

# Postfix, Dovecot, Sogo

Todo:

- Postfix: limiter les utilisateurs : /etc/postfix/sql/sql-relaydomains.cf
- Autoconfig
- Quotas
- backup

## Postfix

1. Installer les paquets nécessaires:

```
apt update
apt-get install postfix postfix-mysql mariadb-server
```

Choisir Internet Site, définir le nom du serveur ainsi que l'adresse pour le postmaster.

2. Sécuriser l'installation maria-db

```
mysql_secure_installation
```

3. Vérifier la configuration Mysql:

1. Editer le fichier client

```
nano /etc/mysql/mariadb.conf.d/50-client.cnf
```

Il faut vérifier que la ligne suivante est présente

```
[client]
default-character-set          = utf8mb4
```

2. Editer le fichier mysql\_client

```
nano /etc/mysql/mariadb.conf.d/50-mysql-client.cnf
```

Et vérifier si la configuration suivante est aussi définie:

```
[mysql]
default-character-set          = utf8mb4
```

3. Editer le fichier server

```
nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

Et vérifier si la configuration suivante est aussi définie:

```
[mysqld]
character-set-client-handshake = FALSE
```

```
character-set-server      = utf8mb4
collation-server         = utf8mb4_unicode_ci
innodb_file_per_table    = TRUE
innodb_file_format       = barracuda
innodb_large_prefix      = TRUE
max_allowed_packet       = 128M
```

4. Ajouter root dans le groupe de postfix

```
adduser root postfix
```

5. Editer le fichier master.cf

```
nano /etc/postfix/master.cf
```

Modifier/ajouter les lignes suivantes:

```
submission inet n      -      y      -      -      smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_tls_auth_only=yes
  -o smtpd_reject_unlisted_recipient=no
  -o smtpd_sasl_type=dovecot
  -o smtpd_sasl_path=private/auth
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
#  -o smtpd_recipient_restrictions=
#  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
# Ancienne version LDA
#dovecot    unix      -      n      n      -      -      pipe
#  flags=DRhu user=vmail:vmail argv=/usr/lib/dovecot/deliver -f
#  ${sender} -d ${recipient}
```

6. Editer à présent le fichier main.cf

```
nano /etc/postfix/main.cf
```

Et veiller à ce que les lignes ressemblent à ceci:

```
myhostname = mail3.makeitsimple.be
mydomain = makeitsimple.be
myorigin = $myhostname
inet_interfaces = all
inet_protocols = all
mydestination = $myhostname, localhost.$mydomain, localhost
smtpd_recipient_restrictions = permit_mynetworks
reject_unauth_destination
smtpd_sender_restrictions = reject_unknown_sender_domain
home_mailbox = Maildir/
```

```
append_dot_mydomain = no
biff = no
config_directory = /etc/postfix
dovecot_destination_recipient_limit = 1
message_size_limit = 4194304
smtpd_tls_key_file = /etc/postfix/ssl/yourkey.key
smtpd_tls_cert_file = /etc/postfix/ssl/yourcertificate.crt
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache

smtp_tls_key_file = /etc/postfix/ssl/yourkey.key
smtp_tls_cert_file = /etc/postfix/ssl/yourcertificate.crt
smtp_use_tls=yes
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
smtp_tls_security_level=may
smtp_tls_note_starttls_offer = yes
smtp_tls_enforce_peername = no

smtpd_tls_security_level=may
#Transport LDA
#virtual_transport = dovecot
#Transport LMTP
virtual_transport = lmtp:unix:private/dovecot-lmtp
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth

proxy_read_maps =
    proxy:unix:passwd.byname
    proxy:mysql:/etc/postfix/sql/sql-aliases.cf
    proxy:mysql:/etc/postfix/sql/sql-domains.cf
    proxy:mysql:/etc/postfix/sql/sql-domains-alias.cf
    proxy:mysql:/etc/postfix/sql/sql-mailboxes.cf
    proxy:mysql:/etc/postfix/sql/sql-relaydomains.cf
    proxy:mysql:/etc/postfix/sql/sql-transports.cf

virtual_mailbox_domains = proxy:mysql:/etc/postfix/sql/sql-domains.cf
virtual_alias_domains = proxy:mysql:/etc/postfix/sql/sql-domains-alias.cf
virtual_alias_maps =
    proxy:mysql:/etc/postfix/sql/sql-aliases.cf
    proxy:mysql:/etc/postfix/sql/sql-mailboxes.cf

relay_domains = proxy:mysql:/etc/postfix/sql/sql-relaydomains.cf
transport_maps = proxy:mysql:/etc/postfix/sql/sql-transports.cf
```

## 7. Créer un répertoire pour stocker les requêtes SQL de postfix

```
mkdir /etc/postfix/sql
cd /etc/postfix/sql
```

## 8. Nous allons à présent créer plusieurs fichiers sql:

### 1. **nano** sql-aliases.cf

Contenu:

```
# Retourne la destination d'un alias
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
query = select destination from aliases a inner join domains b on
a.t_domains = b.id where CONCAT(address,'@',b.domain) = '%s' and
a.active=1 and b.active=1
```

### 2. **nano** sql-domains.cf

Contenu:

```
# Affiche les domaines autorisés
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
query = SELECT domain FROM domains WHERE domain='%s' AND type='0'
AND active=1
```

### 3. **nano** sql-domains-alias.cf

Contenu:

```
# Affiche les domaines autorisés
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
query = select destination from aliases where address='%s' and
active=1
```

### 4. **nano** sql-mailboxes.cf

Contenu:

```
# Retourne la destination d'un alias
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
```

```
query = select concat(a.user,'@',b.domain) from mailboxes a inner
join domains b on a.t_domains = b.id where
CONCAT(a.user,'@',b.domain) = '%s' and a.active=1 and b.active=1
```

#### 5. **nano** sql-relaydomains.cf

Contenu:

```
# # Retourne si un relay est autorisé
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
query = select domain from domains where type in ('1','2','3') and
active =1 and domain='%s'
```

#### 6. **nano** sql-transports.cf

Contenu:

```
# # # Retourne le transport à utiliser
user = posogodo-ro
password = Password
dbname = posogodo
hosts = 127.0.0.1
query = select destination from relay_transports a inner join
domains b on b.id = a.t_domains where b.domain='%s' and a.active =
1 and b.active = 1 and b.type in ('2','3')
```

#### 9. Modifier les droits du répertoire

```
chown root:postfix /etc/postfix/sql -R
chmod 650 /etc/postfix/sql -R
```

#### 10. Redémarrer postfix

```
systemctl restart postfix
```

## Dovecot

### Install & config

#### 1. Installer les paquets

```
apt install dovecot-imapd dovecot-pop3d dovecot-mysql
```

#### 2. Créer un groupe et un utilisateur vmail

```
groupadd -g 6000 vmail
useradd -g vmail -u 6000 vmail -d /srv/vmail -m
```

### 3. Editer le fichier dovecot.conf

```
nano /etc/dovecot/dovecot.conf
```

Modifications à apporter:

```
listen = *, ::

service stats {
    unix_listener stats-reader {
        user = vmail
        group = vmail
        mode = 0660
    }

    unix_listener stats-writer {
        user = vmail
        group = vmail
        mode = 0660
    }
}
```

### 4. Editer le fichier auth-system

```
nano /etc/dovecot/conf.d/auth-system.conf.ext
```

Et commenter tout le fichier. Autrement ceci peut affecter la rapidité du serveur.

### 5. Editer le fichier auth

```
nano /etc/dovecot/conf.d/10-auth.conf
```

Contenu à modifier:

```
disable_plaintext_auth = yes
auth_mechanisms = plain login
!include auth-sql.conf.ext
```

### 6. Editer le fichier auth-sql-conf

```
nano /etc/dovecot/conf.d/auth-sql.conf.ext
```

Voici le contenu:

```
# %u – username
# %n – user part in user@domain, same as %u if there's no domain
# %d – domain part in user@domain, empty if there's no domain
# %h – home directory
```

```
passdb {
  driver = sql
  args = /etc/dovecot/dovecot-sql.conf.ext
}
userdb {
  driver = static      ## Don't forget to change this
  args = uid=vmail gid=vmail home=/srv/vmail/%d/%n/Maildir
}
```

#### 7. Editer le fichier dovecot-sql.conf

```
nano /etc/dovecot/dovecot-sql.conf.ext
```

Et modifier ceci:

```
driver = mysql
connect = host=127.0.0.1 dbname=posogodo user=posogodo-ro
password=Password
default_pass_scheme = SHA512-CRYPT
password_query = SELECT concat(a.user,'@',b.`domain` ) as user,
password FROM posogodo.mailboxes a left join posogodo.domains b on
a.t_domains = b.id where a.active='1' and b.active='1' and
concat(a.user,'@',b.`domain` ) = '%u';
```

#### 8. Editer le fichier 10-mail.conf

```
nano /etc/dovecot/conf.d/10-mail.conf
```

Contenu:

```
mail_location = maildir:/srv/vmail/%d/%n/Maildir
namespace inbox {
  inbox = yes
}
mail_privileged_group = mail
mbox_write_locks = fcntl
```

#### 9. Editer le fichier 10-master

```
nano /etc/dovecot/conf.d/10-master.conf
```

Contenu:

```
service imap-login {
  inet_listener imap {
    port = 143
  }
  inet_listener imaps {
  }
}
service pop3-login {
```

```
inet_listener pop3 {
    port = 110
}
inet_listener pop3s {
}
}
service lmtp {
    unix_listener /var/spool/postfix/private/dovecot-lmtp {
        mode = 0600
        user = postfix
        group = postfix
    }
}
service auth {
    unix_listener /var/spool/postfix/private/auth {
        mode = 0666
        user = postfix
        group = postfix
    }
    unix_listener auth-userdb {
        mode = 0600
        user = vmail
    }
    user = dovecot
}
service auth-worker {
    user = vmail
}
service dict {
    unix_listener dict {
    }
}
}
```

10. Le fichier 10-ssl pour vos certificats:

```
nano /etc/dovecot/conf.d/10-ssl.conf
```

Contenu:

```
ssl = required
ssl_cert = </etc/letsencrypt/live/mail2.makeitsimple.be/fullchain.pem
ssl_key = </etc/letsencrypt/live/mail2.makeitsimple.be/privkey.pem
```

11. Et enfin le fichier mailbox:

```
nano /etc/dovecot/conf.d/15-mailboxes.conf
```

Contenu:

```
namespace inbox {
    # These mailboxes are widely used and could perhaps be created
```

```
automatically:
  mailbox Drafts {
    auto = subscribe
    special_use = \Drafts
  }
  mailbox Spam {
    auto = subscribe
    autoexpunge = 60d
    special_use = \Junk
  }
  mailbox Trash {
    auto = subscribe
    autoexpunge = 60d
    special_use = \Trash
  }

  # For \Sent mailboxes there are two widely used names. We'll mark
  both of
  # them as \Sent. User typically deletes one of them if duplicates are
  created.
  mailbox Sent {
    auto = subscribe
    special_use = \Sent
  }
  mailbox "Sent Messages" {
    special_use = \Sent
  }

  # If you have a virtual "All messages" mailbox:
  #mailbox virtual/All {
  #  special_use = \All
  #  comment = All my messages
  #}

  # If you have a virtual "Flagged" mailbox:
  #mailbox virtual/Flagged {
  #  special_use = \Flagged
  #  comment = All my flagged messages
  #}
}
```

## Sieve

1. Installer les paquets

```
apt install dovecot-sieve dovecot-managesieved
```

2. Editer le fichier 20-managesieve.conf

```
nano /etc/dovecot/conf.d/20-managesieve.conf
```

Et modifier le fichier de la sorte:

```
protocols = $protocols sieve

service managesieve-login {
  inet_listener sieve {
    port = 4190
  }
  service_count = 1

  process_min_avail = 0
  vsz_limit = 64M
}

protocol sieve {
  managesieve_max_line_length = 65536
  mail_max_userip_connections = 10
  managesieve_logout_format = bytes=%i/%o
  managesieve_implementation_string = Dovecot Pigeonhole
  managesieve_max_compile_errors = 5
}
```

### 3. Editer le fichier 90-sieve.conf

```
nano /etc/dovecot/conf.d/90-sieve.conf
```

et modifier le fichier

```
plugin {
  sieve= /srv/vmail/%d/%n/sieve/.dovecot.sieve
  sieve_dir = /srv/vmail/%d/%n/sieve
}
```

### 4. Editer le fichier 20-lmtp.conf

```
nano /etc/dovecot/conf.d/20-lmtp.conf
```

et rajouter **sieve** après \$mail\_plugins

### 5. Editer le fichier 20-imap.conf

```
nano /etc/dovecot/conf.d/20-imap.conf
```

et rajouter **imap-sieve** après \$mail\_plugins

### 6. Pour supporter Sieve dans SOGo, editer le fichier sogo.conf

```
nano /etc/sogo/sogo.conf
```

et rajouter les deux lignes suivantes:

```
NGImap4ConnectionStringSeparator = ".";  
SOGOSieveServer = "sieve://127.0.0.1:4190";
```

## Quota

1. Dans le fichier 10-mail.conf rajouter ceci

```
mail_plugins = quota
```

2. Dans le fichier 20-imap.conf rajouter

```
protocol imap {  
    mail_plugins = $mail_plugins imap_sieve antispam imap_quota  
}
```

3. Dans le fichier 20-lmtp.conf rajouter

```
protocol lmtp {  
    mail_plugins = $mail_plugins sieve quota  
}
```

4. Dans le fichier 90-quota.conf modifier

```
plugin {  
    quota_warning = storage=95%% quota-warning 95 %u  
    quota_warning2 = storage=80%% quota-warning 80 %u  
}  
service quota-warning {  
    executable = script /usr/local/bin/quota-warning.sh  
    user = dovecot  
    unix_listener quota-warning {  
        user = vmail  
    }  
}  
plugin {  
    #quota = dirsize:User quota  
    #quota = maildir:User quota  
    #quota = dict:User quota::proxy::quota  
    quota = count:User quota  
    #quota = fs:User quota  
    quota_vsizes = yes  
}
```

5. Dans le fichier conf.d/auth-sql.conf.ext

```
passdb {  
    driver = sql  
  
    # Path for SQL configuration file, see example-config/dovecot-  
    sql.conf.ext
```

```
args = /etc/dovecot/dovecot-sql.conf.ext
}

# "prefetch" user database means that the passdb already provided the
# needed information and there's no need to do a separate userdb
lookup.
# <doc/wiki/UserDatabase.Prefetch.txt>
#userdb {
# driver = prefetch
#}

userdb {
driver = prefetch
}

userdb {
  driver = sql
  args = /etc/dovecot/dovecot-sql.conf.ext
}
```

6. Dans le fichier 10-mail.conf avec:

```
mail_uid = 6000
mail_gid = 6000
```

7. Dans le fichier /etc/dovecot/dovecot-sql.conf.ext modifier/ajouter les deux lignes suivantes:

```
password_query = SELECT concat(a.user, '@', b.`domain` ) as user,
password, CONCAT('*:storage=', quota) AS quota_rule FROM
posogodo.mailboxes a left join posogodo.domains b on a.t_domains = b.id
where a.active='1' and b.active='1' and concat(a.user, '@', b.`domain` )
= '%u';
user_query = SELECT concat(a.user, '@', b.`domain` ) as
user, concat('/srv/vmail', '/', b.domain, '/', a.user) as home, password,
CONCAT('*:storage=', quota) AS quota_rule FROM posogodo.mailboxes a
left join posogodo.domains b on a.t_domains = b.id where a.active='1'
and b.active='1' and concat(a.user, '@', b.`domain` ) = '%u';
```

8. Faire un script d'alerte pour dépassement de quota:

```
nano /usr/local/bin/quota-warning.sh
```

Avec le contenu suivant:

[quota-warning.sh](#)

```
#!/bin/bash
PERCENT=$1
USER=$2
cat << EOF | /usr/sbin/sendmail $USER -0
"plugin/quota=maildir:User quota:noenforcing"
```

```
From: postmaster@makeitsimple.be
Subject: quota warning
```

```
Attention: Votre boite email est pleine à $PERCENT%.
Passé 100% il ne sera plus possible de recevoir du courrier.
Veuillez prendre les actions nécessaires pour nettoyer votre boite
ou prendre un abonnement plus important.
```

```
EOF
```

#### 9. Donner les droits nécessaires à ce fichier

```
chown vmail:vmail /usr/local/bin/quota-warning.sh
chmod +x /usr/local/bin/quota-warning.sh
```

#### 10. Quelques commandes de debugging:

1. dovecadm mailbox status -u vincent@x.org vsize '\*'
2. dovecadm quota recal
3. dovecadm quota get -u vincent@x.org

#### 11. Pour plus d'infos, dans le fichier 10-loggin mettre mail-debug & auth-debug sur true

## rspamd

Petite note sur rspamd:

- les configs ajoutées dans `local.d` remplacent tout un fichier de configuration.
- les configs ajoutées dans `override.d` remplacent juste les paramètres indiqués.

## Installer rspamd

#### 1. Installer les paquets:

```
apt install redis-server software-properties-common lsb-release
```

#### 2. Ajouter la clé gpg du dépôt rspamd:

```
wget -O- https://rspamd.com/apt-stable/gpg.key | apt-key add -
echo "deb http://rspamd.com/apt-stable/ $(lsb_release -cs) main" | tee
-a /etc/apt/sources.list.d/rspamd.list
```

#### 3. Mettre à jour & installer rspamd:

```
apt update
apt install rspamd
```

#### 4. Modification du port d'écoute:

```
nano /etc/rspamd/local.d/worker-normal.inc
```

## Contenu

```
bind_socket = "127.0.0.1:11333";
```

### 5. Modification du port milter

```
nano /etc/rspamd/local.d/worker-proxy.inc
```

#### Contenu:

```
bind_socket = "127.0.0.1:11332";
milter = yes;
timeout = 120s;
upstream "local" {
    default = yes;
    self_scan = yes;
}
```

### 6. Nous allons à présent créer un code pour l'interface de gestion rspamd:

```
rspamadm pw --encrypt -p Password
```

Garder ce code pour l'étape suivante

### 7. Editer le fichier contenant le mot de passe:

```
nano /etc/rspamd/local.d/worker-controller.inc
```

#### Contenu:

```
password = "lecode-encrypté"
```

### 8. Editer le fichier classifieur-bayes:

```
nano /etc/rspamd/local.d/classifieur-bayes.conf
```

#### Contenu:

```
servers = "127.0.0.1";
backend = "redis";
autolearn = true;
```

### 9. Editer le fichier milter\_headers:

```
nano /etc/rspamd/local.d/milter_headers.conf
```

#### Contenu:

```
extended_spam_headers = true;
skip_local = false;
skip_authenticated = false;
use = ["spam-header"];
```

```
routines {
    spam-header {
#     header = "X-Spam-Flag";
#     value = "Yes";
        header = "X-Spam";
        value = "Yes";
    }
}
```

10. Redémarrer rspamd:

```
systemctl restart rspamd
```

11. Ajouter un reverse proxy dans nginx:

```
nano /etc/nginx/sites-enabled/default
```

Contenu:

```
location ^~ /rspamd/ {
    proxy_pass http://127.0.0.1:11334/;
    proxy_set_header Host $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
}
```

12. Dans 20-lmtp.conf, activer Sieve pour le lmtp

```
/etc/dovecot/conf.d/20-lmtp.conf
```

Contenu:

```
protocol lmtp {
    # Space separated list of plugins to load (default is global
    mail_plugins).
    mail_plugins = $mail_plugins sieve
}
```

13. Lier rspamd à postfix:

```
postconf smtpd_milters=inet:127.0.0.1:11332
postconf non_smtpd_milters=inet:127.0.0.1:11332
postconf milter_protocol=6
postconf milter_mail_macros="i {mail_addr} {client_addr} {client_name}
{auth_authen}"
postconf -e "milter_default_action = accept"
service postfix restart
```

14. Pour tester la config:

```
Pour tester la config
rspamadm configtest
```

```
rspamadm configdump
```

15. Nous allons mettre en place un filtre en dovecot pour déplacer les spams dans le bon dossier

```
nano /etc/dovecot/conf.d/90-sieve.conf
```

Rajouter

```
sieve_after = /etc/dovecot/sieve-after/
```

16. Créer le répertoire ainsi que le fichier avec la règle sieve:

```
mkdir /etc/dovecot/sieve-after  
nano /etc/dovecot/sieve-after/spam-to-folder.sieve
```

Contenu:

```
require ["fileinto","mailbox"];  
  
if header :contains "X-Spam" "Yes" {  
    fileinto :create "Junk";  
    stop;  
}
```

17. Compiler la règle:

```
sievec /etc/dovecot/sieve-after/spam-to-folder.sieve  
service dovecot restart
```

18. Création de deux règles pour whitelister les domaines ou les ips. Editer le fichier multimap.conf

```
nano /etc/rspamd/local.d/multimap.conf
```

Et coller le contenu suivant

```
WHITELIST_SENDER_DOMAIN {  
    type = "from";  
    filter = "email:domain";  
    map = "/etc/rspamd/local.d/whitelist.sender.domain.map";  
    score = -10.0  
}  
  
IP_WHITELIST {  
    type = "ip";  
    prefilter = "true";  
    map = "/etc/rspamd/local.d/whitelist.ip.map";  
    action = "accept";  
}
```

1. Créer ensuite un fichier whitelist.sender.domain.map avec les domaines/emails à whitelister
2. Créer aussi un fichier whitelist.ip.map pour autoriser des ips spécifiques (j'utilise

127.0.0.1 pour éviter que les quota warning ne soient flaggués SPAM)

## Signer DKIM

1. Créer un répertoire pour stocker les clés:

```
mkdir /var/lib/rspamd/dkim/
```

2. Créer un fichier dkim\_signing

```
nano /etc/rspamd/local.d/dkim_signing.conf
```

Avec ceci:

```
# If false, messages with empty envelope from are not signed
allow_envfrom_empty = true;

# If true, envelope/header domain mismatch is ignored
allow_hdrfrom_mismatch = false;

# If true, multiple from headers are allowed (but only first is used)
allow_hdrfrom_multiple = false;

# If true, username does not need to contain matching domain
allow_username_mismatch = false;

# Default path to key, can include '$domain' and '$selector' variables
path = "/var/lib/rspamd/dkim/$domain.$selector.key";

# Default selector to use
selector = "dkim";

# If false, messages from authenticated users are not selected for
signing
sign_authenticated = true;

# If false, messages from local networks are not selected for signing
sign_local = true;

# Map file of IP addresses/subnets to consider for signing
# sign_networks = "/some/file"; # or url

# Symbol to add when message is signed
symbol = "DKIM_SIGNED";

# Whether to fallback to global config
try_fallback = true;

# Domain to use for DKIM signing: can be "header" (MIME From),
"envelope" (SMTP From) or "auth" (SMTP username)
```

```
use_domain = "header";

# Domain to use for DKIM signing when sender is in sign_networks
("header"/"envelope"/"auth")
#use_domain_sign_networks = "header";

# Domain to use for DKIM signing when sender is a local IP
("header"/"envelope"/"auth")
#use_domain_sign_local = "header";

# Whether to normalise domains to eSLD
use_esld = true;

# Whether to get keys from Redis
use_redis = false;

# Hash for DKIM keys in Redis
key_prefix = "DKIM_KEYS";

# map of domains -> names of selectors (since rspamd 1.5.3)
#selector_map = "/etc/rspamd/dkim_selectors.map";

# map of domains -> paths to keys (since rspamd 1.5.3)
#path_map = "/etc/rspamd/dkim_paths.map";

# If `true` get pubkey from DNS record and check if it matches private
key
check_pubkey = false;
# Set to `false` if you want to skip signing if public and private keys
mismatch
allow_pubkey_mismatch = true;
```

### 3. Créer une clé par domaine:

```
rspamadm dkim_keygen -s 'dkim' -b 2048 -d domaine.net -k
/var/lib/rspamd/dkim/domaine.net.dkim.key > domaine.net.txt
```

Dans le fichier .txt vous trouverez la configuration à appliquer dans votre zone DNS avec le sous domaine dkim.\_domainkey

## Apprentissage des spams dans dovecot

### 1. Editer le fichier 20-imap.conf

```
nano /etc/dovecot/conf.d/20-imap.conf
```

et rajouter **imap\_sieve** à la hauteur de mail\_plugins.

### 2. Editer le fichier 90-sieve.conf

```
nano /etc/dovecot/conf.d/90-sieve.conf
```

et rajouter **sieve\_plugins = sieve\_imapsieve sieve\_extprograms** dans le block "plugin".

### 3. Editer le fichier 90-plugin.conf

```
nano /etc/dovecot/conf.d/90-plugin.conf
```

Et ajouter les lignes suivantes:

```
# From elsewhere to Junk folder
imapsieve_mailbox1_name = Junk
imapsieve_mailbox1_causes = COPY
imapsieve_mailbox1_before = file:/etc/dovecot/sieve/learn-spam.sieve

# From Junk folder to elsewhere
imapsieve_mailbox2_name = *
imapsieve_mailbox2_from = Junk
imapsieve_mailbox2_causes = COPY
imapsieve_mailbox2_before = file:/etc/dovecot/sieve/learn-ham.sieve

sieve_pipe_bin_dir = /etc/dovecot/sieve
sieve_global_extensions = +vnd.dovecot.pipe
```

### 4. Créer un dossier de config sieve

```
mkdir /etc/dovecot/sieve
```

### 5. Ensuite créer un fichier learn-spam.sieve

```
nano /etc/dovecot/sieve/learn-spam.sieve
```

dont le contenu est:

```
require ["vnd.dovecot.pipe", "copy", "imapsieve"];
pipe :copy "rspamd-learn-spam.sh";
```

### 6. Créer un fichier learn-ham.sieve

```
nano /etc/dovecot/sieve/learn-ham.sieve
```

dont le contenu est:

```
require ["vnd.dovecot.pipe", "copy", "imapsieve"];
pipe :copy "rspamd-learn-ham.sh";
```

### 7. Compiler les scripts sieve et appliquer les droits

```
sievec /etc/dovecot/sieve/learn-spam.sieve
sievec /etc/dovecot/sieve/learn-ham.sieve
chmod u=rw,go= /etc/dovecot/sieve/learn-{spam,ham}.sieve
```

```
chown vmail.vmail /etc/dovecot/sieve/learn-{spam,ham}.sieve
```

#### 8. Créer un fichier rspamd-learn-spam.sh

```
nano /etc/dovecot/sieve/rspamd-learn-spam.sh
```

dont le contenu est:

```
#!/bin/sh  
exec /usr/bin/rspamd learn_spam -h localhost:11334
```

#### 9. et un fichier rspamd-learn-ham.sh

```
nano /etc/dovecot/sieve/rspamd-learn-ham.sh
```

dont le contenu est:

```
#!/bin/sh  
exec /usr/bin/rspamd learn_ham -h localhost:11334
```

#### 10. Changer les droits et redémarrer dovecot

```
chmod u=rwx,go= /etc/dovecot/sieve/rspamd-learn-{spam,ham}.sh  
chown vmail.vmail /etc/dovecot/sieve/rspamd-learn-{spam,ham}.sh  
service dovecot restart
```

## Filtrer les virus

#### 1. Installer les paquets clamav:

```
apt install clamav clamav-daemon
```

#### 2. Vérifier qu'un cron tourne pour mettre à jour via freshclam

#### 3. Editer le fichier de config

```
nano /etc/rspamd/local.d/antivirus.conf
```

Et mettre le code suivant:

```
clamav {  
    # If set force this action if any virus is found (default unset: no  
    action is forced)  
    # action = "reject";  
    # message = '${SCANNER}: virus found: "${VIRUS}";'  
    # Scan mime_parts seperately - otherwise the complete mail will be  
    transfered to AV Scanner  
    #attachments_only = true; # Before 1.8.1  
    #scan_mime_parts = true; # After 1.8.1  
    # Scanning Text is suitable for some av scanner databases (e.g.  
    Sanesecurity)
```

```
#scan_text_mime = false; # 1.8.1 +
#scan_image_mime = false; # 1.8.1 +
# If `max_size` is set, messages > n bytes in size are not scanned
#max_size = 20000000;
# symbol to add (add it to metric if you want non-zero weight)
symbol = "CLAM_VIRUS";
# type of scanner: "clamav", "fprot", "sophos" or "savapi"
type = "clamav";
# If set true, log message is emitted for clean messages
#log_clean = false;
# Prefix used for caching in Redis: scanner-specific defaults are
used. If Redis is enabled and
# multiple scanners of the same type are present, it is important to
set prefix to something unique.
#prefix = "rs_cl_";
# For "savapi" you must also specify the following variable
#product_id = 12345;
# servers to query (if port is unspecified, scanner-specific default
is used)
# can be specified multiple times to pool servers
# can be set to a path to a unix socket
servers = "127.0.0.1:3310";
# if `patterns` is specified virus name will be matched against
provided regexes and the related
# symbol will be yielded if a match is found. If no match is found,
default symbol is yielded.
patterns {
    # symbol_name = "pattern";
    JUST_EICAR = '^Eicar-Test-Signature$';
}
# In version 1.7.0+ patterns could be extended
#patterns = {SANE_MAL = 'Sanesecurity\.Malware\.*', CLAM_UNOFFICIAL =
'UNOFFICIAL$'};
# `whitelist` points to a map of IP addresses. Mail from these
addresses is not scanned.
whitelist = "/etc/rspamd/antivirus.wl";
}
```

## SOGo

1. Installer la clé gpg du dépôt SOGo:

```
gpg --keyserver hkp://pgp.mit.edu --recv-key 0x810273C4
gpg --armor --export 0x810273C4 | apt-key add -
```

2. Créer un dépôt

```
nano /etc/apt/sources.list.d/sogo.list
```

Avec le contenu:

```
# Commercial
#deb
https://<username>:<password>@packages.inverse.ca/SOGo/release/2/debian
/ buster buster
# Non-Commercial
deb http://packages.inverse.ca/SOGo/nightly/5/debian/ buster buster
```

3. Rafraîchir les dépôts et installer SOGo

```
apt install sogo sogo-activesync memcached
```

4. Se connecter sur le serveur MySQL:

```
mysql -u root -p
```

et faire les commandes suivantes:

```
CREATE DATABASE sogo;
CREATE USER 'sogo'@'localhost' IDENTIFIED BY 'Password';
GRANT ALL privileges ON sogo.* TO 'sogo'@'localhost';
USE posogodo;
CREATE VIEW sogo.sogo_view AS SELECT concat(mailboxes.user,'@',
domains.domain) AS c_uid, concat(mailboxes.user,'@', domains.domain) AS
c_name, domains.domain AS c_domain, concat(mailboxes.user,'@',
domains.domain) AS mail, CONCAT(firstname, ' ',lastname) AS c_cn,
mailboxes.password AS c_password, mailboxes.sogo_kind AS c_kind,
mailboxes.sogo_multibooking AS c_multibooking FROM mailboxes LEFT JOIN
domains ON mailboxes.t_domains = domains.id WHERE mailboxes.active=1 &
domains.active=1 AND mailboxes.sogo_active =1 ;
```

5. Editer le fichier sogo.conf:

```
nano /etc/sogo/sogo.conf
```

Et mettre un contenu similaire

```
{
/* ***** Main SOGo configuration file
*****
*
*
* Since the content of this file is a dictionary in OpenStep plist
format, *
* the curly braces enclosing the body of the configuration are
mandatory. *
* See the Installation Guide for details on the format.
*
*
* C and C++ style comments are supported.
```

```
*
*
*
* This example configuration contains only a subset of all available
*
* configuration parameters. Please see the installation guide more
details. *
*
*
* ~sogo/GNUstep/Defaults/.GNUstepDefaults has precedence over this
file, *
* make sure to move it away to avoid unwanted parameter overrides.
*
*
*
*****
***/

/* Database configuration (mysql://, postgresql:// or oracle://) */
//S0GoProfileURL =
"postgresql://sogo:sogo@localhost:5432/sogo/sogo_user_profile";
//OCSFolderInfoURL =
"postgresql://sogo:sogo@localhost:5432/sogo/sogo_folder_info";
//OCSSessionsFolderURL =
"postgresql://sogo:sogo@localhost:5432/sogo/sogo_sessions_folder";

S0GoProfileURL =
"mysql://sogo:Password@127.0.0.1:3306/sogo/sogo_user_profile";
OCSFolderInfoURL =
"mysql://sogo:Password@127.0.0.1:3306/sogo/sogo_folder_info";
OCSSessionsFolderURL =
"mysql://sogo:Password@127.0.0.1:3306/sogo/sogo_sessions_folder";

/* Mail */
S0GoDraftsFolderName = Drafts;
S0GoSentFolderName = Sent;
S0GoTrashFolderName = Trash;
S0GoJunkFolderName = Junk;
S0GoIMAPServer = "localhost";
//S0GoSieveServer = "sieve://127.0.0.1:4190";
S0GoSMTPServer = "smtp://127.0.0.1";
//S0GoMailDomain = acme.com;
S0GoMailingMechanism = smtp;
//S0GoForceExternalLoginWithEmail = NO;
//S0GoMailSpoolPath = /var/spool/sogo;
//Le paramètre suivant est important pour la génération de filtres
Sieve
//NGImap4ConnectionStringSeparator = ".";
```

```
/* Notifications */
//S0GoAppointmentSendEMailNotifications = NO;
//S0GoACLsSendEMailNotifications = NO;
//S0GoFoldersSendEMailNotifications = NO;

/* Authentication */
//S0GoPasswordChangeEnabled = YES;

/* LDAP authentication example */
//S0GoUserSources = (
// {
//     type = ldap;
//     CNFieldName = cn;
//     UIDFieldName = uid;
//     IDFieldName = uid; // first field of the DN for direct binds
//     bindFields = (uid, mail); // array of fields to use for
indirect binds
//     baseDN = "ou=users,dc=acme,dc=com";
//     bindDN = "uid=sogo,ou=users,dc=acme,dc=com";
//     bindPassword = qwerty;
//     canAuthenticate = YES;
//     displayName = "Shared Addresses";
//     hostname = "ldap://127.0.0.1:389";
//     id = public;
//     isAddressBook = YES;
// }
//);

/* LDAP AD/Samba4 example */
//S0GoUserSources = (
// {
//     type = ldap;
//     CNFieldName = cn;
//     UIDFieldName = sAMAccountName;
//     baseDN = "CN=users,dc=domain,dc=tld";
//     bindDN = "CN=sogo,CN=users,DC=domain,DC=tld";
//     bindFields = (sAMAccountName, mail);
//     bindPassword = password;
//     canAuthenticate = YES;
//     displayName = "Public";
//     hostname = "ldap://127.0.0.1:389";
//     filter = "mail = '*'";
//     id = directory;
//     isAddressBook = YES;
// }
//);

/* SQL authentication example */
/* These database columns MUST be present in the view/table:
*     c_uid - will be used for authentication - it's the username or
```

```
username@domain.tld)
*   c_name - which can be identical to c_uid - will be used to
uniquely identify entries
*   c_password - password of the user, plain-text, md5 or sha
encoded for now
*   c_cn - the user's common name - such as "John Doe"
*   mail - the user's mail address
*   See the installation guide for more details
*/
S0GoUserSources =
(
  {
    type = sql;
    id = directory;
    displayName = "Annuaire";
    viewURL = "mysql://sogo:Pasword@127.0.0.1:3306/sogo/sogo_view";
    canAuthenticate = YES;
    isAddressBook = YES;
    DomainFieldName = "c_domain";
    KindFieldName = "c_kind";
    MultipleBookingsFieldName = "c_multibooking";
    userPasswordAlgorithm = sha512-crypt;
  }
);
MySQL4Encoding = "utf8mb4";
/* Web Interface */
S0GoPageTitle = S0Go;
//S0GoVacationEnabled = YES;
//S0GoForwardEnabled = YES;
//S0GoSieveScriptsEnabled = YES;
//S0GoMailAuxiliaryUserAccountsEnabled = YES;
//S0GoTrustProxyAuthentication = NO;
//S0GoXSRFValidationEnabled = YES;

/* General - S0GoTimeZone *MUST* be defined */
S0GoLanguage = French;
S0GoTimeZone = Europe/Brussels;
S0GoCalendarDefaultRoles = (
  PublicDAndTVviewer,
  ConfidentialDAndTVviewer
);
//S0GoSuperUsernames = (sogo1, sogo2); // This is an array - keep the
parens!
//SxVMemLimit = 384;
//WOPidFile = "/var/run/sogo/sogo.pid";
S0GoMemcachedHost = "127.0.0.1";
/* Debug */
S0GoDebugRequests = YES;
SoDebugBaseURL = YES;
//ImapDebugEnabled = YES;
//LDAPDebugEnabled = YES;
```

```
//PGDebugEnabled = YES;
//S0GoEASDebugEnabled = YES;
MySQL4DebugEnabled = YES;
//S0GoUIxDebugEnabled = YES;
//W0DontZipResponse = YES;
W0LogFile = /var/log/sogo/sogo.log;
}
```

## 6. Editer nginx

```
nano /etc/nginx/sites-enabled/default
```

Voici un exemple de contenu

```
location ^~/S0Go
{
    proxy_pass 'http://127.0.0.1:20000';
    proxy_redirect 'http://127.0.0.1:20000' default;
    # forward user's IP address
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $host;
    proxy_set_header x-webobjects-server-protocol HTTP/1.0;
    proxy_set_header x-webobjects-remote-host 127.0.0.1;
    proxy_set_header x-webobjects-server-name $server_name;
    proxy_set_header x-webobjects-server-url $scheme://$host;
    proxy_set_header x-webobjects-server-port $server_port;
    proxy_connect_timeout 90;
    proxy_send_timeout 90;
    proxy_read_timeout 90;
    proxy_buffer_size 4k;
    proxy_buffers 4 32k;
    proxy_busy_buffers_size 64k;
    proxy_temp_file_write_size 64k;
    break;
}
location /S0Go.woa/WebServerResources/
{
    alias /usr/lib/GNUstep/S0Go/WebServerResources/;
    allow all;
    expires max;
}

location /S0Go/WebServerResources/
{
    alias /usr/lib/GNUstep/S0Go/WebServerResources/;
    allow all;
    expires max;
}

location (^/S0Go/so/ControlPanel/Products/([^/]*)/Resources/(.*)$)
```

```
{
    alias /usr/lib/GNUstep/SOGo/$1.SOGo/Resources/$2;
    expires max;
}

location
(/SOGo/so/ControlPanel/Products/[^/]*UI/Resources/.*\.(jpg|png|gif|css
|js)$)
{
    alias /usr/lib/GNUstep/SOGo/$1.SOGo/Resources/$2;
    expires max;
}
location ^~ /Microsoft-Server-ActiveSync
{
    access_log /var/log/nginx/activesync.log;
    error_log /var/log/nginx/activesync-error.log;
    resolver localhost;
    proxy_connect_timeout 4000;
    proxy_send_timeout 4000;
    proxy_read_timeout 4000;
    proxy_buffers 64 256k;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_pass
http://127.0.0.1:20000/SOGo/Microsoft-Server-ActiveSync;
    proxy_redirect
http://127.0.0.1:20000/SOGo/Microsoft-Server-ActiveSync /;
}
```

## 7. Redémarrer postfix & nginx:

```
service postfix restart
service sogo restart
```

## Sources

- Postfix:
  - <https://computingforgeeks.com/setup-mail-server-on-centos-with-postfix-dovecot-mysql-roundcube/>
  - <https://www.linode.com/docs/email/postfix/email-with-postfix-dovecot-and-mysql/>
  - <https://computingforgeeks.com/setup-mail-server-on-centos-with-postfix-dovecot-mysql-roundcube/>
- Dovecot:
  - <https://kaworu.ch/blog/2014/03/25/dovecot-antispam-with-rspamd/>
- Rspamd:
  - <https://workaround.org/ispmail/stretch/filtering-out-spam-with-rspamd>
  - <https://blog.debugo.fr/serveur-messagerie-rspamd/>
  - <https://github.com/rspamd/rspamd/issues/3078>

- Sogo:
  - <https://forum.iredmail.org/topic10132-iredmail-support-a-lof-of-error-such-as-no-child-available-to-handle-incoming-request.html>
  - <https://forum.iredmail.org/topic10785-iredmail-support-sogo-problem-with-activesync-outlook-2016.html>
  - <https://www.mail-archive.com/users@sogo.nu/msg28614.html>
  - <https://marc.info/?l=sogo-users&m=145570889316335&w=2>
  - <https://forum.zentyal.org/index.php?topic=33233.0>

From:  
<https://wiki.makeitsimple.be/> - **makeITsimple wiki**

Permanent link:  
[https://wiki.makeitsimple.be/doku.php?id=linux\\_mail\\_postfix\\_dovecot\\_sogo](https://wiki.makeitsimple.be/doku.php?id=linux_mail_postfix_dovecot_sogo)

Last update: **2022/03/27 12:20**

