

# CHAPITRE 7

## Postfix & Courrier-imap

Postfix est certainement un des meilleurs serveurs d' e-mail existant, c'est pourquoi mon choix se porte sur celui-ci. Le sujet des e-mails est un sujet compliqué ou les documentations sont nombreuses et confusent assez souvent, je n'en ferai aujourd'hui toujours pas ma spécialité. Je vais donc ne pas m'étendre sur ce sujet.

Nous voulons ici créer un serveur MX principal qui recevra nos mails et les enverra au reste du monde. Nous souhaitons pouvoir envoyé des e-mails depuis ce serveur depuis n'importe où. Dans cette configuration pour ne pas finir en open relay, il n'y a que l'authentification SMTP qui puisse nous protégé de ceci. J'ai testé auparavant le pop-before-smtp qui permet d'en faire autant mais ce n'est qu'un bricolage face à l'authentification SMTP. C'est pourquoi ici on fera une véritable authentification bien sur avec un peu de SSL.

Commençons par installer Postfix et ce qui sera nécessaire à l'authentification SMTP.

```
ALBAN@bebeserv:~$ sudo -i
Password:
bebeserv:~# apt-get update
[...]
bebeserv:~# apt-get install postfix libsasl2-modules sasl2-bin
[...]
```

L'installateur vous pose deux questions pour la configuration de Postfix. Répondez à la première "Site internet" et laissez la seconde qui devrait ressembler à « bebeserv.bebenet.local » tel quel.

Maintenant que c'est fait, stoppez Postfix :

```
bebeserv:~# /etc/init.d/postfix stop
[...]
```

Maintenant configurons notre serveur d'e-mail :

```
bebeserv:~# cd /etc/postfix
bebeserv:/etc/postfix# nano main.cf
```

Afin de garantir la sécurité du MTA faisons le passer pour un serveur Sendmail (un concurrent plus ancien) :

```
mail_name = Sendmail
mail_version = 8.12.10
smtpd_banner = $myhostname ESMTP $mail_name ($mail_version)
```

Maintenant quelques fonctionnalités et configuration :

```
append_dot_mydomain = yes
mynetwork_style = host
proxy_interfaces = 10.0.0.1
relay_domains=
masquerade_domains = bebenet.local
local_transport = virtual:$myhostname
```

Les utilisateurs virtuels

```
virtual_alias_domains =
hash:/etc/postfix/conf.d/virtual_alias_domains
virtual_alias_maps = hash:/etc/postfix/conf.d/virtual_alias
virtual_mailbox_domains =
hash:/etc/postfix/conf.d/virtual_mailbox_domains
virtual_mailbox_base = /var/mail/virtual
virtual_mailbox_maps = hash:/etc/postfix/conf.d/virtual_mailbox
virtual_uid_maps = static:5000
virtual_gid_maps = static:5000
```

Encore un peu de sécurité et pour l'authentification SMTP

```
smtpd_helo_timeout = 10s
smtpd_recipients_limit = 16
smtpd_soft_error_limit = 3
smtpd_hard_error_limit = 12
smtpd_helo_required = yes
smtpd_delay_reject = yes
disable_vrfy_command = yes
strict_rfc821_envelopes = yes

broken_sasl_auth_clients = yes
smtpd_sasl_auth_enable = yes
smtpd_sasl_path = smtpd
smtpd_sasl_exceptions_networks = $mynetworks
smtpd_sasl_security_options = noanonymous

smtpd_helo_restrictions =
    permit_mynetworks,
    reject_invalid_hostname,
    reject_unknown_hostname,
    reject_non_fqdn_hostname
```

```
smtpd_client_restrictions =
    sleep 1,
    reject_unauth_pipelining

smtpd_recipient_restrictions =
    permit_mynetworks,
    permit_sasl_authenticated,
    reject_invalid_hostname,
    reject_unauth_destination,
    reject_unknown_sender_domain,
    reject_unknown_recipient_domain,
    reject_non_fqdn_recipient

smtpd_sender_restrictions =
    reject_unknown_sender_domain,
    reject_non_fqdn_sender,
```

Pour le SSL :

```
smtpd_use_tls=yes
smtpd_tls_cert_file = /etc/postfix/smtpd.cert
smtpd_tls_key_file = /etc/postfix/smtpd.key
```

Enregistrez et quittez.

Générons directement les nouveaux certificats SSL pour Postfix :

```
bebeserv:/etc/postfix# openssl req -new -outform PEM -out
/etc/postfix/smtpd.cert -newkey rsa:2048 -nodes -keyout
/etc/postfix/smtpd.key -keyform PEM -days 365 -x509
bebeserv:/etc/postfix# chmod 640 smtpd.key
```

Le fait d'utiliser Sasl (un serveur d'authentification) nous impose de lui montré comment nous souhaitons authentifier les utilisateurs :

Pour ceci nous créons comme précisé par notre configuration : « smtpd\_sasl\_path » le fichier « /etc/postfix/sasl/smtpd.conf »

```
bebeserv:/etc/postfix# cd sasl
bebeserv:/etc/postfix/sasl# nano smtpd.conf
```

Entrez ceci :

```
pwcheck_method: saslauthd
mech_list: plain login
```

Enregistrer et quitter.

Visiblement la commande suivante est utile pour remédier à une histoire de bug...

```
bebeserv:/etc/postfix/sasl# ln -s /etc/postfix/sasl/smtpd.conf
/usr/lib/sasl2/smtpd.conf
```

Éditez le fichier de configuration de Sasl :

```
bebeserv:/etc/postfix/sasl# nano /etc/default/saslauthd
```

Modifier les lignes suivantes :

```
START=yes
MECHANISMS="rimap"
MECH_OPTIONS="localhost"
OPTIONS="-r -c -m /var/spool/postfix/var/run/saslauthd"
```

Enregistrez et quitter.

Maintenant ajoutons l'utilisateur Postfix au groupe Sasl et faisons en sorte que Sasl puisse créer ses fichiers dans la cage de postfix (ça aussi ça ne s'invente pas) :

```
bebeserv:/etc/postfix/sasl# dpkg-statoverride --add root sasl 710
/var/spool/postfix/var/run/saslauthd
bebeserv:/etc/postfix/sasl# adduser postfix sasl
[...]
```

Maintenant il va falloir alimenter les différentes bases de Postfix :

```
bebeserv:/etc/postfix/sasl# cd ..
bebeserv:/etc/postfix# mkdir conf.d
bebeserv:/etc/postfix# cd conf.d
bebeserv:/etc/postfix/conf.d# nano virtual_mailbox_domains
```

Entrez :

```
bebeserv.local  x
```

Enregistrez et quitter.

```
bebeserv:/etc/postfix/conf.d# nano virtual_mailbox
```

Entrez :

```
test@bebeserv.local    bebeserv.local/test/
```

Enregistrez et quitter.

```
bebeserv:/etc/postfix/conf.d# nano virtual_alias
```

Entrez :

```
postmaster@bebeserv.local    test@bebeserv.local
hostmaster@bebeserv.local    test@bebeserv.local
webmaster@bebeserv.local     test@bebeserv.local
www@bebeserv.local           test@bebeserv.local
ALBAN@bebeserv.local         test@bebeserv.local
```

Enregistrez et quitter.

```
bebeserv:/etc/postfix/conf.d# nano virtual_alias_domains
```

Entrez :

```
localhost bebeserv.local
```

Enregistrez et quitter.

Créons maintenant les bases compilées à l'aide de « postmap » :

```
bebeserv:/etc/postfix/conf.d# postmap virtual_alias
bebeserv:/etc/postfix/conf.d# postmap virtual_alias_domains
bebeserv:/etc/postfix/conf.d# postmap virtual_mailbox
bebeserv:/etc/postfix/conf.d# postmap virtual_mailbox_domains
```

Redirigeons « root » qui est normalement alias de tous les autres démon vers notre utilisateur principal :

```
bebeserv:/etc/postfix/conf.d# nano /etc/aliases
```

Entrez à la fin du fichier :

```
root: ALBAN
```

Enregistrez et quittez.

Remettons, de la même manière que les autres tables, celle-ci à jour :

```
bebeserv:/etc/postfix/conf.d# postalias /etc/aliases
```

Créons l'utilisateur et le groupe « virtual » et le dossier des boîtes e-mail virtuelles :

```
bebeserv:/etc/postfix/conf.d# cd /var/mail
bebeserv:/var/mail# mkdir virtual
bebeserv:/var/mail# groupadd -g 5000 virtual
bebeserv:/var/mail# useradd -u 5000 -g virtual -d /var/mail/virtual
-s /usr/sbin/nologin virtual
bebeserv:/var/mail# chown virtual.virtual virtual
```

Parfait ! Maintenant Postfix est fonctionnel et bien configuré. Nous l'avons configuré de sorte que les e-mails émis par les utilisateurs locaux soit traité par le relay virtuel, que localhost soit assimilé à notre domaine, que les sous domaines de notre domaine soit assimilé à notre domaine. Il ne nous reste plus qu'à installer un client IMAP pour lire nos messages.

Installons « courier-imap-ssl » :

```
bebeserv:/var/mail# apt-get install courier-imap-ssl
```

Stoppons les serveur nouvellement installés :

```
bebeserv:/var/mail# /etc/init.d/courier-authdaemon stop
[...]
bebeserv:/var/mail# /etc/init.d/courier-imap stop
[...]
bebeserv:/var/mail# /etc/init.d/courier-imap-ssl stop
[...]
```

Regénérons un certificat propre à notre domaine pour IMAP :

```
bebeserv:/var/mail# cd /etc/courier
bebeserv:/etc/courier# rm imapd.pem
bebeserv:/etc/courier# nano imapd.cnf
```

Modifiez ceci :

```
[ req_dn ]
C=FR
ST=FRANCE
L=Paris
O=Courier Mail Server
OU=IMAP SSL key
CN=imap.bebenet.local
emailAddress=postmaster@bebenet.local
```

Enregistrez et quittez.

Maintenant créons un beau certificat :

```
bebeserv:/etc/courier# dpkg-reconfigure courier-imap-ssl
[...]
```

Ceci fait, ré-arretons le serveur qui a été redémarrer et configurons l'authentification (ici authdaemon à ne pas confondre avec saslauthd) :

```
bebeserv:/etc/courier# /etc/init.d/courier-imap-ssl stop
[...]
```

Modifiez cette ligne :

```
authmodulelist="authuserdb"
```

Enregistrez et quittez.

Maintenant nous allons créer la base des utilisateurs qui vont pouvoir se connecter à IMAP. Pour simplifier les choses je vous ai concocté un petit script:

```
bebeserv:/etc/courier# mkdir userdb
bebeserv:/etc/courier# chmod 0000 userdb
bebeserv:/etc/courier# nano userdb-utils
```

Entrez ceci :

```

#!/bin/bash
cd /etc/courier
case "$1" in
add)
    if [ $# -lt 2 ]; then
        echo "Usage: ./userdb-utils add user@domain.tld"
        exit 1
    fi
    EMAIL=$2
    DIR=/var/mail/virtual
    USER=$(echo $EMAIL | cut -d@ -f1)
    DOMAIN=$(echo $EMAIL | cut -d@ -f2)
    userdb $DOMAIN/$EMAIL set mail=$DIR/$DOMAIN/$USER
home=$DIR/$DOMAIN/$USER uid=5000 gid=5000
    userdbpw -md5 | userdb $DOMAIN/$EMAIL set systempw
    makeuserdb
    ;;
del)
    if [ $# -lt 2 ]; then
        echo "Usage: ./userdb-utils del user@domain.tld"
        exit 1
    fi
    TODAY=$(date +%Y%m%d)
    FILE=/tmp/userdbcreate$TODAY
    EMAIL=$2
    USER=$(echo $EMAIL | cut -d@ -f1)
    DOMAIN=$(echo $EMAIL | cut -d@ -f2)
    DB=/etc/courier/userdb/$DOMAIN
    sed -e /^$EMAIL/d $DB > $FILE
    rm $DB
    mv $FILE $DB
    makeuserdb
    ;;
list)
    authenumerate
    ;;
*)
    echo "Usage: ./userdb-utils {add|del|list} [user@domain.tld]"
    exit 1
esac
exit 0

```

Enregistrez et quittez.

Maintenant rendons notre script exécutable et disponible dans notre banque de script :

```

bebeserv:/etc/courier# chmod 700 userdb-utils
bebeserv:/etc/courier# ln -s userdb-utils /usr/local/bin/

```

Ok, créons deux utilisateurs pour voir comment ceci fonctionne :

```

bebeserv:/etc/courier# ./userdb-utils add test@bebenet.local

```

```
Password :  
[...]  
bebeserv:/etc/courier# ./userdb-utils add test2@bebenet.local  
Password :  
[...]  
bebeserv:/etc/courier# ./userdb-utils list  
[...]  
bebeserv:/etc/courier# ./userdb-utils del test2@bebenet.local  
bebeserv:/etc/courier# ./userdb-utils list  
[...]
```

C'est terminé.

Il reste à mettre à jour « iptables »

```
bebeserv:/etc/courier# nano /etc/network/iptables
```

Rajouter ces lignes :

```
iptables -t filter -A INPUT -i eth0 -p tcp --dport 25 -d $MYIP -j  
ACCEPT  
iptables -t filter -A INPUT -i eth0 -p tcp --dport 993 -d $MYIP -j  
ACCEPT  
  
iptables -t filter -A OUTPUT -o eth0 -p tcp --dport 25 -s $MYIP -j  
ACCEPT  
iptables -t filter -A OUTPUT -o eth0 -p tcp --sport 25 -s $MYIP -j  
ACCEPT  
iptables -t filter -A OUTPUT -o eth0 -p tcp --dport 993 -s $MYIP -j  
ACCEPT  
iptables -t filter -A OUTPUT -o eth0 -p tcp --sport 993 -s $MYIP -j  
ACCEPT
```

Enregistrez et quittez :

Redémarrons les serveurs :

```
bebeserv:/var/mail# /etc/init.d/iptables-conf restart  
bebeserv:/var/mail# /etc/init.d/postfix start  
bebeserv:/var/mail# /etc/init.d/courier-authdaemon start  
bebeserv:/var/mail# /etc/init.d/courier-imap-ssl start  
bebeserv:/var/mail# /etc/init.d/courier-imap start
```

Et voilà ! Nos utilisateurs doivent se loguer forcément en IMAP et en SMTP le tout couvert du SSL.

Prochain chapitre on ajoute un antivirus et un antispam !

```
bebeserv:/var/mail# exit  
logout  
ALBAN@bebeserv:~$
```