Envoyer et recevoir du courrier pour un réseau multi-utilisateurs

Olivier Hoarau (olivier.hoarau@funix.org)

V3.71 du 8 janvier 2017

1 Historique du document

V3.71 08.01.17 envoi et réception sur internet avec chiffrement SSL/TLS

V3.7 05.01.17 renouvellement de certificat pour dovecot et mise en place connexion chiffrée SSL/TLS avec sendmail

V3.6 24.12.16 modif dans la config de sendmail pour supprimer des erreurs dans les logs et passage à clamav 0.99.2

V3.5 15.10.15 modifications suite passage à Mageia5 et systemd et à spamassassin 3.4.1, DCC 1.3.158, pyzor 0.7.0, spamass-milter 0.4.0 et clamAV 0.98.7

V3.4 4.9.10 suppression de l'outil obsolète mailfilter, passage à SpamAssassin 3.3.1, DCC 1.3.130, clamav 0.96.2, rajout de technique de prises en compte des spams, petite modif dans la config de dovecot

V3.3 26.12.09 passage à SpamAssassin 3.2.5 passage à DCC 1.3.116 et clamav 0.95.3

V3.2 24.08.07 passage à SpamAssassin 3.2.3, razor 2.84, DCC 1.3.58 et ClamAV 0.91.1

V3.1 08.03.07 passage à passage à mailfilter 0.8, SpamAssassin 3.1.8, dcc 1.3.53 et ClamAV 0.90.1

V3.0 25.07.06 passage à SpamAssassin 3.1.8, razor 2.82, DCC 1.3.39, spamass milter 0.3.1 et clamav 0.88.3, adaptation pour installation sous (k)ubuntu, installation d'un serveur POP3 avec dovecot sous (k)ubuntu

V2.9 29.01.06 passage à spamassassin 3.1.0, razor-agent-2.77 et ClamAntiVirus 0.88

V2.81 10.08.05 correction d'une erreur pour le script d'apprentissage des spams pour sa-learn

V2.8 06.08.05 sendmail, un mot sur l'installation avec les packages de la LE2005. passage à spamassassin 3.0.4, razor 2.75, DCC 1.3.12 et clamav 0.86.2

V2.7 06.05.05 passage à SpamAssassin 3.0.3, DCC 1.34, SpamAssassin Milter 0.3.0 et Clam Anti Virus 0.84

V2.6 07.01.05 passage à spamassassin 3.0.2, razor 2.67 et DCC 1.2.66, rajout de la configuration du fichier freshclam.conf pour clamav

V2.5 06.11.04 passage à DCC 1.2.58, SpamAssassin 3.0.1 et Clamav 0.80, modification dans la configuration de clamav et dans l'appel de sa-learn.

V2.4 02.10.04 passage à mailfilter 0.6.2, SpamAssassin 3.0.0, razor 2.61, DCC 1.2.54 et clamav 0.75.1, modifications dans la configuration de spamassassin (fichier de lancement, droits à fixer)

V2.3 31.05.04 Passage à DCC 1.2.49 et clamav 0.71, un mot pour indiquer à spamassassin de classer ou non des mails comme spam

V2.2 18.04.04 Passage à Razor 2.40, DCC 1.2.39 et clamav 0.70

V2.1 27.03.04 Rajout d'un paragraphe sur la lutte anti virus et anti spam, changement de version des différents softs

V2.0 04.05.03 Passage à Mandrake 9.1, légères modifications

V1.9 24.12.02 Passage à Mandrake, changement de version de sendmail

V1.8 07.07.02 Passage à mailfilter 0.4.0

V1.7 09.06.02 Passage à Mandrake 8.2, modifs
- version de sendmail et package
- rajout de confCF_VERSION dans le fichier de config pour rajouter un commentaire dans l'entête de mail
- mise à jour de copier/coller d'entêtes de mail

V1.6 16.12.01 - Passage à Mandrake 8.1, modifs version de sendmail et package
- Rajout de la partie réception du courrier

V1.5 17.06.01 - Rajout de quelques notes et avertissements dans le chapitre configuration de sendmail
- Rajout d'un paragraphe sur la sécurisation de sendmail (commandes vfry et expn, modifier l'invite de sendmail).
- Passage à Mandrake 8.0, changement de numéro de version et du chemin du fichier de conf

V1.4 18.03.01 Rajout d'une remarque pour que les mails envoyés sur le réseau local ne partent pas d'abord chez le FAI.

V1.3 03.12.00 Passage à Mandrake 7.2, modifs :
- version sendmail
- fichiers logs sendmail (rajout d'un paragraphe)
- sendmail-cf sur le CD 2
- genericstable.db qui se trouve maintenant sous /etc/mail
- Modification du script ip-up qui reconfigure sendmail suivant leFAI utilisé pour prendre en compte le changement d'emplacement de genericstable.

2 Préambule

Ce document a pour but de présenter une configuration de **sendmail** sur un poste Linux connecté de façon intermittente ou permanente à internet et qui cache un réseau privé « masqueradisé ». Il a pour but de présenter également la manière de récupérer les mails et de filtrer les spams et les virus.
La dernière version de ce document est téléchargeable à l'URL http://www.funix.org.

3 Recevoir du courrier et configurer un serveur pop

3.1 Présentation de la configuration

Cette paragraphe a pour objet de vous présenter comment récupérer les mails des utilisateurs de votre réseau et les mettre à leur disposition. Le rôle des différents outils présentés dans cette page est le suivant:
- **fetchmail** permet de récupérer les mails des utilisateurs de votre réseau sur plusieurs serveurs **POP**
- **procmail** permet de faire le tri des mails et de dispatcher suivant le destinataire
- le serveur **POP** pemet de rendre accessible aux lecteurs de mails de votre réseau les mails qui sont arrivés, que vos clients soient sous windows ou sous unix.

Niveau architecture, le serveur **pop** et les services de récupération de mail **fetchmail/procmail** tournent sur un serveur de votre réseau domestique, c'est sur même machine qu'on concentrera la récupération des mails des utilisateurs du réseau issus des différents serveurs de mail qu'on peut trouver sur le net accessibles via **IMAP** ou **POP** (version sécurisée ou non). Sur les postes clients il n'y a rien à installer de particulier, mise à part un client de messagerie classique comme **thunderbird** dans lequel on configurera le serveur **pop** local pour récupérer les mails et le serveur sendmail local pour l'émission.

3.2 Configuration de fetchmail

3.2.1 Méthode automatique

Vous avez la possibilité de lui indiquer d'aller récupérer toutes les heures en rajoutant le script **recupmail** dans le répertoire **/etc/cron.hourly/** voilà son contenu

/usr/bin/fetchmail -a -f /root/.fetchmailrc -L /var/log/fetchmail.log
echo "" >> /var/log/fetchmail.log

ATTENTION les commandes lancés dans ce fichier doivent être indiquées avec leur chemin complet. Le fichier **/var/log/fetchmail.log** doit d'abord être créé avec un **touch**.

Maintenant root doit créer un fichier **.fetchmailrc** qui doit se trouver dans sa home directory avec les droits 600 (**chmod 600 ~/.fetchmailrc**). Ce fichier contient les lignes suivantes:

set daemon 600
set logfile /var/log/fetchmail.log
poll pop.fai.fr protocol pop3
user login-fai there with password password-fai is olivier here

poll pop.fnac.net protocol pop3
user login-fnac there with password password-fnac is olivier here

poll pop.free.fr protocol pop3
user login-free there with password password-free is olivier here

poll pop.ifrance.com protocol pop3
user login-ifrance there with password password-ifrance is olivier here

poll pop.fnac.net protocol pop3
user login2-fnac there with password password2-fnac is veronique here

poll pop.ifrance.com protocol pop3
user login2-ifrance there with password password2-ifrance is veronique here

poll pop.libertysurf.fr protocol pop3
user login-liberty there with password password-liberty is olivier here

Le paramètre 600 fixe la période de relevé de la boîte aux lettres, l'unité étant la seconde. Le fichier **fetchmail** sous **/var/log** est le fichier de log, ***pop.fai.fr*** est le nom du serveur pop de votre provider. ***login-fai*** est le nom de votre login chez votre provider, ***password-fai*** est le mot de passe chez le provider, **olivier** est le login de l'utilisateur local correspondant. Vous rajoutez autant de ligne poll et user que vous avez de compte pop à droite

et à gauche, vous noterez qu'on peut en profiter pour relever les emails d'autres utilisateurs de votre réseau (dans l'exemple utilisateur du réseau privé veronique).

ATTENTION: les mots de passe sont marqués en clair (d'où les droits du fichier...).

Par défaut **fetchmail** se repose ensuite sur le serveur **smtp** qui tourne sur la machine sur le port 25 pour délivrer les mails sous **/var/spool/mail**. Dans le cas où vous utilisez un serveur **SMTP** avec une connexion chiffrée **TLS/SSL** sur le port 465 **fetchmail** ne saura pas délivrer les mails. Dans ce cas il faudra se servir de **procmail** comme MDA (mail delivery agent) et faire la modification suivante dans le fichier **.fetchmailrc**

```
poll pop.libertysurf.fr protocol pop3
 user login-liberty there with password password-liberty is olivier here
 mda "mda "/usr/bin/procmail -d %T"
```

maintenant si votre FAI a mis en place un serveur **pop** avec connexion sécurisée avec **TLS/SSL** on va rajouter l'option qui va bien comme ceci

```
poll pop.libertysurf.fr protocol pop3
 user login-liberty there with password password-liberty is olivier here
 option ssl;
 mda "mda "/usr/bin/procmail -d %T"
```

il faudra rajouter ensuite l'option --**sslcertck** à la commande **fetchmail** comme ceci

```
/usr/bin/fetchmail --sslcertck -a -f /root/.fetchmailrc -L /var/log/fetchmail.log
```

3.2.2 Méthode manuelle

Vous avez aussi la possibilité en tant que simple utilisateur de créer votre propre fichier **.fetchmailrc** (syntaxe idem plus haut) que vous placerez dans votre homedirectory et de lancer **fetchmail** d'un shell. Vous pouvez très bien aussi récupérer les mails des autres utilisateurs de votre réseau.

3.3 Limiter la taille des mails récupérés et supprimer les mails sur le serveur

NOTE Si vous voulez limiter la taille des fichiers récupérés à 100Ko par exemple, vous avez l'option:

```
fetchmail -l 100000
```

Ca va laisser tous les fichiers dont la taille est supérieure à 100Ko sur le serveur **pop** du fai, pour visualiser l'header et les supprimer.

```
telnet pop.fai.fr 110
Trying 195.154.205.225...
Connected to pop.fai.fr
Escape character is '^]'.
+OK POP3 mailhub.fai.fr v7.64 server ready
user login-pop
+OK User name accepted, password please
pass password-pop
+OK Mailbox open, 4 messages
list
+OK Mailbox scan listing follows
1 2199201
2 132664
3 388987
4 310757
```

Vous pouvez voir que vous avez 4 messages ainsi que leur taille. Pour visualiser l'header du message 1:

```
top 1   0
```

Vous pouvez visualiser le corps du message mais je ne le vous conseille pas, si c'est une image de 1Mo, ça va bloquer votre shell un certain temps. Je vous donne quand même la commande pour le message 1:

retr 1

Pour supprimer le message 1:

dele 1

Et enfin pour quitter:

quit

3.4 Configuration de procmail

Dans le cas où vous avez un compte pop unique avec plusieurs emails rattachés, **fetchmail** va tout mettre dans la boîte aux lettres de celui qui va lancer la commande **fetchmail**, pour effectuer un tri à la réception, vous devez penser à **procmail**.
Procmail permet de trier le courrier reçu par **fetchmail**. pour cela tout utilisateur avec son **.fetchmailrc** doit avoir un **.procmailrc** également dans sa home directory. Si je prends mon exemple, je disposais d'un compte pop unique chez mon provider fnac.net, mon adresse email était olivier.hoarau@fnac.net, mon compte local est olivier, celle de ma tendre et chère veronique.hoarau@fnac.net et compte local veronique. Si je veux expédier à Véronique tous les courriers dont les champs Destinataire (To) ou Copie (Cc) contiennent le champ veronique ou Véronique ou encore Veronique, voici la tête de mon **.procmailrc**

```
#olivier
:0 c
*^(To|Cc|Bcc):*(veronique|Veronique)
!veronique
```

Celui de ma femme aura cette tête là:

```
#veronique
:0 c
*^(To|Cc|Bcc):*(olivier|Olivier|funboard|Funboard)
!olivier
```

Je suis abonné à une liste funboard, c'est le nom de la liste qui apparaît dans la liste du destinataire ou du destinataire en copie, et non pas mon nom, d'où le critère de tri.

Le ! réexpédie localement le courrier vers le bon destinataire. Vous pouvez très bien aussi faire un fichier unique pour chaque utilisateur qui aura cette tête là:

```
#redirection vers veronique
:0 c
*^(To|Cc|Bcc):*(veronique|Veronique)
!veronique

#redirection vers olivier
:0 c
*^(To|Cc|Bcc):*(olivier|Olivier|funboard|Funboard)
!olivier

# les autres mails au destinataire non identifié vont vers olivier, vous pouvez très bien mettre /dev/null (poubelle) à la place de !olivier
:0
*.*
!olivier
```

A noter que le petit **c** permet de pouvoir gérer les copies, en son absence si un mail arrive avec pour destinataire (To) Véronique et Olivier en copie (Cc), ce n'est que le premier dans la liste qui recevra le mail (en l'occurence Véronique dans mon exemple de fichier),**c** permet qu'olivier reçoive aussi le courrier.
Le courrier échoue sous **/var/spool/mail** dans un fichier qui a pour nom le login de l'utilisateur.

Si vous disposez d'un email unique avec un seul email rattaché et que vous comptez vous en servir pour plusieurs personnes. Vous pouvez demander à vos interlocuteurs de préciser dans le sujet du mail le destinataire et faire un tri similaire à celui vu précédemment en filtrant sur le champ Subject du mail (***^(Subject):*(veronique|Veronique)**).

Voilà un filtre intéressant trouvé à l'adresse suivante http://www.linuxfocus.org/Francais/January2003/article279.shtml. Il permet d'avertir automatiquement l'expéditeur qui vous a envoyé un fichier word.

```
# Promail script to
# reject word documents. Reject the mail, but do not reply to
# error messages "From MAILER-DAEMON"
# If you use ":0 Bc" instead of ":0 B" then you will still get the mail
:0 H
* !^From.*DAEMON
{
# The mime messages with word documents look like this in the body
# of the message:
#------=_NextPart_000_000C_01C291BE.83569AE0
#Content-Type: application/msword;
# name="some file.doc"
#Content-Transfer-Encoding: base64
#Content-Disposition: attachment;
# filename="real file.doc"
:0 B
* ^Content-Type:.*msword
| (formail -r ; cat /home/olivier/reject-text-msword ) | $SENDMAIL -t
}

# par défaut les autres mails sont envoyés à olivier
:0:
!olivier
```

Le fichier **/home/olivier/reject-text-msword** contient un texte décrivant les raisons pour lesquelles vous ne voulez pas recevoir de fichier word et préférez d'autres formats. A noter que ce script est adaptable pour des réponses automatiques en fonction de certains critères.

3.5 Configuration du serveur pop

Vous devez installer le package **dovecot**, voilà le résultat

```
Generating a 2048 bit RSA private key
...................+++
.........................................+++
writing new private key to '/etc/pki/tls/private/dovecot.pem'
-----
```

le message précédent correspond à la création des clés pour les connexions SSL. Le package **dovecot** contient le serveur **imap** et **pop3**, pour ma part j'utilise ce dernier**.** Voilà mon fichier **/etc/dovecot/dovecot.conf** en configurant un serveur **pop3**

```
## Dovecot configuration file

# If you're in a hurry, see http://wiki2.dovecot.org/QuickConfiguration

# "doveconf -n" command gives a clean output of the changed settings. Use it
# instead of copy&pasting files when posting to the Dovecot mailing list.

# '#' character and everything after it is treated as comments. Extra spaces
# and tabs are ignored. If you want to use either of these explicitly, put the
# value inside quotes, eg.: key = "# char and trailing whitespace  "

# Most (but not all) settings can be overridden by different protocols and/or
# source/destination IPs by placing the settings inside sections, for example:
```

```
# protocol imap { }, local 127.0.0.1 { }, remote 10.0.0.0/8 { }

# Default values are shown for each setting, it's not required to uncomment
# those. These are exceptions to this though: No sections (e.g. namespace {})
# or plugin settings are added by default, they're listed only as examples.
# Paths are also just examples with the real defaults being based on configure
# options. The paths listed here are for configure --prefix=/usr
# --sysconfdir=/etc --localstatedir=/var

# Protocols we want to be serving.
#protocols = imap pop3 lmtp
protocols = pop3

# A comma separated list of IPs or hosts where to listen in for connections.
# "*" listens in all IPv4 interfaces, "::" listens in all IPv6 interfaces.
# If you want to specify non-default ports or anything more complex,
# edit conf.d/master.conf.
#listen = *, ::
listen = *

# Base directory where to store runtime data.
#base_dir = /var/run/dovecot/

# Name of this instance. In multi-instance setup doveadm and other commands
# can use -i <instance_name> to select which instance is used (an alternative
# to -c <config_path>). The instance name is also added to Dovecot processes
# in ps output.
#instance_name = dovecot

# Greeting message for clients.
#login_greeting = Dovecot ready.

# Space separated list of trusted network ranges. Connections from these
# IPs are allowed to override their IP addresses and ports (for logging and
# for authentication checks). disable_plaintext_auth is also ignored for
# these networks. Typically you'd specify your IMAP proxy servers here.
#login_trusted_networks =

# Space separated list of login access check sockets (e.g. tcpwrap)
#login_access_sockets =

# With proxy_maybe=yes if proxy destination matches any of these IPs, don't do
# proxying. This isn't necessary normally, but may be useful if the destination
# IP is e.g. a load balancer's IP.
#auth_proxy_self =

# Show more verbose process titles (in ps). Currently shows user name and
# IP address. Useful for seeing who are actually using the IMAP processes
# (eg. shared mailboxes or if same uid is used for multiple accounts).
#verbose_proctitle = no

# Should all processes be killed when Dovecot master process shuts down.
# Setting this to "no" means that Dovecot can be upgraded without
# forcing existing client connections to close (although that could also be
# a problem if the upgrade is e.g. because of a security fix).
#shutdown_clients = yes

# If non-zero, run mail commands via this many connections to doveadm server,
# instead of running them directly in the same process.
#doveadm_worker_count = 0
# UNIX socket or host:port used for connecting to doveadm server
#doveadm_socket_path = doveadm-server
```

```
# Space separated list of environment variables that are preserved on Dovecot
# startup and passed down to all of its child processes. You can also give
# key=value pairs to always set specific settings.
#import_environment = TZ

##
## Dictionary server settings
##

# Dictionary can be used to store key=value lists. This is used by several
# plugins. The dictionary can be accessed either directly or though a
# dictionary server. The following dict block maps dictionary names to URIs
# when the server is used. These can then be referenced using URIs in format
# "proxy::<name>".

dict {
  #quota = mysql:/etc/dovecot/dovecot-dict-sql.conf.ext
  #expire = sqlite:/etc/dovecot/dovecot-dict-sql.conf.ext
}

# Most of the actual configuration gets included below. The filenames are
# first sorted by their ASCII value and parsed in that order. The 00-prefixes
# in filenames are intended to make it easier to understand the ordering.
!include conf.d/*.conf

# A config file can also tried to be included without giving an error if
# it's not found:
!include_try local.conf

}
```

Maintenant dans le répertoire **/etc/dovecot/conf.d** il y a deux, trois bricoles à modifier

Dans le fichier **10-ssl.conf** on spécifie qu'on utilise une connexion chiffrée par **SSL** entre le serveur et le client en mettant

```
ssl = required

ssl_cert = </etc/ssl/certs/dovecot.pem
ssl_key = </etc/ssl/private/dovecot.pem
```

dans le fichier **10-auth.conf** on va mettre

```
disable_plaintext_auth = yes
auth_mechanisms = plain
```

en effet comme on utilise **SSL** on peut utiliser un mécanisme d'authentification en clair .

maintenant dans le fichier **auth-system.conf.ext** on indique qu'on passe par **PAM**

```
passdb {
  driver = pam
  # [session=yes] [setcred=yes] [failure_show_msg=yes] [max_requests=<n>]
  # [cache_key=<key>] [<service name>]
  #args = dovecot
}
userdb {
  # <doc/wiki/AuthDatabase.Passwd.txt>
  driver = passwd
  # [blocking=no]
  #args =

  # Override fields from passwd
  #override_fields = home=/home/virtual/%u
}
```

dans le fichier **10-mail.conf** on indique que les mailbox se trouvent sous **/var/mail** et portent le nom de chaque utilisateur du système

mail_location = mbox:~/mail:INBOX=/var/mail/%u
mail_privileged_group = mail

maintenant on relance **dovecot**

systemctl restart dovecot

si vous avez l'erreur suivante

oct. 10 10:49:26 mana.kervao.fr dovecot[19551]: pop3(olivier): Error: chown(/home/olivier/mail/.imap, group=12(mail)) failed: Operation not permitted (egid=5000(hoarau), group based on /var/mail/olivier - see http://wiki2.dovecot.org/Errors/ChgrpNoPerm)
oct. 10 10:49:26 mana.kervao.fr dovecot[19551]: pop3(olivier): Error: Couldn't open INBOX: Permission denied

il faudra modifier les droits des boites aux lettres comme ceci **/var/mail**

chmod 0600 /var/mail/*

Si dans les logs vous avez ce type de message

janv. 04 17:59:36 mana.kervao.fr dovecot[29061]: master: Warning: /power is no longer mounted. See http://wiki2.dovecot.org/Mountpoints

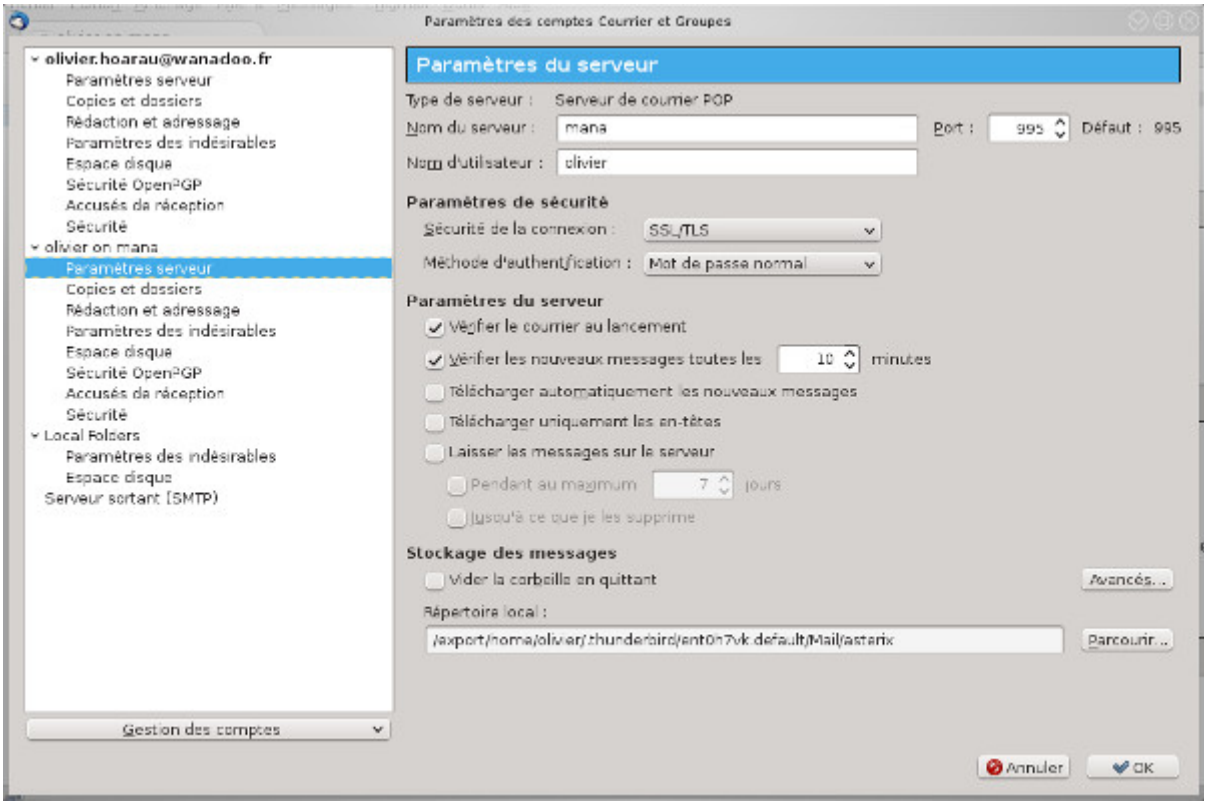
c'est un point de message qui a disparu, il suffit de l'indiquer à **dovecot** en tapant

doveadm mount remove /power

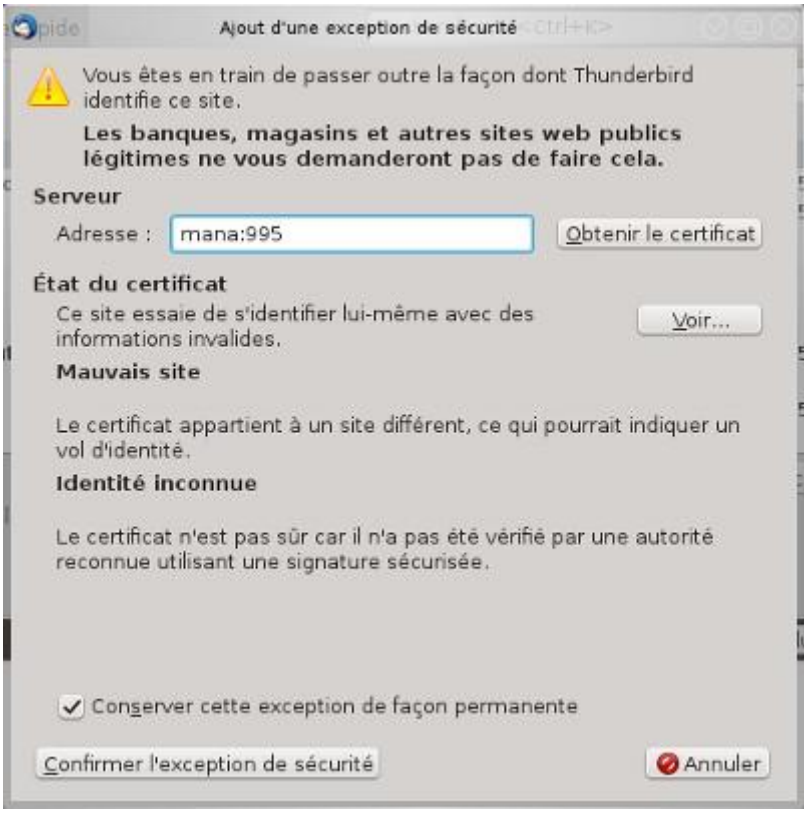
3.6 Configuration du client

Du coté poste client, configurer votre logiciel de mail favori pour que votre serveur Linux soit le serveur **POP** (il suffit de rajouter le nom du poste en question dans le champ qui va bien), au niveau de la sécurité de connexion on indique **SSL/TLS**.

Exemple ici avec **thunderbird**



à la première connexion il faudra sans doute confirmer l'exception de sécurité pour votre serveur de mail



et puis c'est tout, les courriers seront récupérés dans **/var/spool/mail** du serveur.

3.7 Renouvellement de certificat dovecot

Si jamais un jour ou l'autre votre client de messagerie préféré vous sort que le certificat du serveur est expiré et plus valide, il faudra le renouveler. Pour cela on va commencer par sauvegarder la clé privée et le certificat de **dovecot** en tapant:

cp /etc/ssl/private/dovecot.pem /etc/ssl/private/dovecot.pem.old
cp /etc/ssl/certs/dovecot.pem /etc/ssl/certs/dovecot.pem.old

on crée une nouvelle clé privée

openssl genrsa -out /etc/ssl/private/dovecot.pem 1024

voilà le résutat

Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x010001)

on crée le nouveau certificat d'une durée de validité de 2ans. Il est bien évident que ce certificat est perso et n'a aucune validité sur internet, il sera utile pour un usage strictement privé.

openssl req -new -x509 -key /etc/ssl/private/dovecot.pem -out /etc/ssl/certs/dovecot.pem -days 730

voilà le résultat

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]:Bzh
Locality Name (eg, city) []:Brest
Organization Name (eg, company) [Internet Widgits Pty Ltd]:none
Organizational Unit Name (eg, section) []:none
Common Name (e.g. server FQDN or YOUR name) []:mana
Email Address []:olivier.hoarau@funix.org

on relance **dovecot**

systemctl stop dovecot.service
systemctl start dovecot.service

il faudra sans doute à nouveau accepter le nouveau certificat dans le client de messagerie

Si dans les logs vous avez ce type de message

janv. 04 17:59:36 mana.kervao.fr dovecot[29061]: master: Warning: /power is no longer mounted. See http://wiki2.dovecot.org/Mountpoints

c'est un point de message qui a disparu, il suffit de l'indiquer à **dovecot** en tapant

doveadm mount remove /power

4 Sendmail

4.1 Présentation

Des outils comme **Firefox** disposent de leur propre mécanisme qui permette d'envoyer du courrier, pour faire plus "pro" on peut utiliser **sendmail** qui a été longtemps l'outil de référence pour gérer l'expédition de courrier dans le monde UNIX.

Pourquoi s'embêter à configurer **sendmail**, pour la gloire ? Certains disent que tant qu'on a pas chercher à configurer **sendmail**, on n'est pas un vrai administrateur système. Certain gestionnaire de courrier comme **Kmail**, en standard dans la bannière de KDE, fonctionne avec **sendmail**, or la configuration par défaut de ce dernier n'est pas correcte, si vous avez défini un nom de domaine farfelu pour votre machine, tous les courriers sortants vont partir avec au lieu d'utiliser le nom de domaine de votre fournisseur, vos emails seront, à coup sûr, rejeter par la plupart des gestionnaires de courrier sur le net, et le peu de personne qui recevra vos emails, ne pourront y répondre car votre nom de domaine est totalement inconnu sur le net.
Une autre raison d'utiliser sendmail est dans le cas d'un petit réseau privé, avec plusieurs postes Windows (ou autres) et un serveur Linux, pour pouvoir partager une connexion internet, **sendmail** va faire office de serveur SMTP.

Il existe maintenant **postfix** qui paraît-il est plus simple et plus sécurisé, il est largement documenté sur le net, comme les configs sous **sendmail** à l'heure actuelle sont plutôt rares, j'ai choisi d'être le dernier des mohicans.

On suppose que **machine** est le nom de votre machine tel que vous l'aurez défini avec **netcfg** ou **linuxconf**, **domaine.fr** est le nom de votre domaine qui se limite à votre machine ou éventuellement à votre réseau local, **mail.fai.fr**est le serveur de mail de votre provider. **toto** et **tata** sont les deux noms d'utilisateur que vous avez déclarés sur la machine. Vous ne disposez que d'un seul compte chez votre provider, mais par contre vous avez plusieurs emails, du style **toto.nom@fai.fr** et **tata.nom@fai.fr**.

4.2 Installation

Tout d'abord pour installer **sendmail**

urpmi sendmail

Si **postfix** cause des problèmes de conflit, forcer sa désinstallation ainsi

rpm -e --nodeps postfix-2.1.5-6mdk

(exemple donné pour LE 2005) Puis retentez l'installation de **sendmail**. Vous aurez encore besoin des packages suivants

sendmail-cf
m4

Et éventuellement le package de documentation **sendmail-doc**. Sachez que vous pourrez toujours trouver la doc de référence sur le site de sendmail.

4.3 Configuration

Pour une Mageia j'ai créé ensuite un fichier **/usr/share/sendmail-cf/cf/config.mc** qui a cette tête là:

```
include(`../m4/cf.m4')dnl
OSTYPE(`linux')dnl
FEATURE(redirect)dnl
FEATURE(nocanonify)dnl
FEATURE(always_add_domain)dnl
FEATURE(local_procmail)dnl
GENERICS_DOMAIN(machine.domaine.fr machine localhost)
FEATURE(genericstable)
FEATURE(masquerade_envelope)dnl
FEATURE(relay_entire_domain)dnl
FEATURE(accept_unresolvable_domains)dnl
define(`confDOMAIN_NAME',`ppp.fai.fr')dnl
define(`SMTP_MAILER_FLAGS',`e9')dnl
define(`confCON_EXPENSIVE', `True')dnl
define(`confME_TOO', `True')dnl
define(`confCF_VERSION', `Commentaire quelconque')dnl
define(`confCOPY_ERRORS_TO', `Postmaster')dnl
define(`confDEF_CHAR_SET', `ISO-8859-1')dnl
define(`confMIME_FORMAT_ERRORS', `True')dnl
define(`SMART_HOST', `smtp8:[mail.fai.fr]')dnl
define(`confRECEIVED_HEADER', `from fai.fr
        by fai.fr ($v/$Z)$?r with $r$. id $i$?u
        for $u; $|;
        $.$b')
define(`confTO_QUEUEWARN', `24h')dnl
MAILER(local)
MAILER(smtp)
Kpirateo hash -o /etc/mail/pirateo
LOCAL_RULE_0
R$+ < @ $+ > $*              $: < $(pirateo $1 @ $2 $: $) > $1 < @ $2 > $3
R< $+ > $+ < @ $+ > $*       $@ $>97 $1
R<> $+ < @ $+ > $*           $: $1 < @ $2 > $3

                ^^^^^^ tabulation unique à cet endroit, ailleurs un simple espace
```

Le **FEATURE(relay_entire_domain)** permet à **sendmail** d'accepter les mails venants des postes de votre réseau privé appartenant à votre domaine privé, sans ce parametre à l'envoi d'email, vous auriez sur les PC sous windows un message d'erreur du style "Relaying denied".
J'ai rajouté aussi **FEATURE(accept_unresolvable_domains)** car sans quoi si le PC sous linux est off-line pas moyen d'envoyer un mail d'un PC sous Windows vers le PC sous Linux, par contre dès qu'on passe on-line ce paramètre devient parfaitement inutile. Je ne comprends pas trop pourquoi mais je soupçonne une histoire de DNS la dessous.

Les dernières lignes (à partir de **Kpirateo**) permet que le courrier ne part chez le FAI en cas de réponse à un utilisateur du réseau local. Je m'explique, si un utilisateur local **toto** envoit un mail à un autre utilisateur local **tata**, l'email de l'expéditeur va être réécrite (fonction **genericstable**) **toto.nom@fai.fr**, si **tata** répond au mail, la réponse partira vers **toto.nom@fai.fr** et non pas simplement **toto**, en d'autres termes le courrier va d'abord partir chez le fai avant de revenir en local ! Ces lignes permettent que le courrier soit acheminé en local.

ATTENTION: à la tabulation unique dans les trois dernières lignes.

NOTE Attention pour qu'un mail parte en local vers le compte **toto** vous devez taper comme email de destination **toto** (sans le domaine), **toto@machine** ou bien encore **toto@machine.domaine.fr** avec **machine** le nom de votre serveur **sendmail** et **domaine.fr** celui de votre domaine, si vous mettez **toto@domaine.fr**, le mail partira vers le FAI avant de revenir sur le réseau local, même si **domaine.fr** est défini dans la variable **GENERICSTABLE**.

J'ai rajouté aussi **define(`confDOMAIN_NAME',...)** et **define(`confRECEIVED_HEADER',...)**se reporter au paragraphe masquage des domaines.

Ensuite on crée un fichier /**etc/mail/genericstable** , qui contient ces lignes:

toto: toto.nom@fai.fr
tata: tata.nom@fai.fr

Ce fichier fait la correspondance entre les adresses locales et les adresses "officielles".
Attention il faut mettre une tabulation entre le : et l'adresse.

La ligne **define(`confCF_VERSION', `Commentaire quelconque')** permet de rajouter un commentaire quelconque dans l'entête des mails (voir plus bas).

Pour faire prendre en compte la modif de ce fichier, il faut taper ensuite:

sendmail -bi -oA/etc/mail/genericstable

ATTENTION Attention pour les versions 8.9.X de **sendmail**, **generistable** se trouve directement sous **/etc** il faudra prendre en compte cette différence dans la suite des opérations si vous disposez de cette version.

On crée ensuite un fichier /**etc/mail/pirateo**, dans lequel vous mettrez :

toto.nom@fai.fr toto
tata.nom@fai.fr tata

Ensuite pour générer le fichier au format qui va bien on tape :

makemap hash /etc/mail/pirateo < /etc/mail/pirateo

Ce fichier aura le rôle inverse de **/etc/mail/generistable**, il transforme l'adresse du destinataire **toto.nom@fai.fr** en **toto** si celui est un utilisateur local, pour éviter que le mail parte chez le fai. Mon fichier **/etc/hosts** commence par la ligne suivante

127.0.0.1 localhost localhost.localdomain

N'oubliez pas de rajouter vos postes de votre réseau privé avec le FQDN (Fully qualified domain name, nom complet), ça nous donne donc ça:

127.0.0.1 localhost localhost.localdomain
192.168.13.10 machine.domaine.fr machine
192.168.13.11 windows.domaine.fr windows
192.168.13.12 mac.domaine.fr mac

windows et mac étant deux machines de votre réseau privé, les adresses IP sont données à titre indicatif.

ATTENTION: si **sendmail** bloque le boot de la machine, ça peut venir justement du fait qu'on n'a pas modifié la première ligne de /**etc/hosts**, **sendmail** n'arrive pas à trouver le nom de la machine et part dans une recherche qui par défaut dure 3 minutes. Par ailleurs ça peut engendrer des problèmes de résolution de nom sur la machine linux en mode off-line, pour résoudre ces problèmes tout en maintenant la ligne en question inchangée, reportez vous à la page installation d'un serveur DNS.

Pour rebatir le fichier de configuration de **sendmail** , on tape la commande:

cd /usr/share/sendmail-cf/cf/
m4 config.mc > /etc/mail/sendmail.cf

Changer (éventuellement) les droits de ce fichier:

chmod 600 /etc/mail/sendmail.cf

Ca y est c'est fini

4.4 On relance tout

Pour relancer tout, il faut d'abord préalablement tuer **sendmail** s'il tourne, pour cela faire:

systemctl stop sendmail

Puis pour relire le fichier de configuration

sendmail -bd -os

si vous obtenez ces erreurs

554 /etc/sendmail.cf: line 51: unknown configuration line "
"

En fait il suffit d'éditer **/etc/mail/sendmail.cf** et de supprimer quelques lignes vides au niveau de la ligne 51, pour que tout rentre dans l'ordre, tapez à nouveau la commande **sendmail -bd -os**.

Voilà ce que donne **systemctl status sendmail**

Ï sendmail.service - Sendmail Mail Transport Agent
Loaded: loaded (/usr/lib/systemd/system/sendmail.service; enabled)
Active: active (running) since dim. 2015-10-04 20:59:56 CEST; 11min ago
Process: 1665 ExecStart=/bin/sh -c exec /usr/sbin/sendmail.sendmail $DAEMONOPTIONS -bd $(if [-n "$QUEUE"]; then echo -q$QUEUE; fi) (code=exited, status=0/SUCCESS)
Process: 1657 ExecStartPre=/usr/bin/make -C /etc/mail -s (code=exited, status=0/SUCCESS)
Process: 1653 ExecStartPre=/usr/bin/newaliases (code=exited, status=0/SUCCESS)
Main PID: 1667 (sendmail.sendma)
CGroup: /system.slice/sendmail.service
ý1667 sendmail: accepting connections

oct. 04 20:59:55 mana.kervao.fr sendmail[1653]: alias database /etc/aliases rebuilt by root
oct. 04 20:59:55 mana.kervao.fr newaliases[1653]: /etc/aliases: 14 aliases, longest 10 bytes, 152 bytes total
oct. 04 20:59:55 mana.kervao.fr sendmail[1653]: /etc/aliases: 14 aliases, longest 10 bytes, 152 bytes total
oct. 04 20:59:56 mana.kervao.fr sendmail[1667]: starting daemon (8.15.1): SMTP+queueing@01:00:00

4.5 Principe de fonctionnement

Pour info tous les courriers partants se retrouvent en attente dans le répertoire **/var/spool/mqueue**, ceux en arrivée se trouvent dans **/var/spool/mail** avec pour nom le nom du destinataire sur la machine.

ATTENTION: **/var/spool/mail** doit avoir pour droit 01777 (**drwxrwsr-x**)

sous **/var/spool** sous mageia voilà ce que j'ai en tapant **ll** dans un shell

drwxrwx--- 2 mail mail 4096 oct. 15 04:02 clientmqueue/
drwxrwsr-x 5 root mail 4096 oct. 15 12:23 mail/
drwxr-x--- 2 root mail 4096 oct. 15 12:01 mqueue/

Pour envoyer le courrier, une fois connecté vous devez taper:

/usr/sbin/sendmail -q -v

L'option **-v** étant l'option "verbeuse". Pour visualiser les messages dans la file d'attente, vous pouvez taper:

mailq

Quand vous envoyez un courrier en local (de **toto** vers **tata**), le courrier ne va pas transiter par **mqueue**, de même qu'il est inutile de tapez "**sendmail -q**", il va se retrouver directement dans la boite aux lettres du destinataire local, avec dans le champ **From toto@domaine.fr** (c'est le but du paramêtre **FEATURE(always_add_domain)** qui va rajouter automatiquement le nom de domain privé).
En cas d'envoi vers un destinataire extérieur à votre domaine, le courrier va se retrouver dans le répertoire **mqueue**

4.6 Masquage de domaine

Si on se contente d'appliquer la doc de Jacoboni brut de forme, on est confronté à un gros problème qui se voit à l'envoi du courrier vers des adresses échos comme **echo@cnam.fr**, qui se contente de vous renvoyer votre email

avec l'entête complète du mail d'origine, on peut y voir des informations indiscrètes que vous ne voudriez pas forcément voir figurer.

Voici le mail qui part du serveur Linux, avec pour contenu :

Subject: test
première ligne

Contenu de l'email de réponse du serveur écho du CNAM:

------ Le serveur echo du domaine cnam.fr
------ a recu votre message le mar 10 aoû 22:25:28 MET DST 1999

------ Ci-dessous les en-tetes et le corps de votre message

> From toto.nom@fai.fr Tue Aug 10 22:25:27 1999
> Received: from obelix.fai.fr (obelix.fai.fr [210.205.98.21])
> by fermi.cnam.fr (8.8.8/jpm-301097) with ESMTP id WAA11452
> for <echo@cnam.fr>; Tue, 10 Aug 1999 22:25:27 +0200 (MET DST)
> From: toto.nom@fai.fr
> Return-Path: <toto.nom@fai.fr>
> Received: from machine.domaine.fr (IDENT:root@ppptc22.fai.fr [210.205.98.22])
> by obelix.fai.fr (8.9.1/8.9.1/R&D&B-990119) with ESMTP id WAA26056
> for <echo@cnam.fr>; Tue, 10 Aug 1999 22:24:55 +0200
> Received: (from toto@localhost)
> by machine.domaine.fr (8.9.3/8.9.3/Commentaire quelconque) id WAA00754
> for echo@cnam.fr; Tue, 10 Aug 1999 22:25:34 +0200
> Date: Tue, 10 Aug 1999 22:25:34 +0200
> Message-Id: <199908102025.WAA00754@machine.domaine.fr>
> To: echo@cnam.fr
> Subject: test

>
> première ligne
>

------ Fin de votre message

Quelques commentaires:

toto.nom@fai.fr est votre adresse email chez votre fournisseur d'accès, **obelix.fai.fr (IP= 210.205.98.21)** est le nom de la machine chez votre fai qui a "routé" votre email, **ppptc22.fai.fr (IP=210.205.98.22)** c'est votre identité officielle sur le net au moment de votre connexion.

Vous voyez que le nom de votre domaine apparaît dans les champs **Received**, et même le commentaire que vous aurez défini dans le fichier de config de **sendmail**. Certains gestionnaires d'émail pourraient rejeter vos emails sous prétexte de contenir un nom de domaine inconnu.

Voyons maintenant un email arrivant d'un de vos postes sous Windows et partant vers le net.

------ Le serveur echo du domaine cnam.fr
------ a recu votre message le mar 10 aoû 19:47:08 MET DST 1999

------ Ci-dessous les en-tetes et le corps de votre message

> From toto.nom@fai.fr Tue Aug 10 19:47:07 1999
> Received: from obelix.fai.fr (obelix.fai.fr [210.205.98.21])
> by fermi.cnam.fr (8.8.8/jpm-301097) with ESMTP id TAA05598
> for <echo@cnam.fr>; Tue, 10 Aug 1999 19:47:06 +0200 (MET DST)
> Return-Path: <toto.nom@fai.fr>
> Received: from machine.domaine.fr (IDENT:root@ppptc32.fai.fr [210.205.98.23])
> by obelix.fai.fr (8.9.1/8.9.1/R&D&B-990119) with ESMTP id TAA23230
> for <echo@cnam.fr>; Tue, 10 Aug 1999 19:46:35 +0200
> Received: from windows (windows.domaine.fr [192.168.13.11])
> by machine.domaine.fr (8.9.3/8.9.3/Commentaire quelconque) with ESMTP id TAA00863
> for <echo@cnam.fr>; Tue, 10 Aug 1999 19:38:54 +0200
> Message-Id: <199908101738.TAA00863@machine.domaine.fr>
> From: "Toto Nom" <toto.nom@fai.fr>
> To: <echo@cnam.fr>
> Subject: test de windows
> Date: Tue, 10 Aug 1999 19:37:08 +0200
> X-MSMail-Priority: Normal
> X-Priority: 3
> X-Mailer: Microsoft Internet Mail 4.70.1155
> MIME-Version: 1.0
> Content-Type: text/plain; charset=ISO-8859-1
> Content-Transfer-Encoding: 7bit

>
> première ligne
>

------ Fin de votre message

Dans **Received**, on voit en fait le cheminement que suit le mail envoyé du poste **windows**, va sur **machine** puis par chez votre fai (sur **obelix**), on voit donc le nom de votre domaine, les noms du poste Linux et du poste de votre réseau privé d'où a été envoyé l'email, et même l'adresse IP que vous lui avez attribué !

Le problème est qu'on ne peut dans les paramètres de config de **sendmail**, virer les champs **Received**, **FEATURE(masquerade_enveloppe)** ne fait que masquer les adresses emails.

Pour corriger ça, on va d'abord rajouter **define(`confDOMAIN_NAME',`ppp.fai.fr')** qui permet de redéfinir la manière dont notre serveur Linux va se présenter au serveur SMTP du provider, en clair il va changer toutes les occurrences de **machine.domaine.fr** par **ppp.fai.fr** dans les champs **Received**. Pourquoi mettre **ppp.fai.fr** et ne pas mettre tout simplement **fai.fr**, parce que dans ce cas on ne pourra pas envoyer de mail à des utilisateurs du domaine **fai.fr**, **sendmail** croit que ce sont des utilisateurs locaux ! Avec **ppp.fai.fr** pas de problème, en toute rigueur on pourrait mettre ici le nom attribué lors d'une connexion (du style **ppp18-brest.fai.fr** qu'on peut voir en tapant **ifconfig**) pour cela reporter vous au paragraphe attribution d'adresse dynamique.

Reste le problème des emails partant de postes sous Windows, où apparaît le nom et l'adresse IP, on va carrément redéfinir le champs **Received**:

```
define(`confRECEIVED_HEADER', `from fai.fr
      by fai.fr ($v/$Z)$?r with $r$. id $i$?u
      for $u; $|;
      $.$b')
```

Ce qui nous donne pour un mail envoyé d'un PC sous windows:

------ Le serveur echo du domaine cnam.fr
------ a recu votre message le sam jun 1 19:08:31 CEST 2002
------ Ci-dessous les en-tetes et le corps de votre message
> From toto.nom@fai.fr Sat Jun 01 17:08:31 2002

> Return-Path: <toto.nom@fai.fr>
> Delivered-To: echo@cnam.fr
> Received: (qmail 20719 invoked from network); 1 Jun 2002 17:08:31 -0000
> Received: from bougainville.cnam.fr (163.173.128.13)
> by 0 with SMTP; 1 Jun 2002 17:08:31 -0000
> Received: from localhost (localhost [127.0.0.1])
> by bougainville.cnam.fr (Postfix) with ESMTP id 132F82EFB4
> for <echo@cnam.fr>; Sat, 1 Jun 2002 19:08:31 +0200 (CEST)
> Received: from smtp.fai.fr (mail.fai.fr [202.3.225.22])
> by bougainville.cnam.fr (Postfix) with ESMTP id D31492EFAE
> for <echo@cnam.fr>; Sat, 1 Jun 2002 19:08:27 +0200 (CEST)
> Received: from ppp.fai.fr (tc5-bis-014.dialup.fai.fr [202.3.239.14])
> by smtp.fai.fr (Mirapoint Messaging Server MOS 3.1.0.36-EA)
> with ESMTP id ADS19083
> for <echo@cnam.fr>;
> Sat, 1 Jun 2002 07:08:20 -1000 (TAHT)
> Received: from fai.fr
> by fai.fr (8.12.1/8.12.1/Commentaire quelconque) with ESMTP id g51H7FAj002161
> for <echo@cnam.fr>; Sat, 1 Jun 2002 07:07:15 -1000
> Message-ID: <3CF90D0A.5FF74480@fai.fr>
> Date: Sat, 01 Jun 2002 07:06:02 -1100
> From: Toto Nom <toto.nom@fai.fr>
> X-Mailer: Mozilla 4.7 [fr] (WinNT; I)
> X-Accept-Language: fr
> MIME-Version: 1.0
> To: echo@cnam.fr
> Subject: essai
> Content-Type: text/plain; charset=us-ascii
> Content-Transfer-Encoding: 7bit
> X-Virus-Scanned: by AMaViS perl-11

>
> première ligne
>

------ Fin de votre message

Voici l'email qui part de notre serveur Linux.

------ Le serveur echo du domaine cnam.fr
------ a recu votre message le sam jun 1 07:51:15 CEST 2002
------ Ci-dessous les en-tetes et le corps de votre message
> From toto.nom@fai.fr Sat Jun 01 05:51:15 2002
> Return-Path: <toto.nom@fai.fr>
> Delivered-To: echo@cnam.fr
> Received: (qmail 3855 invoked from network); 1 Jun 2002 05:51:14 -0000
> Received: from bougainville.cnam.fr (163.173.128.13)
> by 0 with SMTP; 1 Jun 2002 05:51:14 -0000
> Received: from localhost (localhost [127.0.0.1])
> by bougainville.cnam.fr (Postfix) with ESMTP id CB0A52EFAF
> for <echo@cnam.fr>; Sat, 1 Jun 2002 07:51:14 +0200 (CEST)
> Received: from smtp.fai.fr (mail.fai.fr [202.3.225.22])
> by bougainville.cnam.fr (Postfix) with ESMTP id 8EE6D2EFAE
> for <echo@cnam.fr>; Sat, 1 Jun 2002 07:51:11 +0200 (CEST)
> Received: from ppp.fai.fr (tc5-bis-198.dialup.fai.fr [202.3.239.198])
> by smtp.fai.fr (Mirapoint Messaging Server MOS 3.1.0.36-EA)
> with ESMTP id ADS03894
> for <echo@cnam.fr>;
> Fri, 31 May 2002 19:50:40 -1000 (TAHT)
> Received: from fai.fr
> by fai.fr (8.12.1/8.12.1/Commentaire quelconque) with ESMTP id g515nwFE002664

> for <echo@cnam.fr>; Fri, 31 May 2002 19:49:58 -1000
> Sender: toto.nom@fai.fr
> Message-ID: <3CF86086.351F9560@fai.fr>
> Date: Fri, 31 May 2002 19:49:58 -1000
> From: Toto Nom <toto.nom@fai.fr>
> Organization: Tahiti Connection
> X-Mailer: Mozilla 4.78 [fr] (X11; U; Linux 2.4.18-6mdk i686)
> X-Accept-Language: en
> MIME-Version: 1.0
> To: echo@cnam.fr
> Subject: test
> Content-Type: text/plain; charset=us-ascii
> Content-Transfer-Encoding: 7bit
> X-Virus-Scanned: by AMaViS perl-11

>
> première ligne
>

------ Fin de votre message

4.7 Si vous avez un autre email que celui attribué par le Fai utilisé

Vous pouvez très bien utiliser un FAI du style **fai.fr** et ne pas utiliser une adresse email en **@fai.fr.** Ainsi je me connecte avec **free** et mon adresse email est **olivier.hoarau@fnac.net** , dans ce cas le **Message-Id** et le **Sender** n'ont pas une bonne tête**.**
Exemple avec cette entête renvoyée par **echo@cnam.fr**

> From olivier.hoarau@fnac.net Sat Jul 15 08:19:16 2000
> Received: from postfix3.free.fr (postfix@postfix3.free.fr [212.27.32.22])
> by fermi.cnam.fr (8.8.8/jpm-301097) with ESMTP id IAA05796
> for <echo@cnam.fr>; Sat, 15 Jul 2000 08:19:16 +0200 (MET DST)
> Return-Path: <olivier.hoarau@fnac.net>
> Received: from ppp.free.fr (massy-4-14-209.dial.proxad.net [213.228.14.209])
> by postfix3.free.fr (Postfix) with ESMTP id 62B6286B67
> for <echo@cnam.fr>; Sat, 15 Jul 2000 08:19:15 +0200 (CEST)
> Received: from free.fr
> by free.fr (8.10.1/8.10.1/Olivier Hoarau-992911) with ESMTP id e6F68IJ01419
> for <echo@cnam.fr>; Sat, 15 Jul 2000 08:08:18 +0200
> Sender: olivier@free.fr
> Message-ID: <396FFFD2.447D80E1@fnac.net>
> Date: Sat, 15 Jul 2000 08:08:18 +0200
> From: Olivier Hoarau <olivier.hoarau@fnac.net>
> Organization: Breizh Connection
> X-Mailer: Mozilla 4.73 [fr] (X11; I; Linux 2.2.15-4mdk i686)
> X-Accept-Language: en
> MIME-Version: 1.0
> To: echo@cnam.fr
> Subject: asterix
> Content-Type: text/plain; charset=us-ascii
> Content-Transfer-Encoding: 7bit

Vous voyez que le **Message-Id** a l'extension **fnac.net** alors qu'il devrait être en **free.fr** puisque j'utilise **free**, de même le **Sender** est **olivier@free.fr** qui ne correspond à rien, vu que mon email chez **free** est **ohoarau@free.fr**

Pour régler le problème du **Message-Id** et du **Sender**, j'ai écrit ce petit script :

```
#!/bin/bash
cd /var/spool/mqueue
for nom_mail in $(ls qf*)
do
    awk 'BEGIN { FS=":" }
```

```
      $1!="H??Message-ID"   &&    $1!= "H??Sender" { print $0 }
      $1=="H??Sender" { sub("olivier","ohoarau",$2);print $1,":",$2 }
      $1=="H??Message-ID" {  sub("fnac.net","free.fr",$2); print $1,":",$2 }
    ' $nom_mail > /tmp/mail.tmp
    cp /tmp/mail.tmp $nom_mail
done
```

Vous pourrez très facilement adapter ce script à votre situation, quelques commentaires sont peut être utiles:

- **sub("olivier","ohoarau",$2)** ici c'est pour avoir **Sender: ohoarau@free.fr** au lieu de **Sender: olivier@free.fr**
- **sub("fnac.net","free.fr",$2)** ici c'est pour avoir **Message-ID: <396FFFD2.447D80E1@free.fr** au lieu de **Message-ID: <396FFFD2.447D80E1@fnac.net**

Remplacez les chaînes de caractères adéquates pour que ça marche chez vous. Le proprio du script doit être root, avec des droits en 755, on l'appellera avant **sendmail -q** et qui permettra de changer le **Message-Id** et le **Sender**.

Si ce script s'appelle **chg-message** et se trouve dans **/usr/sbin**, vous pouvez le mettre dans le fichier **/etc/ppp/ip-up** lancé à chaque connexion, comme ceci

```
/usr/sbin/chg-message
/usr/sbin/sendmail -q
```

Voilà en final la tête mon mail envoyé de mon poste linux tel que l'a renvoyé le serveur écho du cnam:

```
------ Le serveur echo du domaine cnam.fr
------ a recu votre message le sam 15 jul 09:23:10 MET DST 2000

------ Ci-dessous les en-tetes et le corps de votre message


> From olivier.hoarau@fnac.net  Sat Jul 15 09:23:09 2000
> Received: from postfix1.free.fr (postfix@postfix1.free.fr [212.27.32.21])
>        by fermi.cnam.fr (8.8.8/jpm-301097)  with ESMTP id JAA10064
>        for <echo@cnam.fr>; Sat, 15 Jul 2000 09:23:09 +0200 (MET DST)
> Return-Path: <olivier.hoarau@fnac.net>
> Received: from ppp.free.fr (massy-2-11-231.dial.proxad.net [213.228.11.231])
>      by postfix1.free.fr (Postfix) with ESMTP id DA7D228043
>      for <echo@cnam.fr>; Sat, 15 Jul 2000 09:23:03 +0200 (MEST)
> Received: from free.fr
>        by free.fr (8.10.1/8.10.1/Olivier Hoarau-992911) with ESMTP id e6F72bP02146
>        for <echo@cnam.fr>; Sat, 15 Jul 2000 09:02:37 +0200
> Sender: ohoarau@free.fr
> Message-ID:  <39700C8D.3443567C@free.fr>
> Date: Sat, 15 Jul 2000 09:02:37 +0200
> From: Olivier Hoarau <olivier.hoarau@fnac.net>
> Organization: Breizh Connection
> X-Mailer: Mozilla 4.73 [fr] (X11; I; Linux 2.2.15-4mdk i686)
> X-Accept-Language: en
> MIME-Version: 1.0
> To: echo@cnam.fr
> Subject: essai
> Content-Type: text/plain; charset=us-ascii
> Content-Transfer-Encoding: 7bit
```

Le **Message-Id** et le **Sender** ont maintenant une bonne tête.

Voilà le message renvoyé d'un mail partant d'un poste client windows du réseau:

------ Le serveur echo du domaine cnam.fr
------ a recu votre message le sam 15 jul 09:46:21 MET DST 2000

------ Ci-dessous les en-tetes et le corps de votre message

> From olivier.hoarau@fnac.net Sat Jul 15 09:46:21 2000
> Received: from postfix2.free.fr (postfix@postfix2.free.fr [212.27.32.74])
> by fermi.cnam.fr (8.8.8/jpm-301097) with ESMTP id JAA11374
> for <echo@cnam.fr>; Sat, 15 Jul 2000 09:46:21 +0200 (MET DST)
> Return-Path: <olivier.hoarau@fnac.net>
> Received: from ppp.free.fr (massy-2-10-239.dial.proxad.net [213.228.10.239])
> by postfix2.free.fr (Postfix) with ESMTP id 7FB1D740DB
> for <echo@cnam.fr>; Sat, 15 Jul 2000 09:46:20 +0200 (MEST)
> Received: from free.fr
> by free.fr (8.10.1/8.10.1/Olivier Hoarau-992911) with ESMTP id e6F7iTG02453
> for <echo@cnam.fr>; Sat, 15 Jul 2000 09:44:30 +0200
> Message-ID: <39701660.10A22F73@free.fr>
> Date: Sat, 15 Jul 2000 09:44:32 +0200
> From: Olivier Hoarau <olivier.hoarau@fnac.net>
> X-Mailer: Mozilla 4.6 [fr] (Win98; I)
> X-Accept-Language: fr
> MIME-Version: 1.0
> To: echo@cnam.fr
> Subject: tavel
> Content-Type: text/plain; charset=us-ascii
> Content-Transfer-Encoding: 7bit

Vous pouvez constater qu'il n'y a pas de champ **Sender**.

4.8 Configurer sendmail pour utiliser une connexion chiffrée via SSL/TLS

Le problème avec la configuration présentée plus haut est que le mot de passe circule en clair entre vos clients et le serveur, on va les chiffrer en utilisant **SSL/TLS**. On commence d'abord à créer une clé privée **sendmail** en tapant:

openssl genrsa -des3 -out /etc/ssl/private/sendmail.key 1024

voilà le résultat

Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x010001)
Enter pass phrase for /etc/ssl/private/sendmail.key:
Verifying - Enter pass phrase for /etc/ssl/private/sendmail.key:

on met les droits qui va bien

chmod 400 /etc/ssl/private/sendmail.key

on crée maintenant une clé publique **sendmail**

openssl rsa -in /etc/ssl/private/sendmail.key -out /etc/ssl/public/sendmail.key.pub

voilà le résultat

Enter pass phrase for /etc/ssl/private/sendmail.key:
writing RSA key

on crée maintenant le certificat d'une durée de 3650 jours (!) sur la base de la clé publique de **sendmail**. Il est bien évident que ce certificat est perso et n'a aucune validité sur internet, il sera utile pour un usage strictement privé.

openssl req -new -x509 -days 3650 -key /etc/ssl/public/sendmail.key.pub -out /etc/ssl/certs/sendmail.crt

voilà le résultat

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]:Bzh
Locality Name (eg, city) []:Brest
Organization Name (eg, company) [Internet Widgits Pty Ltd]:none
Organizational Unit Name (eg, section) []:none
Common Name (e.g. server FQDN or YOUR name) []:mana
Email Address []:olivier.hoarau@funix.org

on lui donne les droits qui vont bien

chmod 400 /etc/ssl/certs/sendmail.crt

on modifie maintenant le fichier de configuration de sendmail **/usr/share/sendmail/cf/config.mc** et on rajoute à la fin

define(`confCACERT_PATH', `/etc/ssl/certs')dnl
define(`confCACERT', `/etc/ssl/certs/sendmail.crt')dnl
define(`confSERVER_CERT', `/etc/ssl/certs/sendmail.crt')dnl
define(`confSERVER_KEY', `/etc/ssl/public/sendmail.key.pub')dnl
define(`confCLIENT_KEY', `/etc/ssl/certs/sendmail.crt')dnl
DAEMON_OPTIONS(`Port=smtps, Name=TLSMTA, M=s')dnl

Pour rebâtir le fichier de configuration de **sendmail** , on tape la commande:

cd /usr/share/sendmail-cf/cf/
m4 config.mc > /etc/mail/sendmail.cf

on arrête **sendmail**

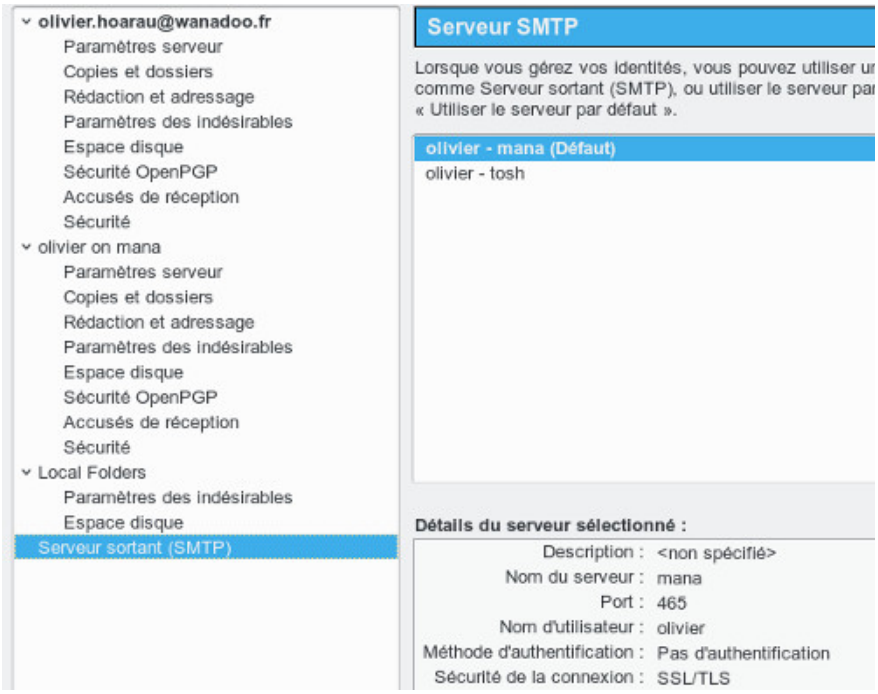
systemctl stop sendmail.service

et on le relance

systemctl start sendmail.service

si vous avez un firewall sur votre serveur il faudra penser à ouvrir le port **SMTPS** 465 et à fermer le port 25.

Maintenant sur **Thunderbird** on configurera le serveur sortant en choisissant **SSL/TLS** et en mettant le port 465 comme ceci



la première fois qu'on enverra un mail il faudra accepter l'exception de sécurité (certificat non valide)

et voilà ce qu'on peut voir dans les logs du serveur de mail

janv. 05 15:07:52 mana.kervao.fr sendmail[24577]: STARTTLS=server, relay=predator.kervao.fr [192.168.0.16], version=TLSv1.2, verify=NO, cipher=DHE-RSA-AES128-SHA, bits=128/128
janv. 05 15:07:52 mana.kervao.fr sendmail[24577]: v05E7qmr024577: from=<olivier.hoarau@funix.org>, size=591, class=0, nrcpts=1, msgid=<48077595-dd21-763b-44b1-b9d96e21aed3@funix.org>, bodytype=8BITMIME, proto=ESMTPS, daemon=TLSMTA, relay=predator.kervao.fr [192.168.0.16]
janv. 05 15:07:52 mana.kervao.fr sendmail[24577]: v05E7qmr024577: Milter add: header: X-Virus-Scanned: clamav-milter 0.98.7 at mana.kervao.fr
janv. 05 15:07:52 mana.kervao.fr sendmail[24577]: v05E7qmr024577: Milter add: header: X-Virus-Status: Clean

janv. 05 15:07:52 mana.kervao.fr spamd[22335]: spamd: connection from mana.kervao.fr [::1]:35296 to port 783, fd 5
janv. 05 15:07:52 mana.kervao.fr spamd[22335]: spamd: handle_user (userdir) unable to find user: 'veronique.hoarau'
janv. 05 15:07:52 mana.kervao.fr spamd[22335]: spamd: processing message <48077595-dd21-763b-44b1-b9d96e21aed3@funix.org> for veronique.hoarau:8
janv. 05 15:07:58 mana.kervao.fr spamd[22335]: spamd: clean message (-2.8/4.0) for veronique.hoarau:8 in 6.5 seconds, 1015 bytes.
janv. 05 15:07:58 mana.kervao.fr spamd[22335]: spamd: result: . -2 - ALL_TRUSTED,AWL,BAYES_00 scantime=6.5,size=1015,user=veronique.hoarau,uid=8,required_score=4.0,rhost=mana.kervao.fr,raddr=::1,rport=35296,mid=<48077595-dd21-763b-44b1-b9d96e21aed3@funix.org>,bayes=0.000000,autolearn=ham autolearn_force=no
janv. 05 15:07:58 mana.kervao.fr sendmail[24577]: v05E7qmr024577: Milter add: header: X-Spam-Status: No, score=-2.8 required=4.0 tests=ALL_TRUSTED,AWL,BAYES_00\n\tautolearn=ham autolearn_force=no version=3.4.1
janv. 05 15:07:58 mana.kervao.fr sendmail[24577]: v05E7qmr024577: Milter add: header: X-Spam-Checker-Version: SpamAssassin 3.4.1 (2015-04-28) on mana.kervao.fr
janv. 05 15:07:58 mana.kervao.fr sendmail[24577]: v05E7qmr024577: to=<veronique.hoarau@funix.org>, delay=00:00:06, mailer=smtp8, pri=30591, stat=queued
janv. 05 15:07:58 mana.kervao.fr spamd[2197]: prefork: child states: II

dans **thunderbird** dans les préférences, on peut visualiser le certificat (on peut voir également celui de dovecot)

OK nous avons chiffré la connexion entre le serveur local et les clients locaux mais qu'en est-il quand les mails partent vers le serveur **SMTP** du fournisseur d'accès sur internet ?
A vrai dire il suffit de le tester, pour ma part je suis chez numericable, on va envoyer un mail puis forcer un envoi avec **sendmail** en mode verbeux comme ceci

sendmail -q -v

voilà le résultat

Running /var/spool/mqueue/v06JL1vY031912 (sequence 1 of 1)
<veronique.hoarau@funix.org>... Connecting to smtp.numericable.fr. via smtp8...

réponse du serveur **SMTP** de numericable

220 smtp2.tech.numericable.fr ESMTP Postfix

mon serveur lui envoie un hello !

>>> EHLO ppp.numericable.fr

le serveur numericable lui envoie ses fonctionnalités

250-smtp2.tech.numericable.fr
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250 8BITMIME

la commande **PIPELINING** permet d'accélérer la connexion **SMTP** en envoyant les commandes d'un bloc et pas l'une après l'autre.
SIZE 102400000 indique que les mails sont limités à 10Mo
ETRN pour dire à mon serveur stmp de lui envoyer tous ses messages
ENHANCEDSTATUSCODES c'est pour étoffer les codes de retour de commande
8BITMIME mode de transfert sous 8bits pour faire passer tous les caractères de l'UTF-8
et enfin **STARTTLS** pour passer en mode chiffré avec **TLS**

mon serveur lance donc une connexion chiffrée

>>> STARTTLS

voilà la réponse du serveur **SMTP** numericable

220 2.0.0 Ready to start TLS

mon serveur lui renvoie un hello

>>> EHLO ppp.numericable.fr

et le serveur **SMTP** de numericable renvoie à nouveau les fonctionnalités

250-smtp2.tech.numericable.fr
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-ENHANCEDSTATUSCODES
250 8BITMIME

le transfert de mail s'effectue ensuite avec un dialogue entre les 2 serveurs

>>> MAIL From:<olivier.hoarau@funix.org> SIZE=1064 BODY=7BIT
250 2.1.0 Ok
>>> RCPT To:<veronique.hoarau@funix.org>
>>> DATA
250 2.1.5 Ok
354 End data with <CR><LF>.<CR><LF>
>>> .
250 2.0.0 Ok: queued as C441561944
<veronique.hoarau@funix.org>... Sent (Ok: queued as C441561944)
Closing connection to smtp.numericable.fr.
>>> QUIT
221 2.0.0 Bye

voilà, voilà, comme quoi il n'y a rien à configurer la connexion est chiffrée par défaut.

4.9 Sendmail et timeout DNS

Si avec Microsoft Internet Mail, il est impossible d'envoyer des mails d'un poste Windows vers le serveur Linux quand celui-ci est offline, avec l'erreur suivante dans les fichiers de log:

00762 >>> 220 machine.domaine.fr ESMTP Sendmail 8.9.3/8.9.3/Commentaire quelconque qui apparaitre dans l'entete - 15/08/99; Sat, 21 Aug 1999 09:32:54 +0200
00762 <<< EHLO windows
00762 >>> 250-machine.domaine.fr Hello windows.domaine.fr [192.168.13.11], pleased to meet you
00762 >>> 250-EXPN
00762 >>> 250-VERB
00762 >>> 250-8BITMIME
00762 >>> 250-SIZE
00762 >>> 250-DSN
00762 >>> 250-ONEX
00762 >>> 250-ETRN
00762 >>> 250-XUSR
00762 >>> 250 HELP
00762 <<< RSET
00762 >>> 250 Reset state
00762 <<< MAIL FROM:<toto.nom@fai.fr>
00763 >>> 250 <toto.nom@fai.fr>... Sender ok
00763 <<< RCPT TO:<echo@cnam.fr>
00763 >>> 250 <echo@cnam.fr>... Recipient ok
00763 <<< [EOF]
00763 >>> 421 machine.domaine.fr Lost input channel from windows.domaine.fr [192.168.13.11]

Et que par contre il n'y a aucun problème quand le serveur est on-line. C'est que vous avez un problème de DNS. A noter que le problème est similaire avec Outlook Express et d'une manière générale avec les outils de mail de Microsoft .

Si avec Netscape l'envoi de mail en mode off-line vers la file d'attente (répertoire **mqueue**) prend au moins 80s autant dire un éternité, que ce soit d'un poste client ou du poste serveur. C'est que vous avez aussi un problème de DNS.

Pour résoudre ça, il faut installer un serveur DNS sur la machine.

4.10 Sendmail et la lutte anti-spam

Vous pouvez faire appel à des serveurs qui listent les serveurs de mails indélicats (Open Relay), bien souvent les spams viennent de ces domaines. **sendmail** ira d'abord vérifier si le mail ne vient pas de ces domaines avant de le délivrer localement. Voilà les lignes à rajouter dans votre fichier **/usr/share/sendmail-cf/cf/config.mc**

FEATURE(dnsbl, `blackholes.mail-abuse.org', `Rejected - see http://www.mail-abuse.org/rbl/')dnl
FEATURE(dnsbl, `dialups.mail-abuse.org', `Dialup - see http://www.mail-abuse.org/dul/')dnl
FEATURE(dnsbl, `relays.mail-abuse.org', `Open spam relay - see http://www.mail-abuse.org/rss/')dnl
FEATURE(`dnsbl', `list.dsbl.org')dnl
FEATURE(`dnsbl', `bl.spamcop.net')dnl
FEATURE(`dnsbl', `sbl.spamhaus.org')dnl

Vous devez ensuite reconstuire **/etc/mail/sendmail.cf** avec **m4** et relancer **sendmail**.

4.11 Sendmail et fichiers de log

Sous une Mandrake 8 les fichiers de logs sont sous **/var/log/mail** et sont décomposés en :

errors contenant les erreurs
info contenant les informations diverses (récupération et expédition de mails)
warnings comme son nom l'indique

Sur les versions antérieurs le fichier de log étaient **/var/log/maillog** ou encore **/var/log/mail.log**, ce fichier rassemblait les erreurs, informations et warnings.

4.12 Sécuriser Sendmail

4.12.1 Les commandes vrfy et expn

Par défaut on peut effectuer un **telnet** sur le port utilisé par **sendmail** et l'interroger avec des commandes, cela permet notamment de connaître les emails déclarés sur le serveur et même les emails d'une liste ou d'un alias. Démonstration, pour la connexion

olivier@zoulou olivier]$ telnet zoulou 25
Trying 192.168.13.11...
Connected to zoulou.kervao.fr.
Escape character is '^]'.
220 rennes-1-a7-7-251.dial.proxad.net ESMTP Sendmail 8.11.3/8.11.3/Olivier Hoarau-992911;
Sat, 16 Jun 2001 09:40:20 –0400

Vous disposer de **vrfy** (verify) pour vérifier l'existence d'une adresse sur le serveur

vrfy olivier
250 2.1.5 olivier@rennes-1-a7-7-251.dial.proxad.net

Quand l'utilisateur est inconnu

vrfy toto
550 5.1.1 toto... User unknown

Vous disposez de la commande **expn** (expand) qui est identique à **vrfy** mais qui permet aussi de lister les personnes d'une liste et autres alias.

expn olivier
250 2.1.5 olivier@rennes-1-a7-7-251.dial.proxad.net

Pour désactiver les commandes **vfry** et **expn** vous devez rajouter au fichier de config **config.mc** la ligne suivante:

define(`confPRIVACY_FLAGS', `novrfy noexpn')dnl

Voilà le résultat:

vrfy olivier
252 2.5.2 Cannot VRFY user; try RCPT to attempt delivery (or try finger)

Pour info la commande **finger** est désactivé sur **sendmail** de la Mandrake 8.X

NOTE Pour sortir du **telnet** je n'ai pas trouvé mieux que de faire un **kill** du process

4.12.2 Modifier l'invite de sendmail

En faisant un **telnet** on peut obtenir des renseignements sur le serveur **sendmai**l, pour éviter cela, on peut rajouter la ligne suivante dans le fichier de config **config.mc**:

define(`confSMTP_LOGIN_MSG',`Serveur de mail; $b')dnl

Voilà ce que ça donne:

telnet zoulou 25
Trying 192.168.13.11...
Connected to zoulou.kervao.fr.
Escape character is '^]'.
220 Serveur ESMTP de mail; Sun, 17 Jun 2001 09:19:37 -0400

5 Lutte anti spam et anti virus

5.1 Présentation de la configuration

Depuis quelque temps, plus d'un email sur deux que je reçois est un spam ou contient un virus, la lutte anti spam et anti virus est donc devenue une nécessité car il devient vraiment pénible d'avoir sa boîte aux lettres pollué de "pourriels" en tout genre. La lutte est d'autant plus nécessaire si vous avez des utilisateurs sous windows qui sont des cibles privilégiées pour les virus.
Dans cette page sont présentées les outils des plus simples (et moins efficaces) aux plus sophistiquées (et plus efficaces mais moins faciles à mettre en place). On commencera par l'outil le plus simple:
- **procmail** permet de filtrer avec des règles statiques des emails qui ont été déjà délivrés par le serveur de mail local (MTA)
Les outils les plus sophistiqués sont
- **spamassassin** pour la lutte anti spam qui utilisent pour une meilleure efficacité trois autres outils du même genre à savoir **Razor**, **Pyzor** et **DCC**
- **clam anti virus** pour la lutte anti virus.
Ces deux derniers outils complètement interfaçables à **sendmail** ce qui permet un filtrage à la source des emails sur votre réseau local sans que l'utilisateur final n'ait à configurer quoi que ce soit.

En mettant en place tous ces outils je peux vous garantir que vous aurez un excellent taux de rejet de "pourriels".

5.2 Filtrage basique avec procmail

procmail est une commande simple qui permet de faire beaucoup de choses. Il est très simple de définir des filtres. L'exemple ci-dessous permet de supprimer les mails contenant dans le sujet I Love You

```
:0
 * ^Subject:.*ILOVEYOU
/dev/null
```

Il suffit d'adapter cette règle en fonction du sujet (ou du from). Cette autre règle très utile permet de sauvegarder dans un fichier **virus** tous les mails arrivant avec les extensions qui y sont citées.

```
:0 H
*^Content-type: (multipart/mixed)
{
:0 B
*^Content-Disposition: (attachment|inline)
*filename=".*\.(ocx|vbs|wsf|shs|exe|com|bat|chm|pif|vbe|hta|scr)"
{
:0
virus
}
}
```

Le fichier virus pourra être ouvert en tant que boîte aux lettres avec un logiciel comme kmail. Le filtre suivant

```
:0
^Subject:.*[^ -~][^ -~][^ -~][^ -~]
/dev/null
```

supprime tous les mails dont le sujet commence par 4 caractères consécutifs non ASCII (cas particulier des mails écrits en asiatiques). **Procmail** est quand même assez limité puisqu'il ne permet que de filtrer sur des règles précises (mot clef).

5.3 Filtrer les spams avec spamassassin

5.3.1 Présentation

Spamassassin est un logiciel anti spam, il repose entre autres sur l'analyse heuristique et bayesienne des emails et utilise d'autres outils anti spam comme **pyzor**, **razor** et **DCC** qui sont vus également dans cette page. D'abord quelques définitions.

5.3.2 Définitions

Le filtrage heuristique

C'est une technique qui permet d'identifier du spam en fonction de certaines caractéristiques communes (ponctuation, html, lien vers une image, ...)

Le filtrage bayésien

le filtrage bayésien repose sur le principe qu'un évènement peut se produire en fonction des mêmes évènements survenus précédemment. En clair pour le mail, si on rencontre certains mots ou phrases plus souvent dans du mail classé spam que dans du mail classé normal on peut penser que la prochaine fois qu'on rencontrera ces mêmes mots et phrases il y a de bonnes chances que ce soit dans un mail de spam.
Pour cela une base de données de mots et phrases est créée et enrichie au fur et à mesure de la réception et de l'envoi de mails qui soient valides ou considérés comme spam. Chaque mot ou sentence reçoit une valeur calculée en fonction de la probabilité qu'il soit relié à du spam, elle dépend du nombre de fois que le terme apparaît dans du spam par rapport au nombre de fois que le même terme est rencontré dans du courrier valide. Par conséquent certains mots pourront avoir une forte probabilité d'être rattaché à du spam pour certains utilisateurs et pour d'autres pas, exemple concret une entreprise travaillant dans le domaine médical le terme "drug" aura une faible probabilité d'être rattaché à du spam car il est très souvent employé dans les mails valides, pour d'autres personnes ce terme sera systématiquement rattaché à du spams. Par conséquent le filtre bayésien a la particularité et l'avantage de s'adapter à l'utilisateur, il réduit le risque des faux positifs (courrier valide considéré comme spam). Par ailleurs le filtre n'est pas statique, la base de donnée est en constante évolution et donc le filtre sera de plus en plus performant de jour en jour et s'adaptera en fonction des utilisateurs de votre réseau et des techniques nouvelles utilisées par les spammeurs.

Exemple concret du dernier point, jusqu'à présent les spammeurs envoyaient des mails avec des mots du style "sex, free, viagra, ...", il était assez simple de mettre en place un filtre basé sur des mots clef pour supprimer les mails en question, les spammeurs ont donc modifié légèrement la sémanthique de mots "s-e-x, f r e e" ou bien encore "viagr$a", avec un simple filtrage par mots clé, il est quasi impossible d'établir une règle efficace pour filtrer ces mails. Le filtre bayésien aura aucun problème pour lui attribuer une valeur de probabilité de spam élevée.

Autre avantage du filtre bayésien et non des moindres, il s'adapte à toutes les langues. En clair pour qu'un spammeur puisse tromper un filtre bayésien il doit connaître l'utilisateur qu'il veut toucher et éviter d'utiliser les mots que l'utilisateur en question utilise le moins...

5.3.3 Installation de razor

Razor repose sur le principe d'un serveur central qui identifie les spams en leur attribuant une signature digitale. Chaque utilisateur de **razor** attribue une signature digitale à chaque email reçu et la compare avec celles du serveur central, permettant ainsi le classement de l'email. Pour identifier les spammeurs, le serveur central diffuse largement des emails valides pour recevoir un max de spams (uniquement du spam, pas de mails valides pour éviter les faux positifs), plus il en reçoit meilleur est **razor** !

Maintenant on va récupérer **razor** qui va compléter **spamassassin** dans la recherche de spam sur cet URL razor.sourceforge.net/. On décompresse l'archive en tapant

tar xvfz razor-agents-2.84.tar.gz

Cela donne le répertoire **razor-agents-2.84**, avant d'aller plus loin il faudra installer en tant que root les modules de **perl** suivants

perl -MCPAN -e shell

Puis au prompt

install Digest::SHA1
install Digest::HMAC_MD5

Dans le répertoire **razor-agents-2.84** on tape maintenant

perl Makefile.PL

Puis

make

Puis en tant que root

make install

Voilà la trace de **razor2** en tapant **journalctl** avec le mode debug de **spamassassin**

oct. 10 08:16:14 mana.kervao.fr spamd[1672]: razor2: part=0 noresponse
oct. 10 08:16:14 mana.kervao.fr spamd[1672]: razor2: results: spam? 0
oct. 10 08:16:14 mana.kervao.fr spamd[1672]: razor2: results: engine 8, highest cf score: 0
oct. 10 08:16:14 mana.kervao.fr spamd[1672]: razor2: results: engine 4, highest cf score: 0

(...)

oct. 10 08:16:14 mana.kervao.fr spamd[1672]: timing: total 1249 ms - read_scoreonly_config: 0.16 (0.0%), signal_user_changed: 1.83 (0.1%), parse: 1.15 (0.1%), extract_message_metadata: 25 (2.0%), get_uri_detail_list: 3.0 (0.2%), tests_pri_-1000: 3.4 (0.3%), tests_pri_-950: 2.3 (0.2%), tests_pri_-900: 2.1 (0.2%), tests_pri_-400: 22 (1.8%), check_bayes: 20 (1.6%), b_tokenize: 10 (0.8%), b_tok_get_all: 4.0 (0.3%), b_comp_prob: 3.2 (0.3%), b_tok_touch_all: 0.31 (0.0%), b_finish: 1.50 (0.1%), tests_pri_0: 1134 (90.8%), check_spf: 0.65 (0.1%), check_dkim_signature: 0.58 (0.0%), check_dkim_adsp: 36 (2.8%), check_dcc: 149 (11.9%), check_pyzor: 171 (13.7%), check_razor2: 592 (47.4%), tests_pri_500: 4.9 (0.4%), tests_pri_1000: 3.2 (0.3%), total_awl: 0.73 (0.1%), rewrite_mail: 0.73 (0.1%), copy_config: 39 (3.1%)

J'en déduis que **razor2** marche correctement.

5.3.4 Installation de SpamAssassin

On récupérera **spamassassin** sur le site www.spamassassin.org/. On décompresse l'archive en tapant

tar xvfz Mail-SpamAssassin-3.4.1.tar.gz

Cela donne le répertoire **Mail-SpamAssassin-3.4.1**. Avant d'aller plus loin j'ai du installer les packages

perl-devel
perl-Net-DNS
perl-NetAddr-IP
perl-Archive-Tar
perl-Mail-SPF
perl-IP-Country
perl-Net-Ident
perl-IO-Socket-INET6
perl-Mail-DKIM
perl-DBI
perl-Encode-Detect
perl-Geo-IP

perl -MCPAN -e shell
install Net::CIDR::Lite
install Net::Patricia

Maintenant revenons dans le répertoire **Mail-SpamAssassin-3.4.1** où l'on tape

perl Makefile.PL

Voilà le résultat

What email address or URL should be used in the suspected-spam report text for users who want more information on your filter installation?

(In particular, ISPs should change this to a local Postmaster contact)
default text: [the administrator of that system] olivier@localhost

NOTE: settings for "make test" are now controlled using "t/config.dist".
See that file if you wish to customise what tests are run, and how.

checking module dependencies and their versions...
checking binary dependencies and their versions...

****
NOTE: the optional fetch binary is not installed.**

Sa-update will use curl, wget or fetch to download updates.
Because perl module LWP does not support IPv6, sa-update as of
3.4.0 will use these standard programs to download rule updates
leaving LWP as a fallback if none of the programs are found.

***IMPORTANT NOTE*: You only need one of these programs.**

dependency check complete...

optional binary missing or nonfunctional: fetch

warning: some functionality may not be available,
please read the above report before continuing!

Generating a Unix-style Makefile
Writing Makefile for Mail::SpamAssassin
Writing MYMETA.yml and MYMETA.json
Makefile written by ExtUtils::MakeMaker 6.98

Concernant le warning, pas d'inquiétude si vous avez **curl** ou **wget** installé sur votre système. On tape alors

make

Puis en tant que root

make install

5.3.5 Installation de DCC

DCC repose également sur un serveur central, chaque mail reçu reçoit une signature digitale, le serveur comptabilise toutes les signatures, plus le nombre d'une signature est élevé sur le serveur plus le risque que ce soit un spam est élevé.
Concrètement, à la réception d'un mail le client **DCC** lui attribue une signature digitale (checksum), récupére sur le serveur le nombre de fois que cette signature apparaît dans le serveur central, si ce nombre dépasse une certaine valeur configurable (threshold) et que l'expéditeur du mail en question n'est pas dans la whiteliste (configurable elle aussi), le mail est considéré comme spam et traité comme tel.

On va étoffer encore **SpamAssassin** avec **DCC** qu'on récupérera ici http://www.rhyolite.com/anti-spam/dcc/. On décompresse l'archive en tapant

tar xvfz dcc.tar.Z

Cela donne le répertoire **dcc-1.3.158**. Pensez maintenant à installer le package **sendmail-devel**, on revient dans le répertoire de **DCC** dans lequel on tape successivement

./configure
make

Puis en tant que root

make install

Maintenant si on tape

cdcc 'info'

On obtient

```
# 10/04/15 11:28:27 CEST  /var/dcc/map
# Re-resolve names after 13:28:15  Check RTTs after 11:43:26
# 1257.96 ms threshold, 1237.35 ms average    12 total, 11 working servers
IPv6 on   version=3

dcc1.dcc-servers.net,-      RTT+1000 ms  anon
#  74.92.232.243,-                              Etherboy ID 1002
#    100% of  1 requests ok  211.81+1000 ms RTT      100 ms queue wait
#  209.169.14.27,-                           x.dcc-servers ID 104
#    100% of  1 requests ok  254.24+1000 ms RTT      100 ms queue wait
#  209.169.14.30,-                           x.dcc-servers ID 104
#    100% of  1 requests ok  254.26+1000 ms RTT      100 ms queue wait

dcc2.dcc-servers.net,-      RTT+1000 ms  anon
#  67.66.138.141,-                                 ID 1356
#    100% of  1 requests ok  240.42+1000 ms RTT      100 ms queue wait
#  69.12.208.70,-                             sonic.net ID 1156
#    100% of  1 requests ok  280.20+1000 ms RTT      100 ms queue wait
#  194.119.212.6,-                              dcc1 ID 1182
#    100% of  1 requests ok  173.16+1000 ms RTT      100 ms queue wait

dcc3.dcc-servers.net,-      RTT+1000 ms  anon
#  38.124.232.176,-                              ID 1102
#    100% of  1 requests ok  207.19+1000 ms RTT      100 ms queue wait
#  209.169.14.29,-                           x.dcc-servers ID 104
#    100% of  1 requests ok  254.25+1000 ms RTT      100 ms queue wait

dcc4.dcc-servers.net,-      RTT+1000 ms  anon
#  192.135.10.194,-                            debian ID 1169
#    100% of  1 requests ok  162.00+1000 ms RTT      100 ms queue wait

dcc5.dcc-servers.net,-      RTT+1000 ms  anon
# *136.199.199.160,-                             URT ID 1060
#    100% of  1 requests ok  137.34+1000 ms RTT      100 ms queue wait
#  192.84.137.21,-                            INFN-TO ID 1233
#    100% of  1 requests ok  157.96+1000 ms RTT      100 ms queue wait

@,-                    RTT-1000 ms  32768
#  127.0.0.1,-
#      not answering

################
# 10/04/15 11:28:27 CEST  greylist /var/dcc/map
# Re-resolve names after 13:28:20
# 1 total, 0 working servers
# continue not asking greylist server 32 seconds after 1 failures

@,-                    Greylist 32768
# *127.0.0.1,6276
#      not answering
```

Le répertoire par défaut se trouve sous **/var/dcc**. les serveurs par défaut se trouvent dans le fichier **/var/dcc/map**, il est automatiquement créé à l'installation à partir du fichier **dcc-1.3.158/homedir/map.txt**. Dans ce dernier répertoire on trouve également le fichier **whitelist**, pour avoir la syntaxe il faut jeter un coup d'oeil dans le fichier **whitecommon**.

En upgradant d'une version précédente j'ai eu droit à l'erreur suivante

open(/var/dcc/map): Too many open files
open(/tmp/map1MdBL2): Too many open files
?

Je n'ai eu d'autres choix que de supprimer **/var/run/dcc** et de refaire un **make install**.

voilà la trace de **DCC** en tapant **journalctl** en mode debug de **spamassassin**

oct. 10 08:16:13 mana.kervao.fr spamd[1673]: dcc: dccproc responded with 'X-DCC-URT-Metrics: mana.kervao.fr 1060; Body=1 Fuz1=1 Fuz2=1_'
oct. 10 08:16:13 mana.kervao.fr spamd[1673]: rules: ran eval rule __DKIM_DEPENDABLE ======> got hit (1)
oct. 10 08:16:13 mana.kervao.fr spamd[1673]: check: tagrun - tag DCCB is now ready, value: URT
oct. 10 08:16:13 mana.kervao.fr spamd[1673]: check: tagrun - tag DCCR is now ready, value: mana.kervao.fr 1060; Body=1 Fuz1=1 Fuz2=1

(...)

oct. 10 08:16:09 mana.kervao.fr spamd[1672]: timing: total 6413 ms - read_scoreonly_config: 1.53 (0.0%), signal_user_changed: 2.2 (0.0%), parse: 1.96 (0.0%), extract_message_metadata: 55 (0.9%), get_uri_detail_list: 2.5 (0.0%), tests_pri_-1000: 50 (0.8%), tests_pri_-950: 2.00 (0.0%), tests_pri_-900: 2.1 (0.0%), tests_pri_-400: 19 (0.3%), check_bayes: 16 (0.3%), b_tokenize: 6 (0.1%), b_tok_get_all: 3.5 (0.1%), b_comp_prob: 3.8 (0.1%), b_tok_touch_all: 0.38 (0.0%), b_finish: 1.53 (0.0%), tests_pri_0: 6179 (96.4%), check_spf: 17 (0.3%), poll_dns_idle: 0.17 (0.0%), check_dkim_signature: 0.71 (0.0%), check_dkim_adsp: 30 (0.5%), check_dcc: 4302 (67.1%), check_pyzor: 170 (2.6%), check_razor2: 1599 (24.9%), tests_pri_500: 10 (0.2%), tests_pri_1000: 6 (0.1%), total_awl: 4.1 (0.1%), check_awl: 0.36 (0.0%), update_awl: 0.21 (0.0%), learn: 6 (0.1%), b_learn: 2.6 (0.0%), rewrite_mail: 1.73 (0.0%), get_report: 0.58 (0.0%), copy_config: 39 (0.6%)

5.3.6 Installation de pyzor

Pyzor repose sur le même principe que **razor**, c'est normal puisque le premier est issu du second et réécrit en python. Cela ne signifie pas qu'il marche exactement de la même manière en utilisant les mêmes serveurs, il a évolué différemment et les deux se complétent très bien.
pyzor s'interface également avec **spamassassin**, on le trouvera ici pyzor.sourceforge.net, on décompresse l'archive en tapant

tar xvfj pyzor-0.7.0.tar.bz2

Cela donne le répertoire **pyzor-0.7.0** dans lequel on tape

python setup.py build

il faudra sans doute installer préalablement le package **python-setuptools**

On passe root puis on tape

python setup.py install

voilà la trace de **pyzor** en tapant journalctl en mode debug de **spamassassin**

oct. 10 08:16:07 mana.kervao.fr spamd[1672]: pyzor: pyzor is available: /usr/bin/pyzor
oct. 10 08:16:07 mana.kervao.fr spamd[1672]: dns: entering helper-app run mode
oct. 10 08:16:07 mana.kervao.fr spamd[1672]: pyzor: opening pipe: /usr/bin/pyzor check < /tmp/.spamassassin167281DDAKtmp
oct. 10 08:16:07 mana.kervao.fr spamd[2520]: util: setuid: ruid=8 euid=8
oct. 10 08:16:07 mana.kervao.fr spamd[1672]: pyzor: [2520] finished: exit 1
oct. 10 08:16:07 mana.kervao.fr spamd[1672]: pyzor: got response: public.pyzor.org:24441 (200, 'OK') 0 0
oct. 10 08:16:07 mana.kervao.fr spamd[1672]: dns: leaving helper-app run mode

oct. 10 08:16:07 mana.kervao.fr spamd[1672]: check: tagrun - tag PYZOR is now ready, value: Reported 0 times.

(...)

oct. 10 08:16:15 mana.kervao.fr spamd[1673]: timing: total 1938 ms - read_scoreonly_config: 0.19 (0.0%), signal_user_changed: 2.9 (0.1%), parse: 1.58 (0.1%), extract_message_metadata: 13 (0.7%), get_uri_detail_list: 3.0 (0.2%), tests_pri_-1000: 3.7 (0.2%), tests_pri_-950: 2.4 (0.1%), tests_pri_-900: 1.81 (0.1%), tests_pri_-400: 22 (1.1%), check_bayes: 20 (1.0%), b_tokenize: 12 (0.6%), b_tok_get_all: 2.8 (0.1%), b_comp_prob: 1.96 (0.1%), b_tok_touch_all: 0.19 (0.0%), b_finish: 1.61 (0.1%), tests_pri_0: 1827 (94.3%), check_spf: 0.72 (0.0%), check_dkim_signature: 0.70 (0.0%), check_dkim_adsp: 43 (2.2%), check_dcc: 148 (7.7%), check_pyzor: 166 (8.6%), check_razor2: 1175 (60.6%), tests_pri_500: 4.1 (0.2%), tests_pri_1000: 2.7 (0.1%), total_awl: 0.76 (0.0%), rewrite_mail: 0.63 (0.0%), copy_config: 39 (2.0%)

5.3.7 Configuration de spamassassin

Revenons à **SpamAssassin**, les fichiers de configuration se trouvent sous **/etc/mail/spamassassin**, on trouve **init.pre** qui est d'abord lu, puis les fichiers spécifiques à des versions (**v310.pre**, **v312.pre**, **v320.pre**, **v330.pre, v340.pre et v341.pre**) à noter que tous les fichiers sont lus, d'une version à une autre on retrouve que les nouvelles fonctionnalités implémentées dans les dernières versions, et enfin le configuration local **local.cf**. A noter qu'on peut générer un fichier de config interactivement à la page www.yrex.com/spam/spamconfig.php.

Voilà le contenu du fichier **init.pre**

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'perldoc Mail::SpamAssassin::Conf' for details of what can be
# tweaked.
#
# This file contains plugin activation commands for plugins included
# in SpamAssassin 3.0.x releases.  It will not be installed if you
# already have a file in place called "init.pre".
#
# There are now multiple files read to enable plugins in the
# /etc/mail/spamassassin directory; previously only one, "init.pre" was
# read.  Now both "init.pre", "v310.pre", and any other files ending in
# ".pre" will be read.  As future releases are made, new plugins will be
# added to new files, named according to the release they're added in.
############################################################################

# RelayCountry - add metadata for Bayes learning, marking the countries
# a message was relayed through
#
# Note: This requires the Geo::IP Perl module
#
loadplugin Mail::SpamAssassin::Plugin::RelayCountry

# URIDNSBL - look up URLs found in the message against several DNS
# blocklists.
#
loadplugin Mail::SpamAssassin::Plugin::URIDNSBL

# Hashcash - perform hashcash verification.
#
loadplugin Mail::SpamAssassin::Plugin::Hashcash

# SPF - perform SPF verification.
#
loadplugin Mail::SpamAssassin::Plugin::SPF
```

Voilà mon fichier de configuration **v310.pre** qu'on trouvera sous **/etc/mail/spamassassin**

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'perldoc Mail::SpamAssassin::Conf' for details of what can be
# tweaked.
#
# This file was installed during the installation of SpamAssassin 3.1.0,
# and contains plugin loading commands for the new plugins added in that
# release.  It will not be overwritten during future SpamAssassin installs,
# so you can modify it to enable some disabled-by-default plugins below,
# if you so wish.
#
# There are now multiple files read to enable plugins in the
# /etc/mail/spamassassin directory; previously only one, "init.pre" was
# read.  Now both "init.pre", "v310.pre", and any other files ending in
# ".pre" will be read.  As future releases are made, new plugins will be
# added to new files, named according to the release they're added in.
############################################################################

# DCC - perform DCC message checks.
#
# DCC is disabled here because it is not open source.  See the DCC
# license for more details.
#
loadplugin Mail::SpamAssassin::Plugin::DCC

# Pyzor - perform Pyzor message checks.
#
loadplugin Mail::SpamAssassin::Plugin::Pyzor

# Razor2 - perform Razor2 message checks.
#
loadplugin Mail::SpamAssassin::Plugin::Razor2

# SpamCop - perform SpamCop message reporting
#
loadplugin Mail::SpamAssassin::Plugin::SpamCop

# AntiVirus - some simple anti-virus checks, this is not a replacement
# for an anti-virus filter like Clam AntiVirus
#
loadplugin Mail::SpamAssassin::Plugin::AntiVirus

# AWL - do auto-whitelist checks
#
loadplugin Mail::SpamAssassin::Plugin::AWL

# AutoLearnThreshold - threshold-based discriminator for Bayes auto-learning
#
loadplugin Mail::SpamAssassin::Plugin::AutoLearnThreshold

# TextCat - language guesser
#
#loadplugin Mail::SpamAssassin::Plugin::TextCat

# AccessDB - lookup from-addresses in access database
#
#loadplugin Mail::SpamAssassin::Plugin::AccessDB

# WhitelistSubject - Whitelist/Blacklist certain subject regular expressions
#
loadplugin Mail::SpamAssassin::Plugin::WhiteListSubject

############################################################################
```

```
# experimental plugins

# DomainKeys - perform DomainKeys verification
#
# This plugin has been removed as of v3.3.0.  Use the DKIM plugin instead,
# which supports both Domain Keys and DKIM.

# MIMEHeader - apply regexp rules against MIME headers in the message
#
loadplugin Mail::SpamAssassin::Plugin::MIMEHeader

# ReplaceTags
#
loadplugin Mail::SpamAssassin::Plugin::ReplaceTags
```

le fichier **v312.pre**

```
################################################
# experimental plugins

# DKIM - perform DKIM verification
#
# Mail::DKIM module required for use, see INSTALL for more information.
#
# Note that if C<Mail::DKIM> version 0.20 or later is installed, this
# renders the DomainKeys plugin redundant.
#
loadplugin Mail::SpamAssassin::Plugin::DKIM
```

le fichier **v320.pre**

```
################################################

# Check - Provides main check functionality
#
loadplugin Mail::SpamAssassin::Plugin::Check

# HTTPSMismatch - find URI mismatches between href and anchor text
#
loadplugin Mail::SpamAssassin::Plugin::HTTPSMismatch

# URIDetail - test URIs using detailed URI information
#
loadplugin Mail::SpamAssassin::Plugin::URIDetail

# Shortcircuit - stop evaluation early if high-accuracy rules fire
#
# loadplugin Mail::SpamAssassin::Plugin::Shortcircuit

# Plugins which used to be EvalTests.pm
# broken out into separate plugins
loadplugin Mail::SpamAssassin::Plugin::Bayes
loadplugin Mail::SpamAssassin::Plugin::BodyEval
loadplugin Mail::SpamAssassin::Plugin::DNSEval
loadplugin Mail::SpamAssassin::Plugin::HTMLEval
loadplugin Mail::SpamAssassin::Plugin::HeaderEval
loadplugin Mail::SpamAssassin::Plugin::MIMEEval
loadplugin Mail::SpamAssassin::Plugin::RelayEval
loadplugin Mail::SpamAssassin::Plugin::URIEval
loadplugin Mail::SpamAssassin::Plugin::WLBLEval

# VBounce - anti-bounce-message rules, see rules/20_vbounce.cf
```

```
#
loadplugin Mail::SpamAssassin::Plugin::VBounce

# Rule2XSBody - speedup by compilation of ruleset to native code
#
# loadplugin Mail::SpamAssassin::Plugin::Rule2XSBody

# ASN - Look up the Autonomous System Number of the connecting IP
# and create a header containing ASN data for bayes tokenization.
# See plugin's POD docs for usage info.
#
# loadplugin Mail::SpamAssassin::Plugin::ASN

# ImageInfo - rules to match metadata of image attachments
#
loadplugin Mail::SpamAssassin::Plugin::ImageInfo
```

le fichier **v330.pre**

```
##################################################

# PhishTag - allows sites to rewrite suspect phish-mail URLs
# (Note: this requires configuration, see http://umut.topkara.org/PhishTag)
#
#loadplugin Mail::SpamAssassin::Plugin::PhishTag

# FreeMail - detect email addresses using free webmail services,
# usable as input for other rules
#
loadplugin Mail::SpamAssassin::Plugin::FreeMail
```

le fichier **v340.pre**

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'perldoc Mail::SpamAssassin::Conf' for details of what can be
# tweaked.
#
# This file was installed during the installation of SpamAssassin 3.4.0,
# and contains plugin loading commands for the new plugins added in that
# release.  It will not be overwritten during future SpamAssassin installs,
# so you can modify it to enable some disabled-by-default plugins below,
# if you so wish.
#
# There are now multiple files read to enable plugins in the
# /etc/mail/spamassassin directory; previously only one, "init.pre" was
# read.  Now both "init.pre", "v310.pre", and any other files ending in
# ".pre" will be read.  As future releases are made, new plugins will be
# added to new files, named according to the release they're added in.
#############################################################################

# AskDNS - forms a DNS query based on 'tags' as supplied by other plugins
#
loadplugin Mail::SpamAssassin::Plugin::AskDNS
```

le fichier **v341.pre**

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'perldoc Mail::SpamAssassin::Conf' for details of what can be
# tweaked.
```

```
#
# This file was installed during the installation of SpamAssassin 3.4.1,
# and contains plugin loading commands for the new plugins added in that
# release.  It will not be overwritten during future SpamAssassin installs,
# so you can modify it to enable some disabled-by-default plugins below,
# if you so wish.
#
# There are now multiple files read to enable plugins in the
# /etc/mail/spamassassin directory; previously only one, "init.pre" was
# read.  Now both "init.pre", "v310.pre", and any other files ending in
# ".pre" will be read.  As future releases are made, new plugins will be
# added to new files, named according to the release they're added in.
############################################################################

# TxRep - Reputation database that replaces AWL
# loadplugin Mail::SpamAssassin::Plugin::TxRep

# URILocalBL - Provides ISP and Country code based filtering as well as
# quick IP based blocks without a full RBL implementation - Bug 7060

# loadplugin Mail::SpamAssassin::Plugin::URILocalBL

# PDFInfo - Use several methods to detect a PDF file's ham/spam traits
# loadplugin Mail::SpamAssassin::Plugin::PDFInfo
```

et voilà mon fichier **local.cf**

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'perldoc Mail::SpamAssassin::Conf' for details of what can be
# tweaked.
#
# Only a small subset of options are listed below
#
###########################################################################

#   Add *****SPAM***** to the Subject header of spam e-mails
#
# rewrite_header Subject *****SPAM*****


#   Save spam messages as a message/rfc822 MIME attachment instead of
#   modifying the original message (0: off, 2: use text/plain instead)
#
# report_safe 1


#   Set which networks or hosts are considered 'trusted' by your mail
#   server (i.e. not spammers)
#
# trusted_networks 212.17.35.


#   Set file-locking method (flock is not safe over NFS, but is faster)
#
# lock_method flock


#   Set the threshold at which a message is considered spam (default: 5.0)
#
required_score 4.0
```

```
#   Use Bayesian classifier (default: 1)
#
use_bayes 1


#   Bayesian classifier auto-learning (default: 1)
#
bayes_auto_learn 1


#   Set headers which may provide inappropriate cues to the Bayesian
#   classifier
#
# bayes_ignore_header X-Bogosity
# bayes_ignore_header X-Spam-Flag
# bayes_ignore_header X-Spam-Status

bayes_path /var/spool/mail/.spamassassin/bayes
bayes_file_mode 0777

#   Whether to decode non- UTF-8 and non-ASCII textual parts and recodeA
#
#   them to UTF-8 before the text is given over to rules processing.
#
# normalize_charset 1

#   Some shortcircuiting, if the plugin is enabled
#
ifplugin Mail::SpamAssassin::Plugin::Shortcircuit
#
#   default: strongly-whitelisted mails are *really* whitelisted now, if the
#   shortcircuiting plugin is active, causing early exit to save CPU load.
#   Uncomment to turn this on
#
# shortcircuit USER_IN_WHITELIST       on
# shortcircuit USER_IN_DEF_WHITELIST   on
# shortcircuit USER_IN_ALL_SPAM_TO     on
# shortcircuit SUBJECT_IN_WHITELIST    on

#   the opposite; blacklisted mails can also save CPU
#
# shortcircuit USER_IN_BLACKLIST       on
# shortcircuit USER_IN_BLACKLIST_TO    on
# shortcircuit SUBJECT_IN_BLACKLIST    on

#   if you have taken the time to correctly specify your "trusted_networks",
#   this is another good way to save CPU
#
# shortcircuit ALL_TRUSTED             on

#   and a well-trained bayes DB can save running rules, too
#
# shortcircuit BAYES_99                spam
# shortcircuit BAYES_00                ham

endif # Mail::SpamAssassin::Plugin::Shortcircuit
```

A noter le chemin **bayes_path** on doit mettre ici le chemin où sera stockée la base de données du filtrage bayésien, comme c'est l'utilisateur **mail** qui est propriétaire du process et que sa homedirectory est fixée à **/var/spool/mail**, j'ai fixé la variable pour pointer vers cet endroit. Si vous voulez placer ces fichiers ailleurs, n'oubliez pas que l'utilisateur **mail** (ou celui propriétaire du process) doit avoir les droits en accés et écriture sur le répertoire en question.

5.3.8 Prise en compte des spams

Voilà deux techniques pour prendre en compte les spams

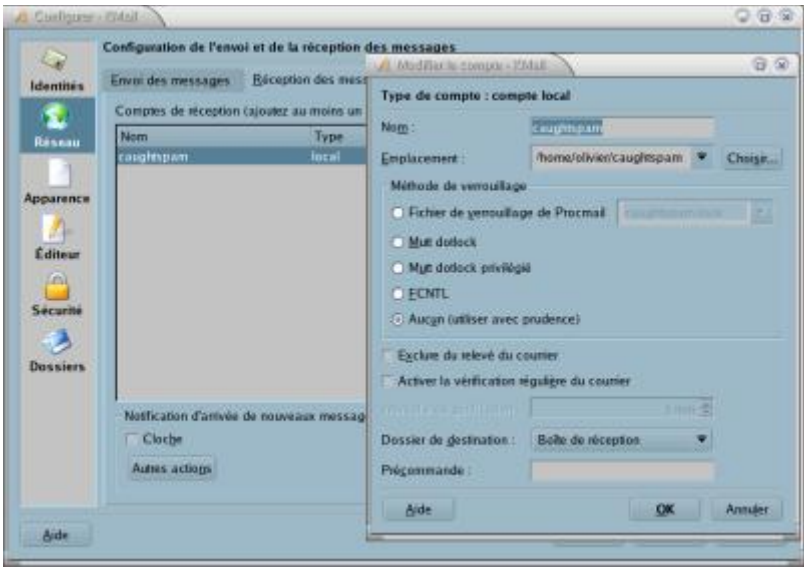
Technique Procmail

Maintenant j'ai créé un fichier **.procmailrc** sous ma homedirectory qui contient

:0fw: spamassassin.lock
*** < 256000**
| spamassassin

:0:
*** ^X-Spam-Status: Yes**
caughtspam

Tous les fichiers ayant une taille inférieur à 256000 octets passent à la moulinette **SpamAssassin**, car la plupart des spams ne dépassent pas cette taille, ceux qui sont considérés comme spams sont sauvegardés dans le fichier **caughtspam**. Maintenant j'ai configuré **kmail** pour qu'il puisse lire ce fichier (capture d'écran ci-dessous)



Technique Thunderbird

Vous ne créez pas ce fichier **.procmailrc** par contre dans la configuration de votre compte dans le paramétrage des indésirables.

Il faut bien prendre soin de cocher la case "**Se fier aux en-têtes de courrier indésirable placés par SpamAssassin**".

Dans ce cas, les mails classés spams se retrouveront automatiquement dans le dossier "indésirables". Dans le cas où un mail est faussement classé comme spam, vous devez l'exporter au formal eml dans un répertoire particulier **fauxspams**, de la même manière un spam non détecté doit être exporté au format eml dans un répertoire marqué **spams**.

Pour les deux techniques

Pour tous vos utilisateurs il faudra créer un répertoire **.spamassassin** dans chaque homedirectory, ce répertoire **.spamassassin** doit avoir les droits 777 pour que l'utilisateur mail puisse y avoir accès.
Maintenant si un mail venant d'un expéditeur particulier (les sites de vente par internet par exemple avec leur email bourré de HTML) est considéré comme spam alors qu'il ne devrait pas l'être. Rajoutez dans le fichier **/home/user/.spamassasin/user_prefs** la ligne

whitelist_from *@ldlc.fr *@rueducommerce.com *@fnac.com

Dans ce répertoire on retrouve un fichier de préférence local qui a pour nom **user_prefs**. Plus d'info sur le fichier de configuration de **SpamAssassin** à cet endroit
http://spamassassin.org/doc/Mail_SpamAssassin_Conf.html

5.3.9 Interfaçage avec sendmail

Spamassassin s'interface facilement avec **sendmail**, le filtrage s'opère aussi bien à la réception qu'à l'envoi d'email de manière totalement transparente ou presque, car il faut savoir que **spamassassin** est assez gourmand en ressource et que ça ralentit beaucoup la réception et l'envoi de mails.

On va récupérer ensuite une "rustine" pour que **spamassassin** puisse s'interfacer avec **sendmail** sur le site http://savannah.nongnu.org/projects/spamass-milt. Avant d'aller plus loin, il faudra installer le package **sendmail-devel** si ce n'est pas déjà fait. On décompresse l'archive en tapant

tar xvfz spamass-milter-0.4.0.tar.gz

Cela donne le répertoire **spamass-milter-0.4.0** dans le quel on tape successivement

./configure
make

Puis en tant que root

make install

Tout d'abord on doit lancer le daemon **spamassassin** en tant que root

spamd -d -D -u mail -H /var/spool/mail

-d mode daemon
-D mode debug (optionnel, utile au tout début)
-u mail l'utilisateur mail sera propriétaire du process.
-H le répertoire où se trouve la base de données bayésienne

Toujours en tant que root on lance **spamass-milter** en tapant

spamass-milter -u mail -p /var/run/spamass.sock -f

On modifie à présent le fichier de configuration de **sendmail**, en supposant qu'il soit sous **/usr/share/sendmail-cf/cf** et qu'il s'appelle **config.mc**

cd /usr/share/sendmail-cf/cf

On rajoute à la fin les lignes suivantes

INPUT_MAIL_FILTER(`spamassassin', `S=local:/var/run/spamass.sock, F=, T=C:15m;S:4m;R:4m;E:10m')dnl

define(`confMILTER_MACROS_CONNECT',`t, b, j, _, {daemon_name}, {if_name}, {if_addr}')dnl
define(`confMILTER_MACROS_HELO',`s, {tls_version}, {cipher}, {cipher_bits}, {cert_subject}, {cert_issuer}')dnl
define(`confMILTER_MACROS_ENVFROM',`i, {auth_authen}, {auth_type}')dnl
define(`confMILTER_MACROS_ENVRCPT',`r, v, Z')dnl

on génère un nouveau fichier de configuration de **sendmail** en tapant

m4 config.mc > /etc/mail/sendmail.cf

au niveau de

 # Milter options
#O Milter.LogLevel
O Milter.macros.connect=t, b, j, _, {daemon_name}, {if_name}, {if_addr}
O Milter.macros.helo=s, {tls_version}, {cipher}, {cipher_bits}, {cert_subject}, {cert_issuer}
O Milter.macros.envfrom=i, {auth_authen}, {auth_type}
O Milter.macros.envrcpt=r, v, Z
O Milter.macros.eom={msg_id}
#O Milter.macros.eoh
#O Milter.macros.data

on modifie ainsi

 # Milter options
#O Milter.LogLevel
O Milter.macros.connect=t, b, j, _, {daemon_name}, {if_name}, {if_addr}
O Milter.macros.helo=s, {tls_version}, {cipher}, {cipher_bits}, {cert_subject}, {cert_issuer}
O Milter.macros.envfrom=i
O Milter.macros.envfrom=i, {auth_authen}, {auth_type}
O Milter.macros.envrcpt=r, v, Z
O Milter.macros.eom={msg_id}
#O Milter.macros.eoh
#O Milter.macros.data

cette modification est utile pour éviter ce genre d'erreur

spamass-milter[2246]: Could not retrieve sendmail macro "i"!. Please add it to confMILTER_MACROS_ENVFROM for better spamassassin results

On relance **sendmail**

systemctl stop sendmail
systemctl start sendmail

5.3.10 Lancement automatique

Pour un lancement automatique de **spamd** et **spamass-milter** on créera le fichier **spamassassin.service** qu'on place sous **/usr/lib/systemd/system/** voilà son contenu
[Unit]
Description=Spamassassin daemon
After=syslog.target network.target

[Service]
Type=forking
ExecStart=/usr/local/bin/spamd -d -D -u mail -H /var/spool/mail --pidfile /var/run/spamd.pid

[Install]
WantedBy=multi-user.target

à noter que l'option **-D** correspond au mode debug facultatif, maintenant pour que le service soit lancé à chaque

boot de la machine il faudra taper

systemctl enable spamassassin.service

voillà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/spamassassin.service to /usr/lib/systemd/system/spamassassin.service.

pour le lancer il suffit maintenant de taper

systemctl start spamassassin.service

A noter que si lors d'un upgrade vous obtenez l'erreur suivante avec la commande suivante

oct. 04 20:59:54 mana.kervao.fr spamd[1634]: config: no rules were found! Do you need to run 'sa-update'?
oct. 04 20:59:55 mana.kervao.fr spamd[1516]: child process [1634] exited or timed out without signaling production of a PID file: exit 25...ne 2989.

il faudra penser à taper d'abord **sa-update** voilà ce que donne la commande suivante **systemctl status spamassassin**

● spamassassin.service - Spamassassin daemon
Loaded: loaded (/usr/lib/systemd/system/spamassassin.service; enabled)
Active: active (running) since dim. 2015-10-04 21:18:02 CEST; 8s ago
Process: 1982 ExecStart=/usr/local/bin/spamd -d -D -u mail -H /var/spool/mail --pidfile /var/run/spamd.pid (code=exited, status=0/SUCCESS)
Main PID: 1984 (spamd)
CGroup: /system.slice/spamassassin.service
├─1984 /usr/bin/perl -T -w /usr/local/bin/spamd -d -D -u mail -H /var/spool/mail --pidfile /var/run/spamd.pid
├─1988 spamd child
└─1989 spamd child

oct. 04 21:18:02 mana.kervao.fr spamd[1989]: plugin: Mail::SpamAssassin::Plugin::Bayes=HASH(0x24e3858) implements 'spamd_child_init', priority 0
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child 1988: entering state 1
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: new lowest idle kid: 1988
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child reports idle
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child states: IS
oct. 04 21:18:02 mana.kervao.fr spamd[1989]: prefork: sysread(12) not ready, wait max 300.0 secs
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child 1989: entering state 1
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: new lowest idle kid: 1988
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child reports idle
oct. 04 21:18:02 mana.kervao.fr spamd[1984]: prefork: child states: II

passons maintenant à **spamass-milter** on va créer le fichier **spamass-milter.service** sous **/usr/lib/systemd/system** voici son contenu

[Unit]
Description = Mail filter for SpamAssassin
Wants = spamassassin.service
After = syslog.target local-fs.target network.target remote-fs.target nss-lookup.target spamassassin.service
Before = sendmail.service

[Service]
Type = simple
ExecStart = /usr/local/sbin/spamass-milter -u mail -p /var/run/spamass.sock

[Install]
WantedBy = multi-user.target

maintenant pour que le service soit lancé à chaque boot de la machine il faudra taper

systemctl enable spamass-milter.service

voillà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/spamass-milter.service to /usr/lib/systemd/system/spamass-milter.service.

pour le lancer il suffit maintenant de taper

systemctl start spamass-milter.service

voilà le résultat de la commande **systemctl status spamass-milter.service**

● spamass-milter.service - Mail filter for SpamAssassin
Loaded: loaded (/usr/lib/systemd/system/spamass-milter.service; enabled)
Active: active (running) since dim. 2015-10-04 21:30:36 CEST; 5s ago
Main PID: 2274 (spamass-milter)
CGroup: /system.slice/spamass-milter.service
└─2274 /usr/local/sbin/spamass-milter -d 3 -u mail -p /var/run/spamass.sock

oct. 04 21:30:36 mana.kervao.fr spamass-milter[2274]: Setting debug level to 0x3f
oct. 04 21:30:36 mana.kervao.fr spamass-milter[2274]: smfi_register succeeded
oct. 04 21:30:36 mana.kervao.fr spamass-milter[2274]: spamass-milter 0.4.0 starting

5.3.11 Fonctionnement

Voilà c'est fait, plus besoin de modifier le fichier **.procmailrc** (il peut être vide) **sendmail** va s'en charger pour vous en amont.

J'ai eu un soucis quand je récupérais les mails de mes utilisateurs, l'utilisateur mail ne pouvant créer de fichier dans leur homedirectory

Oct 2 09:58:53 tosh spamd[1676]: debug: open of AWL file failed: lock: 1676 cannot create tmp lockfile /home/olivier/.spamassassin/auto-whitelist.lock.tosh.kervao.fr.1676 for /home/olivier/.spamassassin/auto-whitelist.lock: Permission denied
Pour résoudre cela, le répertoire **.spamassassin** de tous mes utilisateurs (à créer éventuellement) appartient à l'utilisateur mail ou de mettre les droits à 777. Pour info AWL (autowhitelist) permet de mettre un score aux adresses email que vous utilisez le plus pour mieux démarquer vos interlocuteurs habituels et les autres. l'AWL est activé par défaut, pour le désactiver dans le fichier **/etc/mail/spamassassin/local.cf** il suffit de rajouter la ligne

use_auto_whitelist 0

Vous trouverez ici une idée de ce que donne **journalctl -f** avec un lancement en mode debug du daemon **spamd** à la réception d'un spam. A noter qu'il faut au moins 200 spams dans la base de données bayésienne pour que le filtre puisse fonctionner. Dans le cas de la mise à jour, vous pouvez très bien récupérer vos fichiers **bayes_seen** et **bayes_toks** et les placer sous **/var/spool/mail/.spamassassin** avant lancement du daemon.
Voilà ce que ça donne en réception d'un spam (mode non debug)

oct. 10 08:26:06 mana.kervao.fr spamd[4025]: logger: removing stderr method
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: spamd: server started on IO::Socket::IP [::1]:783, IO::Socket::IP [127.0.0.1]:783 (running version 3.4.1)
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: spamd: server pid: 4027
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: spamd: server successfully spawned child process, pid 4031
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: spamd: server successfully spawned child process, pid 4032
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: prefork: child states: IS
oct. 10 08:26:10 mana.kervao.fr spamd[4027]: prefork: child states: II
oct. 10 08:26:30 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: from=<Benkyi@hinet.net>, size=4903, class=0, nrcpts=1, msgid=<DE97D6B6.9CC36D3B@hinet.net>, bodytype=7BIT, proto=ESMTP, daemon=MTA, relay=localhost [127.0.0.1]
oct. 10 08:26:30 mana.kervao.fr spamd[4031]: spamd: connection from mana.kervao.fr [::1]:49009 to port 783, fd 5
oct. 10 08:26:30 mana.kervao.fr spamd[4031]: spamd: processing message

<DE97D6B6.9CC36D3B@hinet.net> for olivier:8
oct. 10 08:26:32 mana.kervao.fr spamd[4031]: spamd: identified spam (30.2/4.0) for olivier:8 in 2.3 seconds, 5266 bytes.
oct. 10 08:26:32 mana.kervao.fr spamd[4031]: spamd: result: Y 30 - BAYES_99,BAYES_999,CK_HELO_DYNAMIC_SPLIT_IP,CK_HELO_GENERIC,GOOG_REDIR_NO RDNS,HELO_DYNAMIC_IPADDR2,HTML_MESSAGE,HTTP_EXCESSIVE_ESCAPES,IMPOTENCE ,MIME_HTML_ONLY,RCVD_IN_BRBL_LASTEXT,RCVD_IN_PBL,RCVD_IN_RP_RNBL,RCVD_IN_ XBL,RDNS_NONE,TVD_RCVD_SPACE_BRACKET,T_ANY_PILL_PRICE,UNPARSEABLE_RELAY, URIBL_BLACK,URIBL_DBL_SPAM,URIBL_JP_SURBL,URIBL_SBL,URIBL_SBL_A,URIBL_WS_SU RBL
scantime=2.3,size=5266,user=olivier,uid=8,required_score=4.0,rhost=mana.kervao.fr,raddr=::1,rport=490 09,mid=<DE97D6B6.9CC36D3B@hinet.net>,bayes=1.000000,autolearn=unavailable autolearn_force=no
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter add: header: X-Spam-Flag: YES
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter add: header: X-Spam-Status: Yes, score=30.2 required=4.0 tests=BAYES_99,BAYES_999,\n\tCK_HELO_DYNAMIC_SPLIT_IP,CK_HELO_GENERIC,GOOG_RE DIR_NORDNS,\n\tHELO_DYNAMIC_IPADDR2,HTML_MESSAGE,HTTP_EXCESSIVE_ESCAPES,I MPOTENCE,\n\tMIME_HTML_ONLY,RCVD_IN_BRBL_LASTEXT,RCVD_IN_PBL,RCVD_IN_RP_R NBL,RCVD_IN_XBL,\n\tRDNS_NONE,TVD_RCVD_SPACE_BRACKET,T_ANY_PILL_PRICE,UNPA RSEABLE_RELAY,\n\tURIBL_BLACK,URIBL_DBL_SPAM,URIBL_JP_SURBL,URIBL_SBL,URIBL_ SBL_A,\n\tURIBL_WS_SURBL autolearn=unavailable autolearn_force=no version=3.4.1
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter add: header: X-Spam-Level: ******************************
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter add: header: X-Spam-Checker-Version: SpamAssassin 3.4.1 (2015-04-28) on mana.kervao.fr
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter change: header Content-Type: from text/html;\n\tcharset="iso-8859-1" to multipart/mixed; boundary="---------- =_5618AF98.65DFB357"
oct. 10 08:26:32 mana.kervao.fr sendmail[4042]: t9A6QUn6004042: Milter message: body replaced
oct. 10 08:26:32 mana.kervao.fr spamd[4027]: prefork: child states: II
oct. 10 08:26:32 mana.kervao.fr sendmail[4047]: t9A6QUn6004042: to=<olivier@localhost>, delay=00:00:02, xdelay=00:00:00, mailer=local, pri=42382, dsn=2.0.0, stat=Sent

Et voilà une partie du corps d'un spam après traitement par **SpamAssassin**.

From Benkyi@hinet.net Sat Oct 10 08:26:32 2015
Return-Path: <Benkyi@hinet.net>
Received: from fai.fr
by fai.fr (8.15.1/8.15.1/Sendmail de FUNIX) with ESMTP id t9A6QUn6004042
for <olivier@localhost>; Sat, 10 Oct 2015 08:26:30 +0200
Delivered-To: <olivier.hoarau@funix.org>
Received: from pop.online.net [62.210.16.34]
by mana.kervao.fr with POP3 (fetchmail-6.3.26)
for <olivier@localhost> (single-drop); Sat, 10 Oct 2015 08:26:30 +0200 (CEST)
Received: from exim-proxy-1.online.net ([10.42.2.124])
by exim-backend-17.online.net (Dovecot) with LMTP id +YoSKz+VGFY8GgAAi1FXbQ
for <olivier.hoarau@funix.org>; Sat, 10 Oct 2015 06:37:00 +0200
Received: from [10.42.2.120] (helo=111-253-90-139.dynamic.hinet.net)
by exim-proxy-1.online.net with esmtp (Exim 4.76)
(envelope-from <Benkyi@hinet.net>)
id 1Zklu0-0006fE-Dl
for olivier.hoarau@funix.org; Sat, 10 Oct 2015 06:37:00 +0200
Received: from 111-253-90-139.dynamic.hinet.net ([111.253.90.139])
by mx-vit.online.net (MXproxy) for olivier.hoarau@funix.org ;
Sat, 10 Oct 2015 06:16:06 +0200 (CEST)
X-ProXaD-SC: state=SPAM score=300
Received: from snmp.otwaloow.com ([Fri, 09 Oct 2015 20:36:37 -0700])
by snmp.otwaloow.com with ASMTP; Fri, 09 Oct 2015 20:36:37 -0700
Message-ID: <DE97D6B6.9CC36D3B@hinet.net>
Date: Fri, 09 Oct 2015 20:36:37 -0700
Reply-To: "Shirley" <Benkyi@hinet.net>

From: "Shirley" <Benkyi@hinet.net>
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.6a) Gecko/20031030
X-Accept-Language: en-us
MIME-Version: 1.0
To: "Shirley" <olivier.hoarau@funix.org>
Subject: Perfect sexual activity and longevity can be achieved with the help of these nice little pills!
Content-Type: multipart/mixed; boundary="----------=_5618AF98.65DFB357"
Content-Transfer-Encoding: 7bit
X-online-proxy-in: FLEX_NE9LczhXR0lPeVN5c3lzaw==uxDrwFiP0cgbr1/ulm28FoeUko7Nz/te
X-online-to: olivier.hoarau@funix.org
X-original-for: olivier.hoarau@funix.org
X-Spam-Flag: YES
X-Spam-Status: Yes, score=30.2 required=4.0 tests=BAYES_99,BAYES_999,
CK_HELO_DYNAMIC_SPLIT_IP,CK_HELO_GENERIC,GOOG_REDIR_NORDNS,
HELO_DYNAMIC_IPADDR2,HTML_MESSAGE,HTTP_EXCESSIVE_ESCAPES,IMPOTENCE,

MIME_HTML_ONLY,RCVD_IN_BRBL_LASTEXT,RCVD_IN_PBL,RCVD_IN_RP_RNBL,RCVD_IN_XBL,
RDNS_NONE,TVD_RCVD_SPACE_BRACKET,T_ANY_PILL_PRICE,UNPARSEABLE_RELAY,
URIBL_BLACK,URIBL_DBL_SPAM,URIBL_JP_SURBL,URIBL_SBL,URIBL_SBL_A,
URIBL_WS_SURBL autolearn=unavailable autolearn_force=no version=3.4.1
X-Spam-Level: ****************************
X-Spam-Checker-Version: SpamAssassin 3.4.1 (2015-04-28) on mana.kervao.fr

This is a multi-part message in MIME format.

----------=_5618AF98.65DFB357
Content-Type: text/plain; charset=iso-8859-1
Content-Disposition: inline
Content-Transfer-Encoding: 8bit

---------- Début de Rapport SpamAssassin ----------
Ce message est probablement du SPAM (message non sollicité envoyé en masse, publicité, escroquerie...).

Cette notice a été ajoutée par le système d'analyse "SpamAssassin" sur votre serveur de courrier "mana.kervao.fr", pour vous aider à identifier ce type de messages.

Le système SpamAssassin ajoute un en-tête "X-Spam-Flag: YES" aux messages qu'il considère comme étant probablement du Spam. Vous pouvez si vous le souhaitez utiliser cette caractéristique pour régler un filtre dans votre logiciel de lecture de courrier, afin de détruire ou de classer à part ce type de message.

Si ce robot a classifié incorrectement un message qui vous était destiné, ou pour toute question, veuillez contacter l'administrateur du système par e-mail à olivier@localhost .

Voir http://spamassassin.apache.org/tag/ pour plus de détails (en anglais).

Détails de l'analyse du message: (30.2 points, 4.0 requis)
3.5 BAYES_99 BODY: L'algorithme Bayésien a évalué la probabilité de spam entre 99 et 100% [score: 1.0000]
0.2 CK_HELO_GENERIC Relay used name indicative of a Dynamic Pool or Generic rPTR
1.5 CK_HELO_DYNAMIC_SPLIT_IP Relay HELO'd using suspicious hostname (Split IP)
0.0 TVD_RCVD_SPACE_BRACKET No description available.
1.6 URIBL_WS_SURBL Contains an URL listed in the WS SURBL blocklist [URIs: purecuringcompany.ru]
1.2 URIBL_JP_SURBL Contains an URL listed in the JP SURBL blocklist

[URIs: purecuringcompany.ru]
1.3 RCVD_IN_RP_RNBL RBL: Relay in RNBL,
https://senderscore.org/blacklistlookup/
[111.253.90.139 listed in bl.score.senderscore.com]
2.5 URIBL_DBL_SPAM Contains a spam URL listed in the DBL blocklist
[URIs: purecuringcompany.ru]
1.7 URIBL_BLACK Contains an URL listed in the URIBL blacklist
[URIs: purecuringcompany.ru]
3.3 RCVD_IN_PBL RBL: Received via a relay in Spamhaus PBL
[111.253.90.139 listed in zen.spamhaus.org]
0.4 RCVD_IN_XBL RBL: Received via a relay in Spamhaus XBL
1.4 IMPOTENCE BODY: Prétend permettre de combattre l'impuissance
1.6 HTTP_EXCESSIVE_ESCAPES URI: URI: Contient des %-escapes nombreux et
superflus
0.7 MIME_HTML_ONLY BODY: Le message possède uniquement des parties MIME
text/html
0.0 HTML_MESSAGE BODY: HTML inclus dans le message
0.2 BAYES_999 BODY: L'algorithme Bayésien a évalué la probabilité de spam
entre 99.9 et 100%
[score: 1.0000]
1.4 RCVD_IN_BRBL_LASTEXT RBL: No description available.
[111.253.90.139 listed in bb.barracudacentral.org]
0.1 URIBL_SBL_A Contains URL's A record listed in the SBL blocklist
[URIs: purecuringcompany.ru]
1.6 URIBL_SBL Contains an URL's NS IP listed in the SBL blocklist
[URIs: purecuringcompany.ru]
3.6 HELO_DYNAMIC_IPADDR2 Relay HELO'd using suspicious hostname (IP addr
2)
0.0 UNPARSEABLE_RELAY Informational: message has unparseable relay lines
0.8 RDNS_NONE Delivered to internal network by a host with no rDNS
1.4 GOOG_REDIR_NORDNS Google redirect to obscure spamvertised website +
no rDNS
0.0 T_ANY_PILL_PRICE Prices for pills

---------- Fin de Rapport SpamAssassin ----------

Chaque mail reçoit les informations suivantes en en tête du style

X-Spam-Flag: YES
X-Spam-Status: Yes, score=30.2 required=4.0 tests=BAYES_99,BAYES_999,
CK_HELO_DYNAMIC_SPLIT_IP,CK_HELO_GENERIC,GOOG_REDIR_NORDNS,
HELO_DYNAMIC_IPADDR2,HTML_MESSAGE,HTTP_EXCESSIVE_ESCAPES,IMPOTENCE,

MIME_HTML_ONLY,RCVD_IN_BRBL_LASTEXT,RCVD_IN_PBL,RCVD_IN_RP_RNBL,RCVD_IN_
XBL,
RDNS_NONE,TVD_RCVD_SPACE_BRACKET,T_ANY_PILL_PRICE,UNPARSEABLE_RELAY,
URIBL_BLACK,URIBL_DBL_SPAM,URIBL_JP_SURBL,URIBL_SBL,URIBL_SBL_A,
URIBL_WS_SURBL autolearn=unavailable autolearn_force=no version=3.4.1
X-Spam-Level: ****************************
X-Spam-Checker-Version: SpamAssassin 3.4.1 (2015-04-28) on mana.kervao.fr

A présent vous devez aider **spamassassin** à identifier les spams en lui indiquant les mails qui auraient du être qualifiés de spam et ceux qui n'auraient pas du être identifiés comme spam. Avec la technique **procmail**/**kmail** qui me sert à lire la boîte aux lettres de spam, j'ai créé un dossier **fauxspams** où je déplace les mails qui n'auraient pas du être classés comme spams.
Maintenant la commande à taper en tant que root pour qu'il ne prenne plus en compte les mails comme spams est la suivante

sa-learn --ham --dir /home/olivier/.Mail/fauxspam/cur

Avec la technique **thunderbird** qui me sert à lire la boîte aux lettres des mails normaux, j'ai créé deux répertoires **spams** et **fauxspams** dans lequel je déplace les mails qui auraient du être classés comme spam.

Maintenant la commande à taper en tant que root pour qu'il ne prenne plus en compte les mails comme spams est la suivante

/usr/local/bin/sa-learn --ham --dir /export/home/user/fauxspams
/usr/local/bin/sa-learn --spam --dir /export/home/user/spams

Voilà le résultat

Learned from 3 message(s) (3 message(s) examined).

Pour automatiser tout cela vous pouvez créer le fichir **/etc/cron.daily/bayes** contenant

#!/bin/bash
/usr/local/bin/sa-learn --ham --dir /export/home/user1/fauxspams
/usr/local/bin/sa-learn --spam --dir /export/home/user1/spams
/usr/local/bin/sa-learn --ham --dir /export/home/user2/fauxspams
/usr/local/bin/sa-learn --spam --dir /export/home/user2/spams

Et lui donner des droits en exécution

chmod 755 /etc/cron.daily/bayes

5.4 Mettre en place un anti virus

5.4.1 Présentation et installation

Clam Anti virus (**clamav**) comme son nom l'indique est un anti virus qui est totalement libre, le site officiel est http://www.clamav.net/ on y récupérera l'archive qu'on décompresse en tapant

tar xvfz clamav-0.99.2.tar.gz

Cela donne **clamav-0.99.2** avant d'aller plus loin vous pouvez récupérer la trés bonne documentation disponible à cet endroit http://wiki.clamav.net/Main/WebHome. En suivant les instructions on doit d'abord en tant que root créer un utilisateur **clamav**

groupadd clamav
useradd -g clamav -s /bin/false -c "Clam Anti Virus" clamav

Ensuite en tant que simple utilisateur dans le répertoire **clamav-0.99.2** on doit taper

./configure --sysconfdir=/etc --enable-milter --with-openssl=/usr/local

L'option **sysconfdir** permet de retrouver le fichier de configuration sous **/etc**, l'option **enable-milter** est nécessaire si vous utilisez **sendmail**, attention dans ce dernier cas, installez le package **sendmail-devel** omettez cette dernière option si vous n'utilisez pas **sendmail**.

voilà le résultat

configure: Summary of detected features follows
OS : linux-gnu
pthreads : yes (-lpthread)
configure: Summary of miscellaneous features
check : no (auto)
fanotify : yes
fdpassing : 1
IPv6 : yes
configure: Summary of optional tools
clamdtop : -lncurses (auto)
milter : yes
clamsubmit : no (Please use the web interface for submitting FPs/FNs.)
configure: Summary of engine performance features

```
        release mode: yes
        llvm        : yes, from built-in (auto)
        mempool     : yes
configure: Summary of engine detection features
        bzip2       : ok
        zlib        : /usr
        unrar       : yes
        pcre        : /usr
        libxml2     : yes, from /usr
        yara        : yes
```

On tape ensuite sous **clamav-0.99.2**

make

A noter que j'ai utilisé la version 1.0.2d d'**OpenSSL** car avec la dernière 1.1.0c j'avais cette erreur

```
crypto.c: In function ‘cl_load_crl’:
crypto.c:1113:32: erreur: déréférencement d'un pointeur de type incomplet
      tm = cl_ASN1_GetTimeT(x->crl->nextUpdate);
                            ^
```

si vous avez l'erreur suivante

```
bytecode2llvm.cpp:189:25: fatal error: openssl/ssl.h: No such file or directory
 #include <openssl/ssl.h>
```

j'ai créé ce lien en tant que root

ln -s /usr/local/ssl/include/openssl /usr/local/include

on retape **make** puis en tant que root

make install

On rajoute si ce n'est déjé fait la ligne **/usr/local/lib** (et **/usr/local/lib64** pour une version 64bits) dans le fichier **/etc/ld.so.conf** et on tape

ldconfig

Maintenant on crée le répertoire de log de **clamav**, l'utilisateur **clamav** doit en étre propriétaire

mkdir /var/log/clamav
chown clamav:clamav /var/log/clamav

5.4.2 Configuration

On édite le fichier **/etc/clamd.conf** voici comment je l'ai configuré

```
##
## Example config file for the Clam AV daemon
## Please read the clamd.conf(5) manual before editing this file.
##


# Comment or remove the line below.
#Example

# Uncomment this option to enable logging.
# LogFile must be writable for the user running daemon.
# A full path is required.
# Default: disabled
#LogFile /tmp/clamd.log
```

```
LogFile /var/log/clamav/clamd.log

# By default the log file is locked for writing - the lock protects against
# running clamd multiple times (if want to run another clamd, please
# copy the configuration file, change the LogFile variable, and run
# the daemon with --config-file option).
# This option disables log file locking.
# Default: no
#LogFileUnlock yes

# Maximum size of the log file.
# Value of 0 disables the limit.
# You may use 'M' or 'm' for megabytes (1M = 1m = 1048576 bytes)
# and 'K' or 'k' for kilobytes (1K = 1k = 1024 bytes). To specify the size
# in bytes just don't use modifiers. If LogFileMaxSize is enabled, log
# rotation (the LogRotate option) will always be enabled.
# Default: 1M
LogFileMaxSize 2M

# Log time with each message.
# Default: no
#LogTime yes

# Also log clean files. Useful in debugging but drastically increases the
# log size.
# Default: no
#LogClean yes

# Use system logger (can work together with LogFile).
# Default: no
#LogSyslog yes

# Specify the type of syslog messages - please refer to 'man syslog'
# for facility names.
# Default: LOG_LOCAL6
#LogFacility LOG_MAIL

# Enable verbose logging.
# Default: no
#LogVerbose yes

# Enable log rotation. Always enabled when LogFileMaxSize is enabled.
# Default: no
#LogRotate yes

# Log additional information about the infected file, such as its
# size and hash, together with the virus name.
#ExtendedDetectionInfo yes

# This option allows you to save a process identifier of the listening
# daemon (main thread).
# Default: disabled
PidFile /var/log/clamav/clamd.pid

# Optional path to the global temporary directory.
# Default: system specific (usually /tmp or /var/tmp).
TemporaryDirectory /tmp

# Path to the database directory.
# Default: hardcoded (depends on installation options)
DatabaseDirectory /usr/local/share/clamav

# Only load the official signatures published by the ClamAV project.
```

```
# Default: no
#OfficialDatabaseOnly no

# The daemon can work in local mode, network mode or both.
# Due to security reasons we recommend the local mode.

# Path to a local socket file the daemon will listen on.
# Default: disabled (must be specified by a user)
#LocalSocket /tmp/clamd.socket
LocalSocket  /var/log/clamav/clamd.sock


# Sets the group ownership on the unix socket.
# Default: disabled (the primary group of the user running clamd)
#LocalSocketGroup virusgroup

# Sets the permissions on the unix socket to the specified mode.
# Default: disabled (socket is world accessible)
#LocalSocketMode 660

# Remove stale socket after unclean shutdown.
# Default: yes
FixStaleSocket yes

# TCP port address.
# Default: no
#TCPSocket 3310

# TCP address.
# By default we bind to INADDR_ANY, probably not wise.
# Enable the following to provide some degree of protection
# from the outside world. This option can be specified multiple
# times if you want to listen on multiple IPs. IPv6 is now supported.
# Default: no
#TCPAddr 127.0.0.1

# Maximum length the queue of pending connections may grow to.
# Default: 200
#MaxConnectionQueueLength 30

# Clamd uses FTP-like protocol to receive data from remote clients.
# If you are using clamav-milter to balance load between remote clamd daemons
# on firewall servers you may need to tune the options below.

# Close the connection when the data size limit is exceeded.
# The value should match your MTA's limit for a maximum attachment size.
# Default: 25M
#StreamMaxLength 10M

# Limit port range.
# Default: 1024
#StreamMinPort 30000
# Default: 2048
#StreamMaxPort 32000

# Maximum number of threads running at the same time.
# Default: 10
MaxThreads 20

# Waiting for data from a client socket will timeout after this time (seconds).
# Default: 120
ReadTimeout 300
```

```
# This option specifies the time (in seconds) after which clamd should
# timeout if a client doesn't provide any initial command after connecting.
# Default: 5
#CommandReadTimeout 5

# This option specifies how long to wait (in miliseconds) if the send buffer is full.
# Keep this value low to prevent clamd hanging
#
# Default: 500
#SendBufTimeout 200

# Maximum number of queued items (including those being processed by MaxThreads threads)
# It is recommended to have this value at least twice MaxThreads if possible.
# WARNING: you shouldn't increase this too much to avoid running out  of file descriptors,
# the following condition should hold:
# MaxThreads*MaxRecursion + (MaxQueue - MaxThreads) + 6< RLIMIT_NOFILE (usual max is 1024)
#
# Default: 100
#MaxQueue 200

# Waiting for a new job will timeout after this time (seconds).
# Default: 30
#IdleTimeout 60

# Don't scan files and directories matching regex
# This directive can be used multiple times
# Default: scan all
#ExcludePath ^/proc/
#ExcludePath ^/sys/

# Maximum depth directories are scanned at.
# Default: 15
#MaxDirectoryRecursion 20

# Follow directory symlinks.
# Default: no
#FollowDirectorySymlinks yes

# Follow regular file symlinks.
# Default: no
#FollowFileSymlinks yes

# Scan files and directories on other filesystems.
# Default: yes
#CrossFilesystems yes

# Perform a database check.
# Default: 600 (10 min)
#SelfCheck 600

# Execute a command when virus is found. In the command string %v will
# be replaced with the virus name.
# Default: no
#VirusEvent /usr/local/bin/send_sms 123456789 "VIRUS ALERT: %v"

# Run as another user (clamd must be started by root for this option to work)
# Default: don't drop privileges
User clamav

# Initialize supplementary group access (clamd must be started by root).
# Default: no
#AllowSupplementaryGroups no
```

```
# Stop daemon when libclamav reports out of memory condition.
#ExitOnOOM yes

# Don't fork into background.
# Default: no
Foreground yes

# Enable debug messages in libclamav.
# Default: no
# Debug yes

# Do not remove temporary files (for debug purposes).
# Default: no
#LeaveTemporaryFiles yes

# Permit use of the ALLMATCHSCAN command. If set to no, clamd will reject
# any ALLMATCHSCAN command as invalid.
# Default: yes
#AllowAllMatchScan no

# Detect Possibly Unwanted Applications.
# Default: no
#DetectPUA yes

# Exclude a specific PUA category. This directive can be used multiple times.
# See https://github.com/vrtadmin/clamav-faq/blob/master/faq/faq-pua.md for
# the complete list of PUA categories.
# Default: Load all categories (if DetectPUA is activated)
#ExcludePUA NetTool
#ExcludePUA PWTool

# Only include a specific PUA category. This directive can be used multiple
# times.
# Default: Load all categories (if DetectPUA is activated)
#IncludePUA Spy
#IncludePUA Scanner
#IncludePUA RAT

# In some cases (eg. complex malware, exploits in graphic files, and others),
# ClamAV uses special algorithms to provide accurate detection. This option
# controls the algorithmic detection.
# Default: yes
AlgorithmicDetection yes

# This option causes memory or nested map scans to dump the content to disk.
# If you turn on this option, more data is written to disk and is available
# when the LeaveTemporaryFiles option is enabled.
#ForceToDisk yes

# This option allows you to disable the caching feature of the engine. By
# default, the engine will store an MD5 in a cache of any files that are
# not flagged as virus or that hit limits checks. Disabling the cache will
# have a negative performance impact on large scans.
# Default: no
#DisableCache yes

##
## Executable files
##

# PE stands for Portable Executable - it's an executable file format used
# in all 32 and 64-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. If you turn off this option, the original files will still be
# scanned, but without additional processing.
# Default: yes
ScanPE yes

# Certain PE files contain an authenticode signature. By default, we check
# the signature chain in the PE file against a database of trusted and
# revoked certificates if the file being scanned is marked as a virus.
# If any certificate in the chain validates against any trusted root, but
# does not match any revoked certificate, the file is marked as whitelisted.
# If the file does match a revoked certificate, the file is marked as virus.
# The following setting completely turns off authenticode verification.
# Default: no
#DisableCertCheck yes

# Executable and Linking Format is a standard format for UN*X executables.
# This option allows you to control the scanning of ELF files.
# If you turn off this option, the original files will still be scanned, but
# without additional processing.
# Default: yes
ScanELF yes

# With this option clamav will try to detect broken executables (both PE and
# ELF) and mark them as Broken.Executable.
# Default: no
DetectBrokenExecutables yes


##
## Documents
##

# This option enables scanning of OLE2 files, such as Microsoft Office
# documents and .msi files.
# If you turn off this option, the original files will still be scanned, but
# without additional processing.
# Default: yes
ScanOLE2 yes

# With this option enabled OLE2 files with VBA macros, which were not
# detected by signatures will be marked as "Heuristics.OLE2.ContainsMacros".
# Default: no
#OLE2BlockMacros no

# This option enables scanning within PDF files.
# If you turn off this option, the original files will still be scanned, but
# without decoding and additional processing.
# Default: yes
ScanPDF yes

# This option enables scanning within SWF files.
# If you turn off this option, the original files will still be scanned, but
# without decoding and additional processing.
# Default: yes
ScanSWF yes


##
## Mail files
##

# Enable internal e-mail scanner.
```

```
# If you turn off this option, the original files will still be scanned, but
# without parsing individual messages/attachments.
# Default: yes
ScanMail yes

# Scan RFC1341 messages split over many emails.
# You will need to periodically clean up $TemporaryDirectory/clamav-partial directory.
# WARNING: This option may open your system to a DoS attack.
#          Never use it on loaded servers.
# Default: no
ScanPartialMessages yes

# With this option enabled ClamAV will try to detect phishing attempts by using
# signatures.
# Default: yes
PhishingSignatures yes

# Scan URLs found in mails for phishing attempts using heuristics.
# Default: yes
PhishingScanURLs yes

# Always block SSL mismatches in URLs, even if the URL isn't in the database.
# This can lead to false positives.
#
# Default: no
#PhishingAlwaysBlockSSLMismatch no

# Always block cloaked URLs, even if URL isn't in database.
# This can lead to false positives.
#
# Default: no
#PhishingAlwaysBlockCloak no

# Detect partition intersections in raw disk images using heuristics.
# Default: no
#PartitionIntersection no

# Allow heuristic match to take precedence.
# When enabled, if a heuristic scan (such as phishingScan) detects
# a possible virus/phish it will stop scan immediately. Recommended, saves CPU
# scan-time.
# When disabled, virus/phish detected by heuristic scans will be reported only at
# the end of a scan. If an archive contains both a heuristically detected
# virus/phish, and a real malware, the real malware will be reported
#
# Keep this disabled if you intend to handle "*.Heuristics.*" viruses
# differently from "real" malware.
# If a non-heuristically-detected virus (signature-based) is found first,
# the scan is interrupted immediately, regardless of this config option.
#
# Default: no
#HeuristicScanPrecedence yes


##
## Data Loss Prevention (DLP)
##

# Enable the DLP module
# Default: No
#StructuredDataDetection yes

# This option sets the lowest number of Credit Card numbers found in a file
```

```
# to generate a detect.
# Default: 3
#StructuredMinCreditCardCount 5

# This option sets the lowest number of Social Security Numbers found
# in a file to generate a detect.
# Default: 3
#StructuredMinSSNCount 5

# With this option enabled the DLP module will search for valid
# SSNs formatted as xxx-yy-zzzz
# Default: yes
#StructuredSSNFormatNormal yes

# With this option enabled the DLP module will search for valid
# SSNs formatted as xxxyyzzzz
# Default: no
#StructuredSSNFormatStripped yes


##
## HTML
##

# Perform HTML normalisation and decryption of MS Script Encoder code.
# Default: yes
# If you turn off this option, the original files will still be scanned, but
# without additional processing.
#ScanHTML yes


##
## Archives
##

# ClamAV can scan within archives and compressed files.
# If you turn off this option, the original files will still be scanned, but
# without unpacking and additional processing.
# Default: yes
ScanArchive yes

# Mark encrypted archives as viruses (Encrypted.Zip, Encrypted.RAR).
# Default: no
#ArchiveBlockEncrypted no


##
## Limits
##

# The options below protect your system against Denial of Service attacks
# using archive bombs.

# This option sets the maximum amount of data to be scanned for each input file.
# Archives and other containers are recursively extracted and scanned up to this
# value.
# Value of 0 disables the limit
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 100M
#MaxScanSize 150M

# Files larger than this limit won't be scanned. Affects the input file itself
```

```
# as well as files contained inside it (when the input file is an archive, a
# document or some other kind of container).
# Value of 0 disables the limit.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 25M
#MaxFileSize 30M

# Nested archives are scanned recursively, e.g. if a Zip archive contains a RAR
# file, all files within it will also be scanned. This options specifies how
# deeply the process should be continued.
# Note: setting this limit too high may result in severe damage to the system.
# Default: 16
#MaxRecursion 10

# Number of files to be scanned within an archive, a document, or any other
# container file.
# Value of 0 disables the limit.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 10000
#MaxFiles 15000

# Maximum size of a file to check for embedded PE. Files larger than this value
# will skip the additional analysis step.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 10M
#MaxEmbeddedPE 10M

# Maximum size of a HTML file to normalize. HTML files larger than this value
# will not be normalized or scanned.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 10M
#MaxHTMLNormalize 10M

# Maximum size of a normalized HTML file to scan. HTML files larger than this
# value after normalization will not be scanned.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 2M
#MaxHTMLNoTags 2M

# Maximum size of a script file to normalize. Script content larger than this
# value will not be normalized or scanned.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 5M
#MaxScriptNormalize 5M

# Maximum size of a ZIP file to reanalyze type recognition. ZIP files larger
# than this value will skip the step to potentially reanalyze as PE.
# Note: disabling this limit or setting it too high may result in severe damage
# to the system.
# Default: 1M
#MaxZipTypeRcg 1M

# This option sets the maximum number of partitions of a raw disk image to be scanned.
# Raw disk images with more partitions than this value will have up to the value number
# partitions scanned. Negative values are not allowed.
# Note: setting this limit too high may result in severe damage or impact performance.
# Default: 50
```

```
#MaxPartitions 128

# This option sets the maximum number of icons within a PE to be scanned.
# PE files with more icons than this value will have up to the value number icons scanned.
# Negative values are not allowed.
# WARNING: setting this limit too high may result in severe damage or impact performance.
# Default: 100
#MaxIconsPE 200

##
## On-access Scan Settings
##

# Enable on-access scanning. Currently, this is supported via fanotify.
# Clamuko/Dazuko support has been deprecated.
# Default: no
#ScanOnAccess yes

# Don't scan files larger than OnAccessMaxFileSize
# Value of 0 disables the limit.
# Default: 5M
#OnAccessMaxFileSize 10M

# Set the include paths (all files inside them will be scanned). You can have
# multiple OnAccessIncludePath directives but each directory must be added
# in a separate line. (On-access scan only)
# Default: disabled
#OnAccessIncludePath /home
#OnAccessIncludePath /students

# Set the exclude paths. All subdirectories are also excluded.
# (On-access scan only)
# Default: disabled
#OnAccessExcludePath /home/bofh

# With this option you can whitelist specific UIDs. Processes with these UIDs
# will be able to access all files.
# This option can be used multiple times (one per line).
# Default: disabled
#OnAccessExcludeUID 0


##
## Bytecode
##

# With this option enabled ClamAV will load bytecode from the database.
# It is highly recommended you keep this option on, otherwise you'll miss detections for many new viruses.
# Default: yes
Bytecode yes

# Set bytecode security level.
# Possible values:
#       None - no security at all, meant for debugging. DO NOT USE THIS ON PRODUCTION SYSTEMS
#         This value is only available if clamav was built with --enable-debug!
#       TrustSigned - trust bytecode loaded from signed .c[lv]d files,
#               insert runtime safety checks for bytecode loaded from other sources
#       Paranoid - don't trust any bytecode, insert runtime checks for all
# Recommended: TrustSigned, because bytecode in .cvd files already has these checks
# Note that by default only signed bytecode is loaded, currently you can only
# load unsigned bytecode in --enable-debug mode.
#
# Default: TrustSigned
```

#BytecodeSecurity TrustSigned

 # Set bytecode timeout in miliseconds.
#
 # Default: 5000
 # BytecodeTimeout 1000

##
Statistics gathering and submitting
##

 # Enable statistical reporting.
 # Default: no
#StatsEnabled yes

 # Disable submission of individual PE sections for files flagged as malware.
 # Default: no
#StatsPEDisabled yes

 # HostID in the form of an UUID to use when submitting statistical information.
 # Default: auto
#StatsHostID auto

 # Time in seconds to wait for the stats server to come back with a response
 # Default: 10
#StatsTimeout 10

Le fichier de configuration de **clamav-milter** s'appelle **/etc/clamav-milter.conf** le voici, voilà les lignes que j'ai modifiées

#Example

MilterSocket /var/log/clamav/clmilter.sock

ClamdSocket unix:/var/log/clamav/clamd.sock

AddHeader Replace

LogFile /var/log/clamav/clamav-milter.log

pour le reste tout est en commentaire

On configure maintenant le fichier **/etc/freshclam.conf** en mettant en commentaire la ligne suivante

#Example

Puis en modifiant la ligne suivante conformément à ce qui a été défini dans le fichier **clamd.conf**

 # définition de la base des données des virus
DatabaseDirectory /usr/local/share/clamav

J'ai modifié ensuite les lignes suivantes

 # définition du fichier de log de freshclam
UpdateLogFile /var/log/clamav/freshclam.log

 # serveur miroir à contacter pour récupérer les mises oà jour
DatabaseMirror db.fr.clamav.net

 # database.clamav.net is a round-robin record which points to our most
 # reliable mirrors. It's used as a fall back in case db.XY.clamav.net is
 # not working. DO NOT TOUCH the following line unless you know what you
 # are doing.

DatabaseMirror database.clamav.net

on pensera à créer préalablement les fichiers de log

touch /var/log/clamav/clamd.log
touch /var/log/clamav/freshclam.log
touch /var/log/clamav/clamav-milter.log

clamav doit être le propriétaire des deux premiers fichiers

chown clamav:clamav /var/log/clamav/clamd.log
chown clamav:clamav /var/log/clamav/freshclam.log

on doit créer préalablement le répertoire contenant la base de données des virus et **clamav** doit en être propriétaire

mkdir /usr/local/share/clamav
chown clamav:clamav /usr/local/share/clamav

La commande **freshclam** permet de mettre à jour la base de données à partir d'informations récupérées sur internet, en tant que root tapez **freshclam** voilà le résultat:

ClamAV update process started at Sun Oct 4 21:48:27 2015
Downloading main.cvd [100%]
main.cvd updated (version: 55, sigs: 2424225, f-level: 60, builder: neo)
Downloading daily.cvd [100%]
daily.cvd updated (version: 20952, sigs: 1589971, f-level: 63, builder: jesler)
Downloading bytecode.cvd [100%]
bytecode.cvd updated (version: 268, sigs: 47, f-level: 63, builder: anvilleg)
Database updated (4014243 signatures) from db.fr.clamav.net (IP: 178.33.105.132)

Pour information quand cette commande est lancée c'est l'utilisateur **clamav** qui devient propriétaire du process.

5.4.3 Premiers tests

On lance maintenant **clamd** en tant que root

clamd

On va faire un test maintenant sur le répertoire **clamav-0.99.2** en tant que simple utilisateur

clamscan -r -l log.txt clamav-0.99.2/
L'option **-r** permet d'avoir une recherche récursive (à travers le répertoire et ses sous répertoires), **-l** pour logger dans le fichier **log.txt**. Voilà un extrait du contenu du dit-fichier aprés l'exécution de la commande

----------- SCAN SUMMARY -----------
Known viruses: 5357163
Engine version: 0.99.2
Scanned directories: 254
Scanned files: 5371
Infected files: 50
Data scanned: 272.46 MB
Data read: 170.53 MB (ratio 1.60:1)
Time: 39.734 sec (0 m 39 s)

Il y a des virus qui ont été volontairement placés dans ce répertoire pour mener des essais. Pour scanner le répertoire de mail **/var/spool/mail** il faudra être root et rajouter l'option **--mbox**

5.4.4 Lancement automatique

On peut configurer un lancement automatique pour les mises à jour de la base de donnée avec **cron**, pour une mise à jour tous les jours on créera dans le fichier **/etc/cron.daily** le fichier **freshclam**

#!/bin/bash
/usr/local/bin/freshclam --quiet -l /var/log/clamav/clam-update.log

Avec les droits d'exécution

chmod 755 freshclam

On va créer maintenant un fichier de log pour les mises à jour et rendre l'utilisateur **clamav** propriétaire

touch /var/log/clamav/clam-update.log
chmod 600 /var/log/clamav/clam-update.log
chown clamav:clamav /var/log/clamav/clam-update.log

Autre solution pour un lancement simple en tant que daemon (lancement six fois par jour) on tape

freshclam -d -c 6 -l /var/log/clamav/clam-update.log

Maintenant pour un lancement automatique du daemon **clamd** on va créer le fichier **clamd.service** sous **/usr/lib/systemd/system** voici son contenu

[Unit]
Description = clamd scanner daemon
After = syslog.target nss-lookup.target network.target

[Service]
Type = simple
ExecStart = /usr/local/sbin/clamd -c /etc/clamd.conf
Restart = on-failure
PrivateTmp = true

[Install]
WantedBy=multi-user.target

maintenant pour que le service soit lancé à chaque boot de la machine il faudra taper

systemctl enable clamd.service

voillà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/clamd.service to /usr/lib/systemd/system/clamd.service.

pour le lancer il suffit maintenant de taper

systemctl start clamd.service

et voilà ce que ça donne quand on tape **systemctl status clamd**

● clamd.service - clamd scanner daemon
Loaded: loaded (/usr/lib/systemd/system/clamd.service; enabled)
Active: active (running) since dim. 2015-10-04 21:52:59 CEST; 8s ago
Main PID: 2787 (clamd)
CGroup: /system.slice/clamd.service
└─2787 /usr/local/sbin/clamd -c /etc/clamd.conf

Lancement automatique du scanner

Pour un lancement automatique du scanner vous pouvez utiliser **cron**, créer un fichier **scanvirus** à placer sous

/etc/cron.daily (chaque jour) ou **/etc/cron.hourly** (chaque heure) contenant

```
#!/bin/bash
/usr/local/bin/clamscan -r -l /var/log/clamscan/scan.log /home
/usr/local/bin/clamscan -r --mbox /var/spool/mail
```

Il faut le rendre exécutable

chmod 755 /etc/cron.daily/scanvirus.

Il scannera tous les jours les répertoires **/home** et **/var/spool/mail**. Libre à vous de rajouter des scans dans les partages **samba** ou vos partitions windows.

5.4.5 Interfaçage avec sendmail

L'intérêt maintenant d'un anti virus est un scan automatique à la réception mais également à l'envoi d'emails. Pour cela **Clam Anti Virus** peut très facilement s'interfacer avec **sendmail**. Si la compilation s'est bien passée vous devriez trouver un fichier **clamav-milter** sous **/usr/local/sbin**. Dans le fichier de config de **sendmail** (sous **/etc/mail/** ou **/usr/share/sendmail-cf/cf**) on rajoutera tout à la fin les lignes

```
INPUT_MAIL_FILTER(`clmilter', `S=local:/var/log/clamav/clmilter.sock, F=, T=S:4m;R:4m')dnl
define(`confINPUT_MAIL_FILTERS', `clmilter')
```

Dans l'hypothèse où **spamassassin** est déjà interfacé avec **sendmail**, il faudra modifier les dernières lignes comme cela.

```
INPUT_MAIL_FILTER(`clmilter', `S=local:/var/log/clamav/clmilter.sock, F=, T=S:4m;R:4m')dnl
INPUT_MAIL_FILTER(`spamassassin', `S=local:/var/run/spamass.sock, F=,
T=C:15m;S:4m;R:4m;E:10m')dnl
define(`confMILTER_MACROS_CONNECT',`t, b, j, _, {daemon_name}, {if_name}, {if_addr}')dnl
define(`confMILTER_MACROS_HELO',`s, {tls_version}, {cipher}, {cipher_bits}, {cert_subject},
{cert_issuer}')dnl
define(`confMILTER_MACROS_ENVFROM',`i, {auth_authen}, {auth_type}')dnl
define(`confMILTER_MACROS_ENVRCPT',`r, v, Z')dnl
define(`confINPUT_MAIL_FILTERS', `clmilter,spamassassin')
```

Dans le fichier **/etc/clamd.conf** on modifiera la ligne suivante

```
# Path to the local socket. The daemon doesn't change the mode of the
# created file (portability reasons). You may want to create it in a directory
# which is only accessible for a user running daemon.
# je n'ai pas mis le répertoire par défaut car l'utilisateur clamav
# ne peut écrire sous /var/run
LocalSocket /var/log/clamav/clamd.sock
```

On relance maintenant **clamd**

systemctl restart clamd.service

Et on lance **clamav-milter**

clamav-milter -c /etc/clamav-milter.conf

Pour information même en le lançant en tant que root, ce sera l'utilisateur **clamav** qui sera le propriétaire du process **clamav-milter**. Maintenant on relance **sendmail** en supposant que votre fichier de conf se trouve sous **/usr/share/sendmail-cf/cf** et se nomme **config.mc**

```
systemctl stop sendmail
cd  /usr/share/sendmail-cf/cf/
m4  config.mc > /etc/mail/sendmail.cf
systemctl start sendmail
```

Maintenant pour lancer **clamav-milter** automatiquement

[Unit]
Description='ClamAV Milter'
After=clamd.service

[Service]
Type=forking
ExecStart=/usr/local/sbin/clamav-milter --config-file /etc/clamav-milter.conf

[Install]
WantedBy=multi-user.target

maintenant pour que le service soit lancé à chaque boot de la machine il faudra taper

systemctl enable clamav-milter.service

voillà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/clamav-milter.service to /usr/lib/systemd/system/clamav-milter.service.

pour le lancer il suffit maintenant de taper

systemctl start clamav-milter.service

et voilà ce que donne la commande **systemctl status clamav-milter.service**

● clamav-milter.service - 'ClamAV Milter'
Loaded: loaded (/usr/lib/systemd/system/clamav-milter.service; enabled)
Active: active (running) since dim. 2015-10-04 20:59:53 CEST; 55min ago
Main PID: 1533 (clamav-milter)
CGroup: /system.slice/clamav-milter.service
└─1533 /usr/local/sbin/clamav-milter --config-file /etc/clamav-milter.conf

Si au lancement vous avez l'erreur suivante

/usr/local/sbin/clamav-milter: --max-children must be given if --external is not given

Vérifiez bien que vous avez décommenté la ligne suivante dans le fichier **/etc/clamd.conf**

 # Maximal number of threads running at the same time.
 # Default: 10
MaxThreads 20

Maintenant comment sait-on si un virus a été intercepté ? Avec **fetchmail** quand on récupére le courrier on a un message de ce genre

fetchmail: lecture du message olivier.hoarau@funix.org@pop.pro.proxad.net:36 parmi 37 (3143 octets)
fetchmail: éliminé
fetchmail: Le serveur SMTP a refusé de délivrer le courrier

Pour que le mail soit purement et simplement supprimé. Dans votre fichier **.fetchmailrc** il faudra rajouter la ligne suivante

poll pop.fai.net protocol pop3
user olivier.hoarau@funix.org with password machinchose is olivier here
options antispam 550 554;

Sinon il restera dans **/var/spool/mail**. Plus en détail avec **journalctl** on obtient

Aug 23 12:20:08 web sm-mta[5974]: l7NAHEvp005825: to=<olivier@localhost>, delay=00:00:03, xdelay=00:00:00, mailer=local, pri=32226, dsn=2.0.0, stat=Sent
Aug 23 12:20:08 web sm-mta[5825]: l7NAHEvq005825: from=<ron@bakernet.com>, size=1356, class=0, nrcpts=1, msgid=<1INISJ-000JQP-FP@216-201-156-242.res.logixcom.net>, proto=ESMTP, daemon=MTA, relay=localhost [127.0.0.1]
Aug 23 12:20:08 web sm-mta[5825]: l7NAHEvq005825: Milter add: header: X-Virus-Scanned: ClamAV

0.91.1/4016/Tue Aug 21 01:40:52 2007 on web.kervao.fr
Aug 23 12:20:08 web sm-mta[5825]: l7NAHEvq005825: Milter add: header: X-Virus-Status: Infected with Email.Webaccount-4
Aug 23 12:20:08 web sendmail[5976]: l7NAK8cl005976: from=clamav, size=499, class=0, nrcpts=2, msgid=<200708231020.l7NAK8cl005976@web.kervao.fr>, relay=clamav@localhost
Aug 23 12:20:08 web sendmail[5976]: l7NAK8cl005976: to=postmaster, delay=00:00:00, mailer=relay, pri=60499, stat=queued
Aug 23 12:20:08 web sendmail[5976]: l7NAK8cl005976: to=olivier, delay=00:00:00, mailer=relay, pri=60499, stat=queued
Aug 23 12:20:08 web sm-mta[5825]: l7NAHEvq005825: Milter: data, discard
Aug 23 12:20:08 web sm-mta[5825]: l7NAHEvq005825: discarded

Chaque mail se voit rajouter la ligne suivante dans son entête

X-Virus-Scanned: clamav-milter 0.99.2 at mana.kervao.fr
X-Virus-Status: Clean