

# **Installation de Exim-4 sous Fedora-6**

## **+ Dovecot avec format maildir**

## **+ Authentification BD mysql**

## **+ mailman**

Exim est un serveur de messagerie électronique (Mail Transfer Agent) utilisé sur de nombreux systèmes UNIX. Exim gagne de plus en plus du terrain sur ces concurrents sendmail et postfix, et même moi qui ne jurais que par sendmail jusqu'à très peu commence à m'y mettre petit à petit.

Je suis à mon xième serveur Exim installé et je pense que c'est le meilleur moment pour rédiger quelque chose sur ce sujet (d'ici là que j'oublie comment ça se passe, il vaut mieux vite écrire quelque chose dessus). J'espère donc que ce petit manuel pourra aider des gens (y compris moi-même) à installer ou réinstaller Exim sans trop de pétards.

Cette installation comprendra ces différentes parties :

- L'installation d'Exim à partir de sources, donc compilation de Exim et des applications nécessaires au bon fonctionnement d'Exim
- Construction de la base de données MySQL des utilisateurs de la messagerie
- Configuration de Dovecot avec support de MailDir + MySQL
- Installation et configuration de MailMan
- Configuration de Exim avec support de MailDir + MySQL + liste de diffusion MailMan
- Tests et mis en service.

## **I – Compilation de Exim :**

Le serveur que nous installerons supportera le format maildir, et délivrera les mails sous forme de fichiers séparés dans un répertoire, plutôt que d'avoir un seul fichier contenant tous les messages d'un utilisateur. Il trouvera ses utilisateurs dans une base de données mysql plutôt que dans les fichiers du système (passwd et shadow).

1.1 – Prérequis :

Antivirus (clamav) :

<http://www.clamav.net/>

Pas du tout compliqué à installer, faire *yum install clamav*.

Ou bien aller sur le site télécharger les sources puis compiler :

Supposons que vous ayez téléchargé le fichier clamav-0.88.6.tar.gz, faire :

```
$]# tar -zxf clamav-0.88.6.tar.gz
$]# /usr/sbin/useradd clamav -s /sbin/nologin -d /var/log/clamav
$]# cd clamav-0.88.6
$]# ./configure --sysconfdir=/etc/clamav
$]# make
$]# make install
```

Ensuite, éditez le fichier **/etc/clamav/clamd.conf** :

- Mettre un # devant la ligne où est écrit uniquement **Example** pour la commenter.
- Mettre un # devant la ligne commençant par **LocalSocket** pour la commenter
- Décommenter la ligne : **#TCP Socket 3310**
- Décommenter la ligne : **#TCPAddr 127.0.0.1**
- Décommenter la ligne : **#User clamav**
- Décommenter la ligne qui commence par **LogFile** en activant ceci :
  - o **LogFile /var/log/clamav/clamd.log**

A part ces options, il n'y à pas grand-chose à modifier dans ce fichier, vous pouvez tout de même le lire encore plus en détails et le personnaliser en changeant certaines options. Des exemples suivent :

```
# By default clamd uses scan options recommended by libclamav. This option
# disables recommended options and allows you to enable selected ones below.
# DO NOT TOUCH IT unless you know what you are doing.
# Default: disabled # Roger
DisableDefaultScanOptions
```

```
# PE stands for Portable Executable - it's an executable file format used
# in all 32-bit versions of Windows operating systems. This option allows
# ClamAV to perform a deeper analysis of executable files and it's also
# required for decompression of popular executable packers such as UPX, FSG,
# and Petite.
# Default: enabled # Roger
ScanPE
```

```
# Enable internal e-mail scanner.
# Default: enabled # Roger
ScanMail
```

**Mais** : Ne pas activer l'option **MailFollowURLs**

Mais ces derniers changements ne sont pas obligatoires ; j'ai bien fait tourner mon clamav pendant quelques mois sans problème avant d'apporter ces quelques petites modifs.

Editer le fichier **/etc/clamav/freshclam.conf** :

Mettre # devant le mot **Example** comme pour clamd.conf.  
 Et spécifier ce paramètre **UpdateLogFile /var/log/clamav/freshclam.log** en enlevant le # qui commence la ligne.  
 Décommenter la ligne **#DatabaseMirror** et remplacer XY par le country code (ml pour le Mali).

Pour lancer clamd, faire : \$]# /usr/local/sbin/clamd  
Pour lancer freshclam : \$]# /usr/local/bin/freshclam  
Pour lancer freshclam en daemon : \$]# /usr/local/bin/freshclam -d

Pour tester clamav :

\$]# /usr/local/bin/clamscan -r -l scan.txt /chemin/clamav-0.88.6.tar.gz

Et lire le fichier scan.txt.

\$]# /usr/local/bin/clamscan -r -l scan.txt /chemin/clamav-0.88.6.tar.gz

Et lire encore le fichier.

\$]# telnet 127.0.0.1 3310

Trying 127.0.0.1...

Connected to localhost.localdomain (127.0.0.1).

Escape character is '^J'.

**SCAN /home/roger/Install**

/home/roger/Install/clamav-0.88.6.tar.gz: ClamAV-Test-File FOUND

Connection closed by foreign host.

Pour terminer : éditer **/etc/rc.local** et rajouter les lignes **/usr/local/sbin/clamd** et **/usr/local/bin/freshclam -d** pour que ces processus soient lancés au redémarrage de la machine.

Anti-Spam (spamassassin) :

<http://www.spamassassin.org/> (downloader & perl Makefile.PL & make & make install)

Le principal fichier de config dans mon installation est **local.cf** sous **/etc/mail/spamassassin/**

Quelques options de configuration de spamassassin :

**required\_hits** ou **required\_score** : le score à partir duquel un message est classé comme spam. Par défaut, c'est 5, mais j'ai déjà vu des configurations avec 7.

**use\_bayes** et **bayes\_auto\_learn** : Mettre ces 2 paramètres à 1 pour dire à SA de faire attention aux mots ou séquences de mots souvent présents dans les spams et de les apprendre lui-même. La commande **sa-learn** permet de faire apprendre manuellement.

**trusted\_networks** permet de spécifier des réseaux auxquels on a confiance, donc des réseaux non spammeurs.

Il est également possible d'écrire des règles personnelles de filtrage. Quelques exemples :

Règle 1 : Cette règle permet d'augmenter le score d'un message dont le sujet est **Gegen das Vergessen** de 40 points. La description n'est pas obligatoire.

```
header LR SUBJECT_SOBER1    Subject =~ ^Gegen das Vergessen/  
describe LR SUBJECT_SOBER1  Subject: Worm Sober  
score LR SUBJECT_SOBER1    40
```

Règle 2 : S'applique au corps du message

```
body LOCAL_DEMONSTRATION_RULE    /Viagra XXL/  
score LOCAL_DEMONSTRATION_RULE  10
```

Règle 3 : S'applique aux URI se trouvant dans les messages avec un score à 30

```
uri LOCAL_URI_EXAMPLE    /www.example.com\OrderViagra//  
score LOCAL_URI_EXAMPLE  30
```

Des règles plus sophistiquées sont possibles et elles sont toutes à mettre dans le fichier local.cf. Vous pouvez aussi modifier les contenus des fichiers \*.pre (init.pre et v310.pre), mais les options par défaut marchent très bien.

Pour tester :

- Trouver les fichiers sample-nonspam.txt et sample-spam.txt. Dans mon install ils se trouvent dans /usr/share/doc/spamassassin/
- \$]# *spamassassin -t < sample-nonspam.txt > nonspam.out*  
\$]# *spamassassin -t < sample-spam.txt > spam.out*  
Lire les fichiers.

## 1.2 – Préparation à la compilation d’Exim :

Rendons-nous sur le site <http://www.exim.org/> et downloadons la dernière version de exim (exim-4.63.tar.gz)

```
$]# tar -zvxf exim-4.63.tar.gz  
$]# cd exim-4.63  
$]# cp src/EDITEME Local/Makefile  
$]# vi Local/Makefile
```

### Edition du fichier Makefile :

C'est ce fichier qui déterminera la manière dont Exim sera installé. Je mettrai juste les options que j'ai modifiées dans ma config :

```
#BIN_DIRECTORY=/usr/exim/bin # Roger  
BIN_DIRECTORY=/usr/exim-4.63/bin  
# Le repertoire des binaries de Exim  
  
#CONFIGURE_FILE=/usr/exim/configure # Roger  
CONFIGURE_FILE=/etc/exim-4.63/exim.conf  
# Le principal fichier de config de Exim  
  
#EXIM_USER= # Roger  
EXIM_USER=exim  
  
#EXIM_GROUP= # Roger  
EXIM_GROUP=exim  
  
#SPOOL_DIRECTORY=/var/spool/exim # Roger  
SPOOL_DIRECTORY=/var/spool/exim-4.63  
  
# SUPPORT_MAILDIR=yes # Roger  
SUPPORT_MAILDIR=yes  
  
# LOOKUP_MYSQL=yes # Roger  
LOOKUP_DSEARCH=yes  
LOOKUP_MYSQL=yes  
  
#-----  
# Additional libraries and include directories may be required for some  
# lookup styles (e.g. LDAP, MySQL or PGSQL). LOOKUP_LIBS is included only
```

```

# the command for linking Exim itself, not on any auxiliary programs. You
# don't need to set LOOKUP_INCLUDE if the relevant directories are already
# specified in INCLUDE. The settings below are just examples; -lpq is for
# PostgreSQL, -lgds is for Interbase, -lsqlite3 is for SQLite.

# LOOKUP_INCLUDE=-I /usr/local/ldap/include -I /usr/local/mysql/include -I
/usr/local/pgsql/include # Roger
LOOKUP_INCLUDE=-I /usr/include/mysql

# LOOKUP_LIBS=-L/usr/local/lib -lldap -llber -lmysqlclient -lpq -lgds -
lsqlite3 # Roger
LOOKUP_LIBS=-L/usr/lib/mysql -lmysqlclient

# WITH_CONTENT_SCAN=yes # Roger
WITH_CONTENT_SCAN=yes

# WITH_OLD_DEMIME=yes # Roger
WITH_OLD_DEMIME=yes

# LOG_FILE_PATH=/var/log/exim_%slog # Roger
LOG_FILE_PATH=/var/log/exim-4.63/%s.log

```

#### Exim Monitor :

(Pour ceux qui choisissent de ne pas compiler Exim monitor, éditez le fichier Local/Makefile, et mettre un # devant la ligne : EXIM\_MONITOR=eximon.bin et sauter à l'étape suivante)  
Dans le fichier Local/Makefile, ne pas mettre en commentaire cette ligne  
EXIM\_MONITOR=eximon.bin

```
$]# cp exim_monitor/EDITME Local/eximon.conf
$]# vi Local/eximon.conf
```

Puis modifier les options qui vous intéressent (c'est très intuitif).

Mais pour compiler exim monitor, il faudrait que les outils de « développement du logiciel X » soient installés.

#### SA-Exim :

Téléchargez la dernière version de sa-exim sur <http://marc.merlins.org/linux/exim/sa.html>.

Exemple : <http://marc.merlins.org/linux/exim/files/sa-exim-4.2.1.tar.gz>

```
$]# tar -zvxf sa-exim-4.2.1.tar.gz
$]# cd sa-exim-4.2.1
```

Editez le fichier sa-exim.c qui comporte une erreur et assignez le bon fichier de config de sa-exim. Dans ma config, j'ai dû modifier une ligne pour la rendre ainsi :

```
#define SPAMASSASSIN_CONF      "/etc/exim-4.63/sa-exim.conf"
```

```
$]# cp sa-exim.c ..//exim-4.63/src/local_scan.c
$]# make sa-exim.h
$]# cp sa-exim.h ..//exim-4.63/src/
```

Après l'installation de exim, il va falloir copier le fichier **sa-exim.conf** dans le répertoire spécifié dans le fichier sa-exim.c. Dans cette installation, c'est /etc/exim-4.63/ qui est aussi le répertoire où se trouvera le principal fichier de configuration de exim.

### 1.3 – Installation d’Exim :

```
Création de l’utilisateur exim : useradd exim -s /sbin/nologin -d /var/spool/exim-4.63
$]# cd <chemin>/exim-4.63/
$]# make
$]# make install
Grâce aux options choisies dans mon Makefile, les binaires de exim sont dans /usr/exim-4.63/bin/, les logs sont dans /var/log/exim-4.63/, et le principal fichier de configurations (exim.conf) dans /etc/exim-4.63/.
$]# cp <chemin>/sa-exim-4.2.1/sa-exim.conf /etc/exim-4.63/
```

### 1.4 – Quelques petits réglages de sa-exim :

Ouvrir **sa-exim.conf**, et modifier cette variable : **SAEximRunCond: 0** remplacez le 0 par 1 → **SAEximRunCond: 1**  
Remplacez aussi tout ce qui est **/var/spool/exim** par **/var/spool/exim-4.63** (Répertoire spécifié dans le Local/Makefile)

## II – La base de données (Mysql) :

Je suppose ici que le serveur de base de données mysql et toutes les librairies nécessaires sont installés. Ce qui nous reste à faire est juste la création de la base de données des utilisateurs et quelques petits réglages.

Création de la base de données :

Celle-ci s’appellera **users** et le nom de table contenant les utilisateurs sera **user**.  
Les champs de cette table seront userid, password, name, firstname, home, uid, gid, et quota

```
$]# mysql -u root -p
mysql> create database users; // Création de la base de données Exemple
mysql> use users;
mysql> create table user (
    userid VARCHAR(32) NOT NULL,
    password VARCHAR(64) NOT NULL,
    name VARCHAR(32) NOT NULL,
    firstname VARCHAR(32) NOT NULL,
    home VARCHAR(255) NOT NULL,
    uid INTEGER NOT NULL,
    gid INTEGER NOT NULL,
    quota INTEGER NOT NULL
);
```

Création de l’utilisateur de base de données qui a les droits sur cette base de données :

```
mysql> grant
      select, insert, update, delete
      on users.*
      to mailuseradmin@localhost
      identified by 'mailadminpass';
```

En résumé :

Nous avons un serveur de base de données **localhost**, nous avons créé une table **user** dans la base de données **users**, et l'utilisateur qui administre cette base est **mailuseradmin** avec son mot de passe **mailadminpass**.

Ajout d'un utilisateur de test dans la base de données des utilisateurs de la messagerie :

```
mysql> insert into user values (
    'roger',
    'passwordcrypté',
    'Yerbang',
    'Roger',
    '/home-mails/roger',
    '42', '42', '500');
```

42 étant l'uid de exim sur mon système et 500 représentant 500Mo par exemple.

### III – Serveur IMAP+POP3 (Dovecot) :

Nous allons configurer Dovecot pour qu'il sache retrouver les utilisateurs et leurs mots de passe dans la base de données users que nous venons de créer, et lire les mails sous format maildir.

Les versions antérieures à la 1.0 de dovecot ne supportent pas l'authentification par base de données mysql (enfin, c'est ce que je crois avoir constaté).

Le fichier de config principal est **/etc/dovecot.conf**. Voici ce que je fais :

```
$]# cd /etc
$]# mkdir dovecot
$]# cd dovecot
$]# ln -s ../dovecot.conf .
```

Ensute, repérez le fichier dovecot-sql.conf. Dans mon cas, il se trouve dans :  
/usr/share/doc/dovecot-1.0/examples. Puis copier ce fichier dans /etc/dovecot

```
$]# cp /usr/share/doc/dovecot-1.0/examples/dovecot-sql.conf dovecot-mysql.conf
```

Ainsi donc, tous les fichiers de config de dovecot se trouvent dans /etc/dovecot.

Editez le fichier de config principal dovecot.conf :

```
# Enable les protocoles imap et pop3
protocols = imap pop3
# Faire du maildir et les mails seront dans /home-mails/<utilisateur>/mail
# A spécifier aussi dans exim
default_mail_env = maildir:/home-mails/%u/mail

# Dans la section Authentication proccesses
Mettre en commentaire toute la section commençant par passdb pam {
Mettre en commentaire toute la section commençant par userdb passwd {
```

Rajouter ces deux sous-sections, toujours dans la section Authentication processes :

```

# SQL database
passdb sql {
    # Path for SQL configuration file, see doc/dovecot-sql.conf for example
    args = /etc/dovecot/dovecot-mysql.conf
}

# SQL database
userdb sql {
    args = /etc/dovecot/dovecot-mysql.conf
}

```

L'argument est le fichier qu'on a déjà copié dans **/etc/dovecot**.

Modification du fichier **/etc/dovecot/dovecot-mysql.conf** :

```

# Database driver: mysql, pgsql, sqlite, pour nous, c'est mysql
driver = mysql
# Les paramètres de connexion à la base de données
connect = host=localhost dbname=users user=mailuseradmin password=mailadminpass
# Le schéma de cryptage, je choisis CRYPT pour faire du DES.
default_pass_scheme = CRYPT
# Query to retrieve the user information.
password_query = SELECT userid as user, password, home as userdb_home, uid
as userdb_uid, gid as userdb_gid FROM user WHERE userid = '%u'

```

Voilà, config de dovecot terminé. Lancer dovecot, et voir le maillog. Pour tester que IMAP et POP3 fonctionne bien, des détails sont sur :

<http://www.yerbynet.com/Cours/POP3IMAPCommands.html>

#### IV – MailMan (Pour les listes de diffusion) :

Installer la version rpm.

Editer le fichier **/etc/mailman/mm\_cfg.py**

Modifier une seule option :

```
DEFAULT_EMAIL_HOST = 'votredomaine'
```

Lancer la commande **/usr/lib/mailman/bin/newlist** et suivre les instructions pour créer la liste mailman.

Et enfin, lancer mailman : **/etc/init.d/mailman start**

L'URL pour accéder au module d'administration par interface web est :

<http://<nomdevotreserveur>/mailman/admin>

#### V – Configuration de Exim :

**/etc/exim-4.63/exim.conf** :

Communiquer les paramètres de base de données à exim en ajoutant cette ligne dans la section principale de exim.conf :

```
# Parametres d'accès à la base de données users
hide mysql_servers = localhost/users/mailuseradmin/mailadminpass
# Serveur = localhost
# Base de données = users
# User = mailuseradmin
# Mot de passe = mailadminpass
```

Pour ceux qui utilisent les utilisateurs du système (/etc/passwd), cette étape (précédente) n'est pas nécessaire.

On suppose que le domaine sur lequel va tourner exim est yerbynet.com et que nous avons un autre domaine secondaire yerbynet.org. Les relays seront permis pour les domaines yerbytest.org et yerbytest.net. D'où les config suivantes :

# Paramètres pour Mailman dans la section principale

```
MAILMAN_HOME=/usr/lib/mailman
MAILMAN_VAR=/var/lib/mailman
MAILMAN_WRAP=MAILMAN_HOME/mail/mailman
MAILMAN_USER=mailman
MAILMAN_GROUP=mailman
domainlist mm_domains=yerbynet.com
```

```
primary_hostname = mail.yerbynet.com # Nom du serveur
qualify_domain = yerbynet.com
```

```
# Le fichier localdomains contient la liste des domaines locaux et virtuels
# Le fichier localdomains.db est généré par exim_dbmbuild en se basant sur
# le fichier localdomains
domainlist local_domains = dbm:/etc/exim-4.63/localdomains.db
# Les domaines à relayer sont dans relay_domains
# relay_domains.db est généré à partir de relay_domains
domainlist relay_to_domains = dbm:/etc/exim-4.63/relay_domains.db
# Fichier texte accept_relay pour les réseaux autorisés
hostlist relay_from_hosts = /etc/exim-4.63/accept_relay
```

ACL pour le contrôle des mails entrant :

```
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_data
```

Activation de l'antivirus :

```
av_scanner = clamd:127.0.0.1 3310
```

Activation de l'antispam :

```
spamd_address = 127.0.0.1 783
```

Ne jamais envoyer de message au root :

```
never_users = root
```

Détails des ACL :

acl\_check\_rcpt non modifié.

Modifications de l'acl\_check\_data :

```
acl_check_data:
```

```

# Deny if the message contains a virus. Before enabling this check, you
# must install a virus scanner and set the av_scanner option above.
#
# Roger a décommenté
deny    malware   =
        message = This message contains a virus ($malware_name).

# Roger a rajouté
deny      message = This message contains malformed MIME
{$demime_reason}.
        demime   =
        condition = ${if > {$demime_errorlevel}{2}{1}{0}}
#
deny message = Found blacklisted file attachment
        demime = vbs:com:bat:pif:prf:lnk

# Accept the message.

accept

```

Pour ceux qui veulent que Exim recherchent les utilisateurs dans une base de données mysql, continuer, sinon, s'arrêter :

#### *Dans la section des routers :*

Rajoutez ce router pour mailman avant tous les autres routers :

```

# Router pour Mailman
mailman_router:
    driver = accept
    domains = +mm_domains
    require_files = MAILMAN_VAR/lists/$local_part/config.pck
    local_part_suffix_optional
    local_part_suffix = -bounces : -bounces** : \
                      -confirm* : -join : -leave : \
                      -owner : -request : -admin
    transport = mailman_transport

```

Pour gérer des domaines virtuels, rajoutez ce routeur avant le router dnslookup :

```

# Le répertoire /etc/exim-4.63/virtualdomains doit exister et contenir les domaines virtuels
### Pour les domaines virtuels
virtual:
    driver = redirect
    domains = dsearch;/etc/exim-4.63/virtualdomains
    allow_fail
    data = ${lookup{$local_part@$domain}lsearch*{/etc/exim-
4.63/virtualdomains/$domain}}

```

Rajoutez ce router avant le router localuser :

```

# Recherche de l'utilisateur dans la base de données mysql
mysql_localuser:
    driver = redirect
    allow_fail
    allow_defer
    data = ${lookup mysql{select userid from user where
userid='$local_part'}{$value}fail}
    redirect_router = localuser
    no_more

```

La base de données users contient une table user qui a un champ userid, et c'est dans ce champ que exim recherche les utilisateurs locaux.

Modifiez le router localuser :

```
localuser:  
    driver = accept  
#  check_local_user  # Roger  
    transport = local_delivery  
    cannot_route_message = Unknown user
```

*Dans la section des transports :*

Rajoutez le transport pour mailman :

```
# Transport pour Mailman  
mailman_transport:  
    driver = pipe  
    command = MAILMAN_WRAP \  
        '$if def:local_part_suffix \  
            ${sg{$local_part_suffix}{-(\\w+)(\\.+.*?){$1}}}\ \  
            {post}' \  
        $local_part  
    current_directory = MAILMAN_VAR  
    home_directory = MAILMAN_VAR  
    user = MAILMAN_USER  
    group = MAILMAN_GROUP
```

Commentez le local\_delivery ou supprimé le :

```
#Roger a commenté la section suivante  
#local_delivery:  
#  driver = appendfile  
#  file = /var/mail/$local_part  
#  delivery_date_add  
#  envelope_to_add  
#  return_path_add  
#  group = mail  
#  mode = 0660
```

Rajoutez un nouveau local\_delivery

```
#Roger a ajouté la section suivante  
local_delivery:  
    driver = appendfile  
    user = exim  
    group = exim  
    delivery_date_add = true  
    envelope_to_add = true  
    return_path_add = true  
    directory = /home-mails/$local_part/mail  
    quota = 250M  
    maildir_format = true
```

```
*****FIN de la config *****
```

Evidemment, des config plus sophistiquées sont encore possibles.

Derniers réglages :

```
$]# mkdir /home-mails
```

```
$]# chown -R exim:exim /home-mails
$]# mkdir /var/log/exim-4.63/
$]# chown -R exim:exim /var/log/exim-4.63/
Remplacement de sendmail par exim (pas du tout obligatoire)
$]# cd /etc/alternatives
$]# rm mta mta-mailq
$]# ln -s /usr/exim-4.63/bin/exim mta
$]# ln -s /usr/exim-4.63/bin/exim mta-mailq
```

## VI – Tests et lancement de Exim :

```
$]# export PATH=/usr/exim-4.63/bin/:$PATH
Test pour savoir si le fichier de configuration exim.conf ne comporte pas d'erreurs
$]# exim -bV
```

Vérification de ce que exim fera avec un utilisateur local :

```
$]# exim -bt localuser
```

Essaie de livraison locale réelle :

```
$]# exim -v -odf localuser
```

```
test
```

```
.
```

```
LOG: MAIN
```

```
<= root@yerbynet.com U=root P=local S=324
delivering 1F13yK-00087F-Qt
```

```
LOG: MAIN
```

```
=> localuser <localuser@yerbynet.com> R=localuser T=local_delivery
```

```
LOG: MAIN
```

```
Completed
```

Vérifier que localuser a bien reçu un message :

```
$]# cd /home-mails/localuser/mail/
```

Il doit contenir les répertoires **cur**, **new** et **tmp**.

Essaie de livraison distante réelle :

```
$]# exim -d -odf user@domain
```

Lancement de Exim :

```
$]# /etc/init.d/spamassassin restart
```

```
$]# /etc/init.d/clamd restart ou bien /usr/local/sbin/clamd
```

```
$]# /etc/init.d/mysql restart
```

```
$]# exim -bd -q20m
```

Vérification du relayage à partir d'autres machines dans le réseau autorisé : 196.203.50.0/26, et à partir de machines dans d'autres réseaux.

```
$]# exim -bh 196.2.6.1
```

Lancement de Exim Monitor :  
Sous environnement X-Widows, lancez la commande ***eximon***.

### **Bonus : Ajout de routeur et transport pour vacation**

A/ Exemple avec texte de vacance dans une base de données

```
# Routeur uservacation transport à rajouter avant le routeur par défaut userforward
uservacation:
    driver = accept
    condition = ${lookup mysql{select reply from user where
userid='$local_part'}{$value}fail}
    #require_files = /home-mails/$local_part/.vacation.msg
    no_verify
    transport = vacation_transport
    unseen

# Transport de vacation
vacation_transport:
    driver          = autoreply
    text            = ${lookup mysql{select replytext from user
where userid='$local_part'}{$value}fail}
    once           = /home-mails/${local_part}/.vacation.db
    from           = "$local_part <${local_part}@${domain}>"
    to             = $sender_address
    subject        = "[AutoReply] Re: \$h_subject:"
```

Pour mettre en place ces routeur et transport, on a dû modifier la structure de la table **user** de la base de données **users**, pour rajouter 2 champs : **reply** qui est un booleen, et **replytext** qui est un champ texte. reply ne doit contenir que YES, NO ou être vide. Il permet de déterminer si le user est en vacance (YES) ou s'il ne l'est pas. Dans le routeur **uservacation**, on teste ce champ, s'il est à YES, on appelle le transport **vacation\_transport** qui utilise le champ replytext qui contient le texte de réponse automatique à renvoyer.

Requêtes SQL de modification de la table user :

```
mysql> ALTER TABLE user ADD reply VARCHAR(3);
mysql> ALTER TABLE user ADD replytext TEXT;
```

B/ Exemple avec texte de vacance dans un fichier

```
# Routeur uservacation transport à rajouter avant le routeur par défaut userforward
uservacation:
    driver = accept
    require_files = /home-mails/$local_part/.vacation.msg
    no_verify
    transport = vacation_transport
    unseen
```

```

# Transport de vacation
vacation_transport:
  driver          = autoreply
  file           = /home-mails/${local_part}/.vacation.msg
  file_expand
  once           = /etc/exim-4.63/${local_part}.vacation.db
  from           = "$local_part <${local_part}@${domain}>"
  to             = $sender_address
  subject        = "[AutoReply] Re: \$h_subject:"

```

Pour cet exemple, il suffit juste de créer le fichier .vacation.msg contenant le texte d'auto-reply pour l'activer, et le supprimer pour le désactiver.

### Sources :

<http://www.exim.org/>  
<http://marc.merlins.org/linux/exim/sa.html>  
[http://dman13.dyndns.org/~dman/config\\_docs/exim-spamassassin/](http://dman13.dyndns.org/~dman/config_docs/exim-spamassassin/)  
<http://www.timj.co.uk/linux/Exim-SpamAndVirusScanning.pdf>  
<http://www.flatmtn.com/computer/Linux-Exim4.html>  
<http://www.clamav.net>  
<http://wiki.apache.org/spamassassin/>

© Janvier 2007  
 Roger YERBANGA  
[www.rogeryerbanga.fr.st](http://www.rogeryerbanga.fr.st)