlm = no  
help = yes  
modify quota = yes  
domain autofill = no  
modify spam check = no
```

Ou encore, avec antispam :

```
qmailadmin 1.2.9  
Current settings  
-----  
cgi-bin dir = /usr/lib/cgi-bin  
html dir = /var/www  
image dir = /var/www/images/qmailadmin  
image URL = /images/qmailadmin  
template dir = /usr/local/share/qmailadmin  
qmail dir = /var/qmail  
vpopmail dir = /home/vpopmail  
autorespond dir = /usr/bin  
ezmlm dir = /usr/local/bin/ezmlm  
ezmlm idx = yes  
mysql for ezmlm = no  
help = yes  
modify quota = yes  
domain autofill = no  
modify spam check = yes  
spam command = | /usr/bin/mailedrop ../mailfilter
```

Il ne reste plus qu'à compiler et à installer :

```
janus# make  
janus# make install-strip
```

Osez installer qmail

© Christian CALECA
<http://christian.caleca.free.fr/qmail/>

Et nous devrions avoir un qmailadmin en état de fonctionner :



Nous pouvons ici, très simplement, voir les comptes existants, en créer, en supprimer, gérer les redirections, etc.

Le formulaire de création d'une liste de diffusion est suffisamment complet pour répondre à la plupart des cas de figure. Pour pouvoir en faire une copie d'écran complète, j'ai dû adopter une police vraiment petite :

The screenshot shows a Mozilla Firefox browser window titled "QmailAdmin maison.mrs". The address bar shows the URL "http://janus.maison.mrs/cgi-bin/qmailadmin/com/addmailir". The main content area is a form titled "Ajouter une Liste de diffusion".

Form fields:

- Nom de la Liste de diffusion: [] @maison.mrs
- E-mail du responsable: []
- Préfixe: []

Messagerie de signalisation:

- Tout le monde peut poster.
- Seulement les abonnés peuvent poster, tous les autres sont rejetés.
- Seulement les abonnés peuvent poster, tous les autres vont aux modérateurs pour approbation.
- Seulement les modérateurs peuvent poster, tous les autres sont rejetés.
- Seulement les modérateurs peuvent poster, tous les autres passent par les modérateurs pour approbation.

Liste des options:

- Les réponses sont envoyées vers:
 - L'expéditeur d'origine
 - la liste entière
 - l'adresse: []
- Inclure un pied de page à la fin de chaque message.
- Installer une version résumé de la liste. (Nommé Nom de la liste -digest.)
- Demandes de services envoyées à Nom de la liste -request. (En plus de Nom de la liste -subscribe, Nom de la liste -unsubscribe, etc., les utilisateurs peuvent envoyer des commandes à Nom de la liste -request.)

Administration à distance:

- Permettre l'administration à distance par des modérateurs.
- Faire une liste privée. (Seulement les admins peuvent abonner, désabonner, accéder à l'index, etc...)

Les admins à distance peuvent :

- Regarder et récupérer la liste des abonnés.
- Éditer les fichiers du répertoire texte. (par exemple, le pied de page, les fichiers d'aide, etc...)

Demandes d'abonnement:

Demandes d'inscription à la liste :

- Exige la confirmation par une réponse à un message envoyé à l'adresse d'abonnement.
- Exige l'approbation d'un modérateur.

Demandes de désabonnement à la liste :

- Exige la confirmation par une réponse à un message envoyé à l'adresse d'abonnement.

Avertissement: si vous n'avez pas besoin de la confirmation de l'expéditeur, il sera possible que des personnes usurent l'email pour abonner ou désabonner quelqu'un de cette liste sans leur autorisation.

Archives des Messages:

- Messages de liste d'archives. La récupération d'archives est limitée aux modérateurs []
- Indexation des archives pour l'accès web via ezmlm-cgi.

Buttons: [Ajouter]

Footer: [[Listes de diffusion](#) | [Menu principal](#) | [Déconnexion](#)]

Bottom bar: Terminé

Si vous n'êtes pas familier avec les options des listes de diffusion, prenez le temps nécessaire pour bien comprendre les diverses options proposées.

Nous allons créer une liste "demo" :



Notez que le responsable de la liste a été inscrit :



Il est bien sûr possible de revenir sur le paramétrage de cette liste. Toute personne désireuse de s'abonner peut le faire par le simple envoi d'un message vide à demo-subscribe@maison.mrs. La liste est paramétrée de façon à ce qu'une confirmation soit demandée au nouvel arrivant, ceci afin d'éviter que n'importe qui inscrive n'importe quelle adresse. Faisons-le pour un compte quelconque, créé pour la circonstance : prof@maison.mrs par exemple. Immédiatement, prof reçoit le message suivant :

```
Return-Path: <demo-return-@maison.mrs>
Delivered-To: prof@maison.mrs
Received: (qmail 6014 invoked by uid 89); 18 Mar 2006 17:14:34 -0000
Mailing-List: contact demo-help@maison.mrs; run by ezmlm
List-Help: <mailto:demo-help@maison.mrs>
List-Post: <mailto:demo@maison.mrs>
```

```
List-Subscribe: <mailto:demo-subscribe@maison.mrs>
Date: 18 Mar 2006 17:14:34 -0000
Message-ID: <1142702074.6013.ezmlm@maison.mrs>
From: demo-help@maison.mrs
To: prof@maison.mrs
Delivered-To: responder for demo@maison.mrs
Received: (qmail 6009 invoked by uid 89); 18 Mar 2006 17:14:34 -0000
Received: from unknown (HELO localhost.localdomain) (192.168.0.15)
  by 0 with SMTP; 18 Mar 2006 17:14:34 -0000
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Reply-To: demo-sc.1142702074.klgmfilccjnlmkpofjk-prof=maison.mrs@maison.mrs
Subject: confirm subscribe to demo@maison.mrs
```

Hi! This is the ezmlm program. I'm managing the demo@maison.mrs mailing list.

I'm working for my owner, who can be reached at demo-owner@maison.mrs.

To confirm that you would like

prof@maison.mrs

added to the demo mailing list, please send an empty reply to this address:

demo-sc.1142702074.klgmfilccjnlmkpofjk-prof=maison.mrs@maison.mrs

Usually, this happens when you just hit the "reply" button. If this does not work, simply copy the address and paste it into the "To:" field of a new message.

This confirmation serves two purposes. First, it verifies that I am able to get mail through to you. Second, it protects you in case someone forges a subscription request in your name.

Some mail programs are broken and cannot handle long addresses. If you cannot reply to this request, instead send a message to <demo-request@maison.mrs> and put the entire address listed above into the "Subject:" line.

--- Administrative commands for the demo list ---

I can handle administrative requests automatically. Please do not send them to the list address! Instead, send your message to the correct command address:

For help and a description of available commands, send a message to: <demo-help@maison.mrs>

To subscribe to the list, send a message to: <demo-subscribe@maison.mrs>

To remove your address from the list, just send a message to the address in the ``List-Unsubscribe'' header of any list message. If you haven't changed addresses since subscribing, you can also send a message to: <demo-unsubscribe@maison.mrs>

For addition or removal of addresses, I'll send a confirmation message to that address. When you receive it, simply reply to it to complete the transaction.

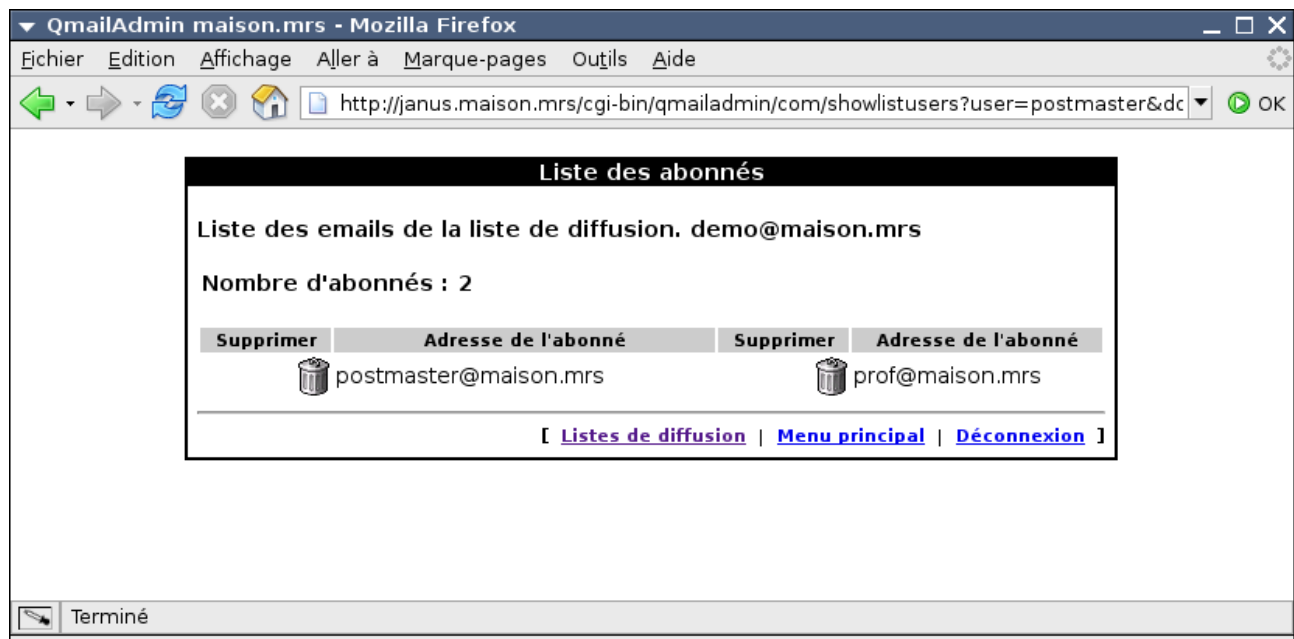
If you need to get in touch with the human owner of this list, please send a message to:

<demo-owner@maison.mrs>

Please include a FORWARDED list message with ALL HEADERS intact to make it easier to help you.


```
--- Enclosed is a copy of the request I received.  
  
Return-Path: <prof@maison.mrs>  
Received: (qmail 6009 invoked by uid 89); 18 Mar 2006 17:14:34 -0000  
Received: from unknown (HELO localhost.localdomain) (192.168.0.15)  
  by 0 with SMTP; 18 Mar 2006 17:14:34 -0000  
Date: Sat, 18 Mar 2006 18:14:34 +0100  
From: Prof <prof@maison.mrs>  
To: demo-subscribe@maison.mrs  
Message-ID: <20060318181434.4f24e211@localhost.localdomain>  
X-Mailer: Sylpheed-Claws 1.9.14 (GTK+ 2.8.6; i486-pc-linux-gnu)  
Mime-Version: 1.0  
Content-Type: text/plain; charset=US-ASCII  
Content-Transfer-Encoding: 7bit
```

Si Prof répond à ce message, il sera immédiatement inscrit sur la liste :



Les révocations se font de façon analogue, en envoyant un message à demo-unsubscribe@maison.mrs.

Bien sûr, postmaster peut faire tout ce qu'il veut depuis qmailadmin.

Filtrage des mails

Discussion sur les stratégies de filtrage

Virus et vers

Un antivirus comme Clamav génère à priori très peu de "faux positifs" (erreur d'appréciation qui consiste à penser qu'un message est viral, alors qu'il ne l'est pas). Autrement dit, un message considéré comme viral le sera très probablement. Il est donc pertinent de rejeter d'entrée de tels messages. Notez toutefois que ce type de traitement générera par défaut un message de notification de refus qui, la plupart du temps, ne sera d'aucune utilité, l'adresse de l'émetteur étant généralement falsifiée. Nous verrons qu'il est possible d'éviter cette notification.

Spam

Le spam, en revanche est un cas un peu particulier. Spamassassin utilise un certain nombre d'indices pour attribuer un score à un message filtré. Si ce score dépasse une certaine valeur, le message sera considéré comme du spam.

Clairement, un antispam génèrera des faux positifs, des message supposés indésirables alors que ce n'est pas le cas. De même, des messages non désirés pourront passer quand même.

Enfin, un message non désiré par un utilisateur peut être acceptable pour un autre.

Au final, il est plus qu'hasardeux de choisir de rejeter un message estampillé comme non souhaité par spamassassin, sans l'accord du destinataire.

Où vaut-il mieux filtrer ?

Il est possible de filtrer à deux niveaux :

- lorsque le message entre sur le smtp (qmail-queue),
- lorsque le message va être placé dans la boîte aux lettres du destinataire, en ajoutant dans la chaîne un MDA comme procmail ou maildrop.

Nous allons filtrer à l'entrée, en rejetant les messages viraux, mais en laissant passer les spams, après les avoir estampillés comme tels. L'utilisateur pourra lui-même décider de l'avenir de ces messages.

Clamav

Antivirus sous licence GPL, existe en paquetage binaire sur toute bonne distribution. Debian en est une, donc :

```
janus# apt-get install clamav clamav-daemon
....
Les paquets supplémentaires suivants seront installés :
clamav-base clamav-freshclam libclamav1 clamav-daemon
Paquets suggérés :
unrar lha clamav-docs
Paquets recommandés :
arj unzoo
Les NOUVEAUX paquets suivants seront installés :
clamav clamav-base clamav-freshclam libclamav1 clamav-daemon
```

```
...
```

L'installation vous demandera la méthode à choisir pour mettre à jour la base de virus. Personnellement, j'utilise le "daemon".

Choisissez ensuite le miroir le plus proche.

Enfin, installez les paquets suggérés et recommandés, si ce n'est déjà fait :

```
janus# apt-get install arj unzoo unrar lha clamav-docs
```

Avec les dépendances, nous devrions avoir les paquets suivants :

- clamav
- clamav-base
- clamav-daemon
- clamav-docs
- clamav-freshclam
- libclamav1

L'installation a créé l'utilisateur clamav :

```
janus# cat /etc/passwd | grep clamav
```

```
clamav:x:110:110::/var/lib/clamav:/bin/false
```

Entrer ici dans le détail du fonctionnement de clamav nous mènerait trop loin. Pour aller très vite, vous trouverez après installation, deux services :

```
janus# ps -aef | grep clam
clamav  3941      1  0  08:39 ?        00:00:01 /usr/sbin/clamd
clamav  3992      1  0  08:39 ?        00:00:00 /usr/bin/freshclam -p /var/run/clamav/freshclam.pid -d
--quiet
```

- freshclam se charge de surveiller la publication de nouvelles signatures de virus, et de garder la base locale à jour,
- clamd est le service antivirus, à l'écoute d'une connexion UNIX ou TCP.

Bien entendu, ces services sont démarrés classiquement par des scripts dans /etc/init.d. Rien n'interdit, bien au contraire, de les faire démarrer plutôt par les daemontools.

Spamassassin

Antispam sous licence GPL, existe en paquetage binaire sur toute bonne distribution. Debian en est une, donc :

```
janus# apt-get install spamassassin spamc
...
Les paquets supplémentaires suivants seront installés :
libdigest-hmac-perl libdigest-shal-perl libnet-dns-perl libnet-ip-perl libsocket6-perl
Paquets suggérés :
razor libnet-ident-perl libio-socket-ssl-perl dcc-client pyzor
Paquets recommandés :
libmail-spf-query-perl
Les NOUVEAUX paquets suivants seront installés :
libdigest-hmac-perl libdigest-shal-perl libnet-dns-perl libnet-ip-perl libsocket6-perl spamassassin
spamc
...
```

A vous de décider des paquets suggérés à ajouter.

Là encore, entrer ici dans le détail du fonctionnement de spamassassin nous mènerait trop loin. Il y a tout de même quelques petites choses à dire.

Debian, par défaut, inhibe le démarrage de spamassassin. Le paquetage fourni pilote spamd (le service), par un script "classique" dans /etc/init.d. Toujours par défaut, si spamd démarrait, il le ferait

sous root, ce qui est potentiellement dangereux, voire inutilisable avec Vpopmail.

Ici, nous allons autoriser le démarrage de spamd sous l'identité de Vpopmail en modifiant comme suit le fichier /etc/default/spamassassin :

```
# /etc/default/spamassassin
# Duncan Findlay

# WARNING: please read README.spamd before using.
# There may be security risks.

# Change to one to enable spamd
ENABLED=1

# Options
# See man spamd for possible options. The -d option is automatically added.

# SpamAssassin uses a preforking model, so be careful! You need to
# make sure --max-children is not set to anything higher than 5,
# unless you know what you're doing.

#OPTIONS="--create-prefs --max-children 5 --helper-home-dir"
OPTIONS="-d -x -u vpopmail -m 5 -H"

# Pid file
# Where should spamd write its PID to file? If you use the -u or
# --username option above, this needs to be writable by that user.
# Otherwise, the init script will not be able to shut spamd down.
PIDFILE="/var/run/spamd.pid"

# Set nice level of spamd
#NICE="--nicelevel 15"
```

Ceci est fait au cas où le filtrage des spams serait finalement décidé au niveau de Vpopmail. Si le filtrage est fait par un appel de simscan, ce qui sera le cas ici, le choix de l'utilisateur peut être différent.

Nous démarrons le service par :

```
janus# /etc/init.d/spamassassin start
```

Et vérifions que spamd est bien démarré :

```
janus# ps -aef | grep spam
root 5836 1 0 17:29 ? 00:00:03 /usr/sbin/spamd -d -c -u vpopmail -m 5 -H -d -pidfile=/var/run/spamd.p
d
vpopmail 5838 5836 0 17:30 ? 00:00:00 spamd child
vpopmail 5839 5836 0 17:30 ? 00:00:00 spamd child
```

Par la suite, ce sera sans doute une très bonne idée de piloter plutôt spamd via les daemontools.

Simscan

En utilisant la version patchée "netqmail", il est possible d'exploiter simscan, au niveau de qmail-queue pour filtrer les messages entrants dans notre SMTP.

Simscan va permettre de se raccorder à un antivirus (Clamav, par exemple) et (ou) à un antisпам (spamassassin). Travaillant au niveau de l'entrée des messages, il sera possible pour Qmail de rejeter des messages considérés comme indésirables. Simscan supporte d'autres antivirus, si vous préférez.

Dans la mesure où nous avons décidé de transmettre les spams, en se contentant de les marquer comme tels, nous pouvons aussi bien filtrer au niveau de qmail-queue avec simscan, qu'au niveau de la distribution locale (vdlivermail), en utilisant les services de procmail ou de maildrop. Nous faisons ici le choix de filtrer les virus et les spams au niveau de qmail-queue.

Simscan peut adopter des comportements divers suivant le résultat du filtrage antivirus et (ou)

antispam, et il convient de bien réfléchir au problème, parce que ce comportement se décide en grande partie dans les options de compilation.

Il est donc utile de passer le temps nécessaire à une bonne compréhension de la documentation fournie⁶.

D'une première lecture de ce document, il ressort que :

- simscan et clamav doivent tourner sous le même utilisateur, nous choisirons clamav,
- nous ne bloquerons pas les messages considérés comme du spam, mais nous le ferons pour ceux qui contiennent des virus,
- nous ferons un "drop" des messages viraux, sans notification. Si vous vous demandez pourquoi, rappelez-vous de la quantité de notifications de ce genre que vous avez déjà reçues, alors même que vous êtes certains de ne jamais avoir envoyé de virus à personne.

A l'heure où ces lignes sont écrites, il existe une version 1.2 de simscan en test, sur le site d'inter⁷. Nous allons cependant installer la version 1.1. Il nous faudra la télécharger.

```
janus# cd /usr/local/src/tar
janus# wget http://shupp.org/software/simscan-1.1.tar.gz
janus# cd ../
janus# tar -xzf tar/simscan-1.1.tar.gz
janus# cd simscan-1.1
```

Adoptons une configuration de ce type (attention, les chemins sont prévus pour une distribution Debian) :

```
janus# ./configure \
--enable-user=clamav \
--enable-clamav=y \
--enable-clamdsclan=/usr/bin/clamdsclan \
--enable-spam=y \
--enable-spam-passthru=y \
--enable-spamc=/usr/bin/spamc \
--enable-qmail-queue=/var/qmail/bin/qmail-queue \
--enable-dropmsg=y \
--enable-per-domain=y \
--enable-received=y \
--enable-spamassassin-path=/usr/bin/spamassassin \
--enable-sigtool-path=/usr/bin/sigtool \
--enable-clamavdb-path=/var/lib/clamav
```

La fin de la configuration donne le résumé suivant :

```
Current settings
-----
user                = clamav
qmail directory     = /var/qmail
work directory      = /var/qmail/simscan
control directory   = /var/qmail/control
qmail queue program = /var/qmail/bin/qmail-queue
clamdsclan program  = /usr/bin/clamdsclan
clamav scan         = ON
trophie scanning    = OFF
attachement scan    = OFF
ripmime program     = OFF
custom smtp reject  = OFF
drop message        = ON
regex scanner       = OFF
quarantine processing = OFF
domain based checking = ON
add received header = ON
spam scanning       = ON
spamc program       = /usr/bin/spamc
spamc arguments     =
spamc user          = OFF
```

⁶ <http://qmailwiki.inter7.com/Simscan/README>

⁷ <http://www.inter7.com/?page=simscan>

```
spam passthru = ON
```

Compilation et installation :

```
janus# make
janus# make install-strip
```

Le répertoire de travail a bien été créé, avec "clamav" comme propriétaire :

```
janus# cd /var/qmail
janus# ls -l
total 12
drwxr-sr-x 2 alias qmail 75 2005-12-29 18:48 alias
drwxr-xr-x 2 root qmail 4096 2006-01-05 16:06 bin
drwxr-xr-x 2 root qmail 137 2005-12-28 18:14 boot
lrwxrwxrwx 1 root root 10 2005-12-28 17:37 control -> /etc/qmail
drwxr-xr-x 3 root root 20 2006-01-01 18:18 control.origin
drwxr-xr-x 2 root qmail 4096 2005-12-28 18:14 doc
drwxr-xr-x 10 root qmail 94 2005-12-28 18:14 man
drwxr-x--- 11 qmailq qmail 109 2005-12-28 18:14 queue
-rwxr-xr-x 1 root root 284 2005-12-30 14:20 rc
drwxr-x--- 2 clamav root 6 2006-01-05 16:06 simscan
drwxr-xr-x 5 root root 59 2006-01-02 13:55 supervise
drwxr-xr-x 2 root qmail 47 2006-01-01 18:19 users
```

Quant à simscan, il a été installé dans /var/qmail/bin :

```
janus# ls -l bin | grep simscan
-rws--x--x 1 clamav root 12524 2006-01-05 16:06 simscan
-rwxr-xr-x 1 root root 9396 2006-01-05 16:06 simscanmk
```

Et le propriétaire en est bien "clamav".

Bon. Et comment tout cela va fonctionner ?

Par l'entremise d'une variable d'environnement, "QMAILQUEUE", que l'on passera à Qmail par le tcpserver. Il nous faut modifier /home/vpopmail/etc/tcp.smtp (et remettre à jour tcp.smtp.cdb).

```
janus# qmailctl stop
Stopping qmail...
  qmail-smtpd
  qmail-send
```

```
janus# nano /home/vpopmail/etc/tcp.smtp
```

Par exemple, en y mettant ceci :

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.:allow,RELAYCLIENT="",QMAILQUEUE="/var/qmail/bin/simscan"
:allow,QMAILQUEUE="/var/qmail/bin/simscan"
```

N'oublions pas de mettre à jour la base .cdb :

```
janus# qmailctl cdb
```

Comme nous avons choisi l'option "per-domain" (probablement inutile, mais qui permet tout de même plus de souplesse de configuration, le cas échéant), nous devons aussi créer une règle par défaut dans /var/qmail/control :

```
janus# echo ":clam=yes,spam=yes,spam_passthru=yes" > /var/qmail/control/simcontrol
```

Et créer la fichier .cdb correspondant :

```
janus# /var/qmail/bin/simscanmk
```

Comme nous avons choisi de marquer dans les en-têtes les versions des "scanners", nous devons créer la base simversions.cdb :

```
janus# /var/qmail/bin/simscanmk -g
```

Bien entendu, pour que les informations restent à jour, il faut relancer cette commande à chaque mise à jour d'un élément des scanners. Le moyen le plus simple (et pas forcément le plus propre), est de relancer, par exemple toutes les demi-heures, cette commande par crond.

Nous pouvons redémarrer Qmail :

```
janus# qmailctl start
```

Contrôles

Au moyen d'un hôte du réseau, par exemple avec Thunderbird, nous envoyons un message à `postmaster@maison.mrs`. Un message tout simple, juste pour voir si l'on retrouve les marques de `spamassassin` et de `clamav` dans l'en-tête.

```
Return-Path: <prof@maison.mrs>
Delivered-To: postmaster@maison.mrs
Received: (qmail 7850 invoked by uid 89); 23 Mar 2006 16:33:00 -0000
Received: by simscan 1.1.0 ppid: 7843, pid: 7845, t: 5.3135s
      scanners: clamav: 0.88/m:36/d:1353 spam: 3.1.0
X-Spam-Checker-Version: SpamAssassin 3.1.0 (2005-09-13) on janus.maison.mrs
X-Spam-Level:
X-Spam-Status: No, score=-1.4 required=5.0 tests=ALL_TRUSTED autolearn=failed
      version=3.1.0
Received: from unknown (HELO ?192.168.0.15?) (postmaster@maison.mrs@192.168.0.15)
by 0 with ESMTPA; 23 Mar 2006 16:32:55 -0000
Message-ID: <4422CDB7.9010702@maison.mrs>
Date: Thu, 23 Mar 2006 17:32:55 +0100
From: christian Caleca <prof@maison.mrs>
User-Agent: Mozilla Thunderbird 1.0.7 (X11/20051013)
X-Accept-Language: fr, en
MIME-Version: 1.0
To: postmaster <postmaster@maison.mrs>
Subject: test simscan
Content-Type: text/plain; charset=ISO-8859-1; format=flowed
Content-Transfer-Encoding: 7bit
```

Ca marche !

Le message est bien filtré par `clamav` et `spamassassin`.

Envoyons maintenant un message notoirement vérolé, pour vérifier le fonctionnement de `clamav`.

L'installation de `clamav` sur Debian place les logs dans `/var/log/clamav/clamav.log`.

Un forward d'un message compromis par un ver à la mode, fait apparaître dans ces logs :

```
Thu Mar 23 17:46:58 2006 -> SelfCheck: Database status OK.
Thu Mar 23 17:46:58 2006 -> /var/qmail/simscan/1143132418.79839.8977/msg.1143132418.79839.8977:
      Exploit.HTML.IFrame FOUND
```

Et le message n'arrive jamais à destination, sans notification, comme c'était prévu.

Livraison locale

Que faire des spams ?

Très clairement, la lecture de sa messagerie devrait passer par IMAP plutôt que par POP, mais chacun est libre, n'est-ce pas ?

Avec POP3

L'utilisateur n'aura accès qu'au répertoire "INBOX". Il ne pourra donc pas profiter des possibilités de dossiers de rangement distants. Dans un tel cas, il ne faut pas prendre l'initiative de placer les spams dans un dossier spécial, au moment de la livraison. En effet, ils vont s'y accumuler sans aucun moyen pour l'utilisateur de consulter ni de purger ce dossier.

La meilleure approche dans ce cas est sans doute de laisser l'utilisateur trier localement en fonction du marquage de spamassassin.

Avec IMAP

Ici, l'utilisateur peut accéder à un dossier de rangement. Deux choix sont alors possibles :

- trier les spams au moment de la livraison locale,
- laisser l'utilisateur trier lui-même, en fonction des "tags" ajoutés par spamassassin.

Bien sûr, c'est au client de choisir ce qu'il préfère.

Nous allons utiliser le MDA maildrop pour éventuellement filtrer à la livraison du message. L'utilisateur pourra décider si oui ou non il souhaite ce service, par l'intermédiaire de l'interface qmailadmin.

Maildrop

Composant de la suite "courier", maildrop est comparable à procmail. C'est un MDA (Mail Delivery Agent) qui pourra modifier son comportement en fonction d'un fichier de configuration.

Sur la Debian Etch, il y a deux paquetages Maildrop :

- maildrop,
- courier-maildrop

Les deux sont utilisables. Cependant, si vous envisagez d'utiliser le webmail sqwebmail, préférez courier-maildrop.

Il est possible, techniquement, de laisser le soin à chaque utilisateur de définir son fichier de configuration, mais pratiquement, ceci pose de nombreux problèmes. Nous nous contenterons donc d'un filtrage "passe partout", que l'utilisateur pourra activer ou non.

Lorsque nous avons configuré qmailadmin, nous avons éventuellement choisi les options :

```
--enable-modify-spam \  
--enable-spam-command="| /usr/bin/maildrop ../mailfilter"
```


La première permet de faire apparaître une case à cocher supplémentaire dans l'interface :

QmailAdmin maison.mrs - Mozilla Firefox

Fichier Edition Affichage Aller à Marque-pages Outils Aide

http://janus.maison.mrs/cgi-bin/qmailadmin/com/mc OK

Modifier l'utilisateur: prof@maison.mrs

Nom complet: prof

Nouveau mot de passe:

Mot de passe (à nouveau):

Quota (MO): NOQUOTA (Entrez NOQUOTA pour annuler les quotas)

Routing: Dévalider la notification d'absence

Supprimé

Rediriger le mail vers :

Valider la redirection et la copie

Valider la notification d'absence

Sujet de la notification :

Message de la notification d'absence :

Détection des spams ?

Modifier l'utilisateur

[[Compte Pop](#) | [Menu principal](#) | [Déconnexion](#)]

Terminé

Si cette case est cochée, un fichier .qmail sera créé dans le répertoire virtuel de l'utilisateur :

```
janus# ls -la /home/vpopmail/domains/maison.mrs/prof/
total 16
drwx----- 3 vpopmail vchkpw 48 2006-03-28 19:12 .
drwx----- 5 vpopmail vchkpw 4096 2006-03-26 19:18 ..
-rw----- 1 vpopmail vchkpw 4 2006-03-26 19:45 lastauth
drwx----- 11 vpopmail vchkpw 4096 2006-03-26 19:10 Maildir
-rw----- 1 vpopmail vchkpw 34 2006-03-28 19:12 .qmail
```

Et ce fichier contiendra ce qui a été défini dans la seconde option de configuration :

```
janus# cat /home/vpopmail/domains/maison.mrs/prof/.qmail
```

```
| /usr/bin/mailedrop ../mailfilter
```

Reste à étudier un peu la syntaxe des fichiers de filtrage mailedrop et de réaliser un fichier convenable dans /home/vpopmail/domains/maison.mrs/ :

```
janus# cat /home/vpopmail/domains/maison.mrs/mailfilter
```

```
SHELL="/bin/bash"
VHOME="/home/vpopmail/bin/vuserinfo -d $EXT@$HOST"

# Creation d'un dossier "Spam" s'il n'existe pas encore
DUMMY=`test -d $VHOME/Maildir/.Spam`
if ( $RETURNCODE == 1 )
{
    DUMMY="/usr/bin/mailedirmake -f Spam $VHOME/Maildir/"
    DUMMY=`echo INBOX.Spam >> $VHOME/Maildir/courierimapsubscribed`
}

# Distribution suivant le marquage de Spamassassin
if (/^X-Spam-Status: *Yes/)
{
    exception {
        to "$VHOME/Maildir/.Spam/"
    }
}
else
{
    to "$VHOME/Maildir/"
}
}
```

- vuserinfo est donc fourni avec Vpopmail et permet d'obtenir le chemin d'accès absolu au répertoire d'un utilisateur vpopmail :

```
janus# /home/vpopmail/bin/vuserinfo -d prof@maison.mrs
/home/vpopmail/domains/maison.mrs/prof
```

- mailedirmake vient avec courier et permet de créer simplement un répertoire de stockage dans la BAL d'un utilisateur.

Ici, le script teste la présence d'un répertoire IMAP "Spam" et le crée s'il n'existe pas, puis l'ajoute à la liste des dossiers souscrits par le client (ce n'est pas du tout obligé de le faire, surtout si le client sait correctement utiliser IMAP).

Dans la deuxième partie du script, si une ligne du message commence par "X-Spam-Status: " et contient "Yes", le message sera placé dans le répertoire Spam, sinon, il sera placé dans le répertoire de base.

Bien entendu, ceci n'est qu'un exemple, il est sûrement possible de faire mieux.

Il est possible aussi, par cette méthode, de faire passer les messages dans spamassassin à ce niveau plutôt que dans qmail-queue. A vous de voir ce qui vous convient le mieux.

mailedrop est un MDA très puissant, probablement plus que procmail, malheureusement assez mal documenté. Les utilisateurs rompus aux charmes de procmail pourront sans doute trouver une solution très comparable, avec leur MDA coutumier.

Conclusions

Améliorations possibles

Ce très long chapitre permettra, j'espère, à ceux qui souhaitent construire une solution de messagerie évoluée, de parvenir à un résultat honorable.

Il est possible de faire beaucoup plus simple, sans la gestion des domaines virtuels (Vpopmail), ou même encore plus compliqué, avec par exemple l'emploi de mysql ou d'un annuaire ldap pour enregistrer les utilisateurs.

Nous n'avons pas utilisé ucspi-tcp pour faire démarrer clamav ni spamassassin. Il serait sans doute préférable de le faire, si ce n'est que les paquetages de la distribution n'étant pas prévus pour ce type de fonctionnement, les mises à jour ultérieures nécessiteront plus d'attention. A vous de choisir un compromis entre fiabilité maximale et facilité de maintenance.

Nous n'avons pas non plus évoqué le cas des listes noires, grises ou blanches, qui permettent au niveau de qmail-queue de refuser, de faire perdre du temps ou d'accepter des messages de clients en fonction de leur adresse IP. Tout ceci est bien sûr également possible, en configurant Qmail convenablement. Dans ce cas, le patch "spamcontrol"⁸ pour Qmail peut vous être de quelque utilité. Ne me posez pas de questions à son sujet, je n'ai fait aucun essai avec.

Enfin, simscan n'a pas été employé avec toutes ses possibilités. Je vous encourage encore une fois à passer du temps sur sa documentation⁹.

Spamassassin nécessiterait à lui seul tout un chapitre. Ici, nous l'avons juste mis en place. Il reste à le configurer de façon efficace.

Nous n'avons pas installé de "webmail", mais c'est bien entendu possible. Vous pouvez par exemple utiliser squirrelmail ou sqwebmail, en utilisant les paquetages debian.

Quand tout va mal

L'arrêt ou le redémarrage des services pilotés par ucspi-tcp ne sont pas toujours des opérations très commodes à réaliser. Il peut se faire que le script qmailctl ne remplisse pas parfaitement son office.

En cas de problèmes de ce genre, la méthode la plus sûre est sans doute d'opérer de la sorte :

1. éditer le fichier /etc/inittab pour commenter la ligne :

```
SV:123456:respawn:/command/svscanboot,
```

2. forcer le processus init à relire son fichier de configuration avec la commande :

```
init q
```

3. tuer manuellement (kill) le processus svscan
4. tuer manuellement (kill) tous les processus gérés par "supervise",
5. ré éditer /etc/inittab pour décommenter la ligne :

```
SV:123456:respawn:/command/svscanboot,
```

6. re forcer le processus init à relire son fichier de configuration avec la commande :

```
init q
```

C'est long, c'est lourd, mais c'est le meilleur moyen pour repartir proprement lorsqu'un incident se produit.

⁸ <http://www.fehcom.de/qmail/spamcontrol.html>

⁹ <http://qmailwiki.inter7.com/Simscan/Guide>