Set up a mail server with PostfixAdmin and MariaDB on CentOS 7

Update the system and install necessary packages

```
yum update && yum install wget nano
```

Create system user

For security reasons, we will create a new system user who will be the owner of all mailboxes.

```
useradd -r -u 150 -g mail -d /var/vmail -s /sbin/nologin -c "Virtual Mail User" vmail
```

```
mkdir -p /var/vmail
```

```
chmod -R 770 /var/vmail
```

```
chown -R vmail:mail /var/vmail
```

Install MariaDB

MariaDB 5.5 is shipped in the default CentOS 7 repository, to install it just run:

```
yum install mariadb-server
```

To start the MariaDB service and enable it to start on boot, execute the following commands:

```
systemctl start mariadb.service
```

```
systemctl enable mariadb.service
```

Run the following command to secure your MariaDB installation:

```
mysql_secure_installation
```

Next, we need to create a database for our postfixadminHQ instance.

```
mysql -uroot -p
```

```
MariaDB [(none)]> CREATE DATABASE postfixadmin;
```

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON postfixadmin.* TO 'postfixadmin'@'localhost' IDENTIFIED BY 'strong_password';
```

```
MariaDB [(none)]> FLUSH PRIVILEGES;
```

```
MariaDB [(none)]> \q
```

Install PHP and all necessary PHP modules

CentOS 7 ships with PHP version 5.4, to install PHP and necessary modules, run:

```
yum install php php-mysql php-imap php-mbstring php-common
```

If you don’t have Apache installed, install it with:

```
yum install httpd
```

Install PostfixAdmin

The latest version of PostfixAdmin, version 3, supports **MySQL/MariaDB**, **PostgreSQL**, and SQLite databases. In this guide, we will use MariaDB.
Download the PostfixAdmin archive from SourceForge and extract it in the /var/www/html/ directory:

```
wget -q -O - "https://downloads.sourceforge.net/project/postfixadmin/postfixadmin/postfixadmin-3.0.2/postfixadmin-3.0.2.tar.gz" | tar -xzf -
-C /var/www/html
```

Open the mail configuration file and edit the following values:

```
nano /var/www/html/postfixadmin-3.0.2/config.inc.php
```

```
$CONF['configured'] = true;
```

```
$CONF['database_type'] = 'mysqli';

$CONF['database_host'] = 'localhost';

$CONF['database_user'] = 'postfixadmin';

$CONF['database_password'] = 'strong_password';

$CONF['database_name'] = 'postfixadmin';




$CONF['domain_path'] = 'NO';

$CONF['domain_in_mailbox'] = 'YES';

chown -R apache: /var/www/html/postfixadmin-3.0.2
```

##

vim /var/www/html/postfixadmin/config.inc.php

$CONF[‘configured’] = true;
$CONF[‘setup_password’] = ‘YOUR-STRONG-PASSWORD’;

```
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'postfixadmin';
$CONF['database_password'] = 'yourPASSword';
$CONF['database_name'] = 'postfixadmin';
$CONF['show_password'] = 'YES';
$CONF['page_size'] = '30';
$CONF['default_aliases'] = array (
'abuse' => 'abuse@ceae.info',
'hostmaster' => 'hostmaster@worldcm.net',
'postmaster' => 'postmaster@ worldcm.net',
'webmaster' => 'webmaster@ worldcm.net'
);
$CONF['domain_path'] = 'NO';
$CONF['domain_in_mailbox'] = 'YES';
$CONF['maildir_name_hook'] = 'NO';
$CONF['transport'] = 'YES';
$CONF['vacation'] = 'YES';
$CONF['vacation_domain'] = 'autoreply. worldcm.net';
$CONF['vacation_control'] ='YES';
```

If your domain do not exist, activate this

```
$CONF['emailcheck_resolve_domain']='NO';
```

##

To populate the database go to `https://Your_IP_Address/postfixadmin-3.0.2/setup.php` and you should see something like below:

```
Testing database connection - OK - mysqli://postfixadmin:xxxxx@localhost/postfixadmin
Everything seems fine... attempting to create/update database structure
```

Create a new admin user:

```
bash /var/www/html/postfixadmin-3.0.2/scripts/postfixadmin-cli admin add admin@your_domain_name.com --password strong_password22 --password2
strong_password22 --superadmin 1 --active 1
```

Install and configure postfix

To install postfix run the command bellow:

```
yum install postfix
```

Once the installation is completed, we need to create configuration files:

```
mkdir -p /etc/postfix/sql/
```

```
nano /etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf
```

```
user = postfixadmin
```

```
password = strong_password
```

```
hosts = localhost
```

```
dbname = postfixadmin
```

```
query  = SELECT goto FROM alias,alias_domain WHERE alias_domain.alias_domain = '%d' and alias.address = CONCAT('@',
alias_domain.target_domain) AND alias.active = 1 AND alias_domain.active='1'
```

```
nano /etc/postfix/sql/mysql_virtual_alias_domain_mailbox_maps.cf
```

```
user = postfixadmin
```

```
password = strong_password
```

```
hosts = localhost
```

```
dbname = postfixadmin
```

```
query = SELECT maildir FROM mailbox,alias_domain WHERE alias_domain.alias_domain = '%d' and mailbox.username = CONCAT('%u', '@',
alias_domain.target_domain) AND mailbox.active = 1 AND alias_domain.active='1'
```

```
nano /etc/postfix/sql/mysql_virtual_alias_domain_maps.cf
```

```
user = postfixadmin
```

```
password = strong_password
```

```
hosts = localhost

dbname = postfixadmin

query = SELECT goto FROM alias,alias_domain WHERE alias_domain.alias_domain = '%d' and alias.address = CONCAT('%u', '@',
alias_domain.target_domain) AND alias.active = 1 AND alias_domain.active='1'
```

nano /etc/postfix/sql/mysql_virtual_alias_maps.cf

```
user = postfixadmin

password = strong_password

hosts = localhost

dbname = postfixadmin

query = SELECT goto FROM alias WHERE address='%s' AND active = '1'

#expansion_limit = 100
```

nano /etc/postfix/sql/mysql_virtual_domains_maps.cf

```
user = postfixadmin

password = strong_password

hosts = localhost

dbname = postfixadmin

query          = SELECT domain FROM domain WHERE domain='%s' AND active = '1'

#query          = SELECT domain FROM domain WHERE domain='%s'

#optional query to use when relaying for backup MX

#query           = SELECT domain FROM domain WHERE domain='%s' AND backupmx = '0' AND active = '1'

#expansion_limit = 100
```

nano /etc/postfix/sql/mysql_virtual_mailbox_limit_maps.cf

```
user = postfixadmin
```

```
password = strong_password

hosts = localhost

dbname = postfixadmin

query = SELECT quota FROM mailbox WHERE username='%s' AND active = '1'
```

```
nano /etc/postfix/sql/mysql_virtual_mailbox_maps.cf
```

```
user = postfixadmin

password = strong_password

hosts = localhost

dbname = postfixadmin

query           = SELECT maildir FROM mailbox WHERE username='%s' AND active = '1'

#expansion_limit = 100
```

Edit the `main.cf` file:

```
postconf -e "myhostname = $(hostname -f)"
```

```
postconf -e "virtual_mailbox_domains = proxy:mysql:/etc/postfix/sql/mysql_virtual_domains_maps.cf"
```

```
postconf -e "virtual_alias_maps = proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_maps.cf,
proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_maps.cf, proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf"
```

```
postconf -e "virtual_mailbox_maps = proxy:mysql:/etc/postfix/sql/mysql_virtual_mailbox_maps.cf,
proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_mailbox_maps.cf"
```

```
postconf -e "smtpd_tls_cert_file = /etc/pki/tls/certs/localhost.crt"
```

```
postconf -e "smtpd_tls_key_file = /etc/pki/tls/private/localhost.key"
```

```
postconf -e "smtpd_use_tls = yes"
```

```
postconf -e "smtpd_tls_auth_only = yes"




postconf -e "smtpd_sasl_type = dovecot"

postconf -e "smtpd_sasl_path = private/auth"

postconf -e "smtpd_sasl_auth_enable = yes"

postconf -e "smtpd_recipient_restrictions = permit_sasl_authenticated, permit_mynetworks, reject_unauth_destination"




postconf -e "mydestination = localhost"

postconf -e "mynetworks = 127.0.0.0/8"

postconf -e "inet_protocols = ipv4"

postconf -e "inet_interfaces = all"
```

```
postconf -e "virtual_transport = lmtp:unix:private/dovecot-lmtp"
```

Open the `master.cf` file, find `submission inet n` and `smtps inet n` sections and edit as follows:

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

```
submission inet n       -       n       -       -       smtpd

  -o syslog_name=postfix/submission

  -o smtpd_tls_security_level=encrypt

  -o smtpd_sasl_auth_enable=yes

#  -o smtpd_reject_unlisted_recipient=no

#  -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

smtps     inet  n       -       n       -       -       smtpd

  -o syslog_name=postfix/smtps

#  -o smtpd_tls_wrappermode=yes

  -o smtpd_sasl_auth_enable=yes

#  -o smtpd_reject_unlisted_recipient=no

#  -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
```

Enable the postfix service

```
systemctl enable postfix

systemctl restart postfix
```

Install and Configure Dovecot

Install dovecot with MySQL support using the command bellow:

```
yum install dovecot dovecot-mysql
```

Open the `/etc/dovecot/conf.d/10-mail.conf` file and change the following values:

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

```
mail_location = maildir:/var/vmail/%d/%n


mail_privileged_group = mail


mail_uid = vmail


mail_gid = mail


first_valid_uid = 150


last_valid_uid = 150
```

Open the `/etc/dovecot/conf.d/10-auth.conf` file and change the following values:

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

auth_mechanisms = plain login


#!include auth-system.conf.ext


!include auth-sql.conf.ext
```

Create a new `dovecot-sql.conf.ext` file:

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

```
driver = mysql
```

```
connect = host=localhost dbname=postfixadmin user=postfixadmin password=strong_password
```

```
default_pass_scheme = MD5-CRYPT
```

```
password_query = SELECT username as user, password, '/var/vmail/%d/%n' as userdb_home, 'maildir:/var/vmail/%d/%n' as userdb_mail, 150 as
userdb_uid, 8 as userdb_gid FROM mailbox WHERE username = '%u' AND active = '1'
```

```
user_query = SELECT '/var/vmail/%d/%u' as home, 'maildir:/var/vmail/%d/%u' as mail, 150 AS uid, 8 AS gid, concat('dirsize:storage=',  quota)
AS quota FROM mailbox WHERE username = '%u' AND active = '1'
```

In the `/etc/dovecot/conf.d/10-ssl.conf` file enable SSL support:

```
ssl = yes
```

Open the `/etc/dovecot/conf.d/15-lda.conf` file and set the `postmaster_address` email address.

```
postmaster_address = postmaster@your_domain_name.com
```

Open the `/etc/dovecot/conf.d/10-master.conf` file, find the service lmtp section and change it to:

```
service lmtp {

  unix_listener /var/spool/postfix/private/dovecot-lmtp {

    mode = 0600

    user = postfix

    group = postfix

  }

}
```

find the service auth section and change it to:

```
service auth {

  unix_listener /var/spool/postfix/private/auth {

    mode = 0666
```

```
    user = postfix
    group = postfix
  }
  unix_listener auth-userdb {
    mode = 0600
    user = vmail
    #group = vmail
  }
  user = dovecot
}
```

Change the service auth-worker section to the following:

```
service auth-worker {

  user = vmail

}
```

Set the permissions:

```
chown -R vmail:dovecot /etc/dovecot

chmod -R o-rwx /etc/dovecot
```

Enable and restart the dovecot service

```
systemctl enable dovecot

systemctl restart dovecot
```

Install and configure Spamassassin

Install spamassassin using the command bellow:

```
yum install spamassassin
```

Create a spamassassin system user:

```
groupadd spamd
```

```
useradd -g spamd -s /bin/false -d /var/log/spamassassin spamd
```

```
chown spamd:spamd /var/log/spamassassin
```

Configure Postfix to use SpamAssassin

Open the `master.cf` file and edit as follows:

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

change

```
smtp      inet  n       -       n       -       -       smtpd
```

with

```
smtp      inet  n       -       n       -       -       smtpd -o content_filter=spamassassin
```

add the following line at the end of the file:

```
spamassassin unix - n n - - pipe flags=R user=spamd argv=/usr/bin/spamc -e /usr/sbin/sendmail -oi -f ${sender} ${recipient}
```

Enable and restart the spamassassin service

```
systemctl enable spamassassin
```

```
systemctl restart spamassassin
```

Restart the postfix service

```
systemctl restart postfix
```

If everything is set up correctly now you should be able to log in to your PostfixAdmin backend by going to `http://Your_IP_Address/postfixadmin-3.0.2.2` and create your first virtual domain and mailbox.

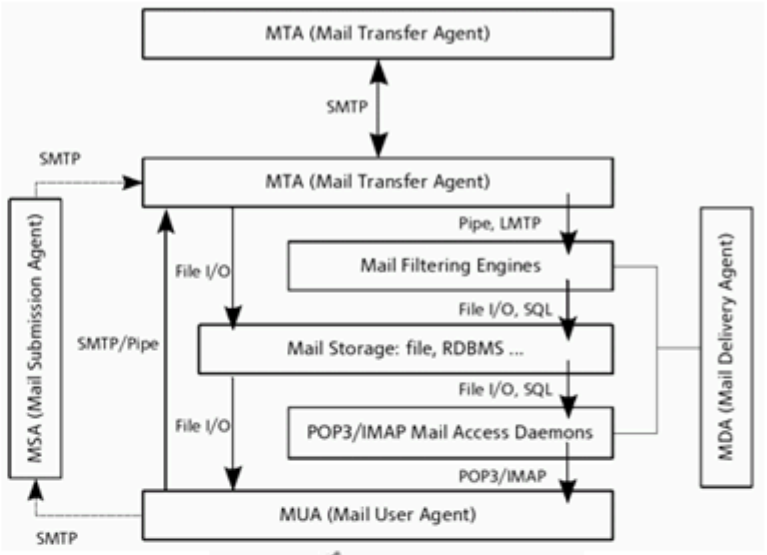
I build my own mail server with Postfix and Dovecot

https://habrahabr.ru/post/193220/

- System administration ,
- Configuring Linux
- Tutorial

```
# yum install mysql-server
# chkconfig mysqld on
# service mysqld start
# yum update postfix
# yum install dovecot dovecot-mysql
# chkconfig dovecot on
# service dovecot start
# yum install amavisd-new clamav
# yum install spamassassin
```

1. Draw some circles



2. Draw the rest of the owl

As part of the unification program installed server systems faced with the task of remaking the mail server. Thoughtful study manuals and guides showed a curious fact - was nowhere to be found uniquely authentic leadership or similarity of Best Practice for deployment mailer.

Manual step by step, is based on internal company records and affects quite obvious questions. Guru can not waste time, know-how is not here - is a

grab-bag guide and published only because of all the results by manual deployment mailer resembled a picture of how to draw the owl.

For those who do not want to collect everything by hand, the best option will probably be the package iRedMail . Excellent build Postfix, Dovecot, Apache, MySQL / PostgreSQL, Policyd, Amavis, Fail2ban, Roundcube and even Awstats. Placed easily, it is stable, has a beautiful admin panel (free) and very beautiful admin panel (fee required) is not going to any comparison with the poor PostfixAdmin. Fans of the manual labor can continue reading.

Old server is running Gentoo and was carrying a thermonuclear charge of Postfix + VDA with the Courier and buggy implemented SASL, who solved connect to mysql only the first authentication. Rework plan was to migrate to our internal standard - CentOS. The role of MTA and MDA assigned to the bundle and Postfix Dovecot, and as auxiliary artillery: Amavis + SpamAssassin + ClamAV + Postgrey + Fail2Ban. Letters are stored in files, and user accounts and domains - in MySQL. On the server, spinning several mail domains, and there is support for virtual quotas.

[*] Connecting additional repositories. I had the **epel, rpmforge, centalt and remi**. Not all of them must be constantly enabled, and you can install the plugin yum-priorities; well, or if too lazy to sort out what to include, disable their hands. Next, I will say that some of the repository set.

[*] Working with **SELinux** worthy of a separate article, so in this article we assume that selinux is lame made permissive or disabled.

[*] Do not forget about **the ntp** :

```
yum install ntp
ntpdate <ваш ntp сервер>
chkconfig ntpd on && /etc/init.d/ntpd start
```

This will avoid possible problems Dovecot'a «Time has moved backwards». Can also be in the */ etc / sysconfig / ntpd* change the key to -L, ntpd to not listen to the network.

[*] At the end of the preparatory phase we put the tools that will facilitate us to test and further work:

```
yum install wget mlocate bind-utils telnet mailx sharutils
```

For database use **MySQL 5.5 by Remi** . MariaDB is possible and, of course, but as long as MySQL is still alive, the aforementioned assembly I was completely satisfied. Version is important because when you upgrade Postfix to 2.10 he wants the new version, and if you put the 5.1 in base, then upgrade Postfix from CentALT drags MariaDB. Who prefer PgSQL - please put it. Deployment will not differ, even postfix you can take the same configuration files. Will differ only configuration of postgresql and create a database.

To start suitable embodiment "of the box" (in the repository below is slightly advanced my.cnf). Create a user **postfix** with the same base, and all the rights to it.

```
CREATE USER postfix@localhost IDENTIFIED BY 'mypassword';
CREATE DATABASE postfix;
GRANT ALL PRIVILEGES ON postfix.* TO postfix;
```

Antivirus works for me **ClamAV** . It is worth noting that the latest version is on CentALT, but she does not want to stop downloaded from there, dying in the attempt to download a 50MB clamav-db. Therefore we put out **EPEL** a couple of minor versions less weather it did not. Clam will work with us through the socket, so */etc/clamd.conf* comment line:

```
#TCPSocket 3310
#TCPAddr 127.0.0.1
```

Update the antivirus database automatically connect, the utility responsible for it **freshclam** . We will check that the relevant file was in *cron.daily* and run

anti-virus service

```
freshclam
chkconfig clamd on && /etc/init.d/clamd start
```

Deploy or not to deploy a Web-based interface - a private matter. I thought he needed to monitor the migration process. You may need it to create a database and domain management structures, boxes, aliases, etc. For the latter task in most manuals are actively invited **PostfixAdmin** , but I categorically do not like it. As well as I would recommend to stick to the principle of separation on which mail server should be engaged in mail processing, web server to keep web applications, and DB-server to work with databases.

For those who do not want to deploy Web subsystem, enclose the SQL-dump of the database for the mail server for all occasions. Even with some unused possibilities: mysql_dump.sql on githabe .

If we need to PostfixAdmin - put **nginx / apache + php** and actually he **PostfixAdmin** . And note, that will not be able to deploy it on top of the dump This misleading - the structure removed some of the "extra" table. Nuances settings PostfixAdmin bit. Edit *the config.inc.php* , paying attention to the following parameters:

```
## должен содержать тот же метод хэширования пароля, что и dovecot
$CONF['encrypt'] = 'md5crypt';
$CONF['transport_default'] = 'virtual';
$CONF['emailcheck_resolve_domain']='NO';
## Поскольку будет использоваться dovecot, убираем префикс
$CONF['create_mailbox_subdirs_prefix']='';
## На случай, если захочется использовать хранение квот не в maildir, а в базе данных через dict
$CONF['new_quota_table'] = 'yes';
```

You can then go to domain.tld / postfixadmin / setup.php , generate a password and create an account super-administrator. Now the generated hash must

change the config.inc.php file and change its status:

```
$CONF['configured'] = true;
$CONF['setup_password'] = 't8h9i9s2i7s7m2y4l9o8g9i4n:a0n9d5p2a5s2s9w5o4r7d';
```

[!] Postfixadmin itself creates the database structure and mysql, and postgresql setup.php at startup. If you intend to use it, the installation should be carried out on an empty database.

We will check that the database is created, all postfix default 'structure and move on to installing the MTA and MDA. **Postfix** already comes bundled with CentOS, but not the new one. We update it from **CentALT** and from there put Dovecot.

```
yum update postfix
yum install dovecot dovecot-mysql
```

All major vehicle systems will operate with files in */ var / vmail* under separate juzverej:

```
groupadd  -g 1000 vmail
useradd -d /var/vmail/ -g 1000 -u 1000 vmail
chown vmail:vmail /var/vmail
```

We do self-signed SSL'ki

```
mkdir /etc/postfix/certs
openssl req -new -x509 -days 3650 -nodes -out /etc/postfix/certs/cert.pem -keyout /etc/postfix/certs/key.pem
```

Nastiest stage of assembly - to convince Postfix to work with the database:

```
mkdir /etc/postfix/mysql
```

In this directory, create a file with the following content:

Конфигурация postfix mysql

Edit the file */etc/postfix/main.cf* , teaching Postfix to work with the database of the newly created files:

```
# ==========================================================
# MySQL mappings
# ==========================================================
relay_domains = mysql:/etc/postfix/mysql/relay_domains.cf
virtual_alias_maps = mysql:/etc/postfix/mysql/virtual_alias_maps.cf,
                     mysql:/etc/postfix/mysql/virtual_alias_domain_maps.cf
virtual_mailbox_domains = mysql:/etc/postfix/mysql/virtual_mailbox_domains.cf
virtual_mailbox_maps = mysql:/etc/postfix/mysql/virtual_mailbox_maps.cf
```

Good e-mail server transmits and authorizes their foes. Authentication to work correctly, run the **Submission**, raising the SMTP service in addition to the 587 port. Smartphones when creating new accounts with inscribing smtp server with authentication offer 587 default port; You do not want to explain to customers that is not enough to write mail.domain.tld, and still have the ports and some prescribe. In general, in */etc/postfix/master.cf* edit section responsible for submission:

```
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_wrappermode=no
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_recipient_restrictions=permit_mynetworks,permit_sasl_authenticated,reject
  -o smtpd_relay_restrictions=permit_mynetworks,permit_sasl_authenticated,defer_unauth_destination
  -o milter_macro_daemon_name=ORIGINATING
```

[!] Be sure to pay attention to the gaps before **-o** keys - without configuration will not be valid.

While the master.cf put aside, we'll come back to it later and continue with */etc/postfix/main.cf*

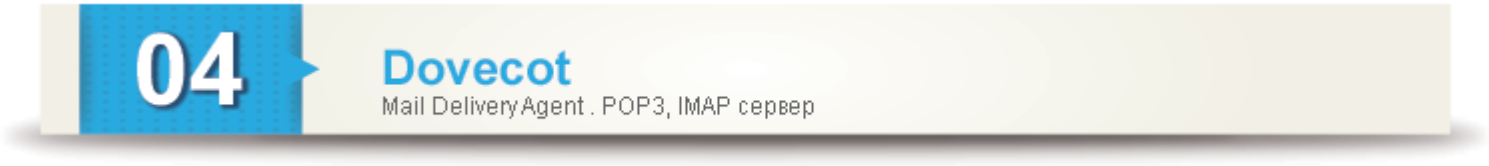
```
soft_bounce = no
myhostname = mail.domain.tld
mydomain = domain.tld
myorigin = $myhostname

# Убираем $myhostname и $mydomain из локального класа
# Чтобы не было do not list domain in BOTH mydestination and virtual_mailbox_domains
mydestination = localhost.$mydomain, localhost

## Вписываем свои сетки
## Обязательно отредактировать и сузить до максимально узкого диапазона
## Лучше вообще не использовать и не позволять отправку почты без авторизации
mynetworks = 192.168.0.0/16, 127.0.0.0/8

## Убираем nis:/ чтобы избежать спама в лог
## dict_nis_init: NIS domain name not set - NIS lookups disabled
alias_maps = hash:/etc/aliases

smtpd_banner = $myhostname ESMTP $mail_name
debug_peer_level = 2
debug_peer_list = 127.0.0.1
```

They were changing the default strings. Now add some sections of our settings. Check for duplicates, removing them from their native configuration, if they meet there. I offer my setup enter structured blocks in the bottom of the file */etc/postfix/main.cf* :

Another configuration sheet

[!] To use or not to use the black list - your choice. I commented out the directives **reject_rbl_client** , not to procreate holivary. Greylisting often enough, and Spamhaus and his ilk profess controversial policy, but in practice it turned out that "honest guys" in the black list just did not skidding and false positives, we did not.Lucky, I guess. So - a matter of taste include RBL directive or not. Consider that I have them for informational purposes.
[!] The parameters are divided into groups - carefully revise and adjust them to suit your needs. No worse option than blindly slapped a foreign configuration without edits.

[!] Malamutrightly observed that the option `permit_mynetworks` extremely doubtful and dangerous. Much better would be to remove it and allow the sending of correspondence only to authenticated users.
[!] To file main.cf we'll be back again, adding a **postgrey** , **amavis** and **dovecot** , and now let's MDA



What is it:

- This is our Mail Delivery Agent, local transport
- Own SASL, which will run through the Postfix
- Work with quotas
- Providing the user with imap, and pop3

Couple of changes in */etc/dovecot/dovecot.conf* :

```
protocols = imap pop3
login_greeting = Hello there.
```

The rest of the configuration file is conveniently broken down into components and well documented:

10-auth.conf

10-logging.conf

Separate listing will not be - at the request include the required options.

10-mail.conf

```
mail_location = maildir:/var/vmail/%d/%n
mail_uid = 1000
mail_gid = 1000
mail_plugins = quota
```

10 master.conf

```
# Поднимаем imap и imaps
service imap-login {
  inet_listener imap {
    port = 143
  }
  inet_listener imaps {
    port = 993
    ssl = yes
  }
}

# Поднимаем pop3 и pop3s
service pop3-login {
  inet_listener pop3 {
    port = 110
  }
  inet_listener pop3s {
    port = 995
    ssl = yes
  }
}

# Поднимаем SASL, куда будет ходить postfix
service auth {
  unix_listener auth-userdb {
    mode = 0600
    user = vmail
    group = vmail
```

```
  }
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user=postfix
    group=postfix
  }
}
```

10-ssl.conf

```
ssl = yes
ssl_cert = </etc/postfix/certs/cert.pem
ssl_key = </etc/postfix/certs/key.pem
```

15-lda.conf

```
quota_full_tempfail = no
lda_mailbox_autocreate = yes
lda_mailbox_autosubscribe = yes
protocol lda {
mail_plugins = $mail_plugins autocreate
}
```

20-imap.conf

```
protocol imap {
mail_plugins = $mail_plugins autocreate quota imap_quota
}
```

90-plugin.conf

```
# Данные для вас могут отличаться.
# Такие папки создавались для безболезненной миграции со старой системы
```

```
plugin {
  autocreate = Trash
  autocreate2 = sent-mail
  autocreate3 = drafts
  autosubscribe = Trash
  autosubscribe2 = sent-mail
  autosubscribe3 = drafts
}
```

90-quota.conf

```
# Отредактируйте значения в соответствии с собственными нуждами.
plugin {
  quota_rule = *:storage=200M
  quota_rule2 = Trash:storage=+10M
}

plugin {
  quota_warning = storage=90%% quota-warning 90 %u
}

# Проверьте, чтобы у скрипта был нужный chmod
# Плюс чтобы в quota-warning.sh был указан корректный путь до dovecot-lda
service quota-warning {
  executable = script /usr/local/bin/quota-warning.sh
  user = vmail
  unix_listener quota-warning {
    user = vmail
  }
}

# Формат хранения квоты - maildir
# При желании можно менять на dict, fs или dirsize
# Подробная документация в http://wiki2.dovecot.org/Quota
plugin {
  quota = maildir:User quota
}
```

auth-sql.conf.ext

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

Создаём */etc/dovecot/dovecot-sql.conf.ext*:

dovecot-sql.conf.ext

```
driver = mysql
connect = host=localhost dbname=postfix user=postfix password=mypassword
default_pass_scheme = MD5-CRYPT
user_query = SELECT '/var/vmail/%d/%n' as home, 'maildir:/var/vmail/%d/%n'as mail, 1000 AS uid, 1000 AS gid, concat('*:bytes=', quota) AS quota_rule
FROM mailbox WHERE username = '%u' AND active = '1'
password_query = SELECT username as user, password, '/var/vmail/%d/%n' as userdb_home, 'maildir:/var/vmail/%d/%n' as userdb_mail, 1000 as
userdb_uid, 1000 as userdb_gid, concat('*:bytes=', quota) AS userdb_quota_rule FROM mailbox WHER
E username = '%u' AND active = '1'
```

Now, make friends **Postfix** with **Dovecot** . Add two sections to */etc/postfix/main.cf* :

```
# =============================================================
# SASL
# =============================================================
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
```

```
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth

# ==========================================================
# VIRTUAL
# ==========================================================
virtual_mailbox_base = /var/vmail
virtual_minimum_uid = 1000
virtual_uid_maps = static:1000
virtual_gid_maps = static:1000
virtual_transport = dovecot
dovecot_destination_recipient_limit = 1
```

And put **Postfix** face the fact that mail delivery has been dovecot. In */etc/postfix/master.cf* :

```
# ==========================================================================
# DOVECOT
# ==========================================================================
dovecot unix    -       n       n       -       -      pipe
  flags=DRhu user=vmail:vmail argv=/usr/libexec/dovecot/deliver -f ${sender} -d ${recipient}
```

Next, you must check that warning about exceeding the quota script */usr/local/bin/quota-warning.sh* practiced correctly. In my case CentOS way to it was invalid and had to edit it manually. It anyway, by the way, to fix by manually editing the address of the sender, which by default is specified as *postmaster@domain.tld* . Find the required binary

```
updatedb
locate dovecot-lda
chmod 755 /usr/local/bin/quota-warning.sh
```

And adjusted the way in the */usr/local/bin/quota-warning.sh* , adding to the script more descriptive headers if desired.

05 Spamassassin + Amavisd-new

Because we will work **Amavis** , which is the layer between the post and antiviral agents, anti-spam systems that run spamd does not need to separately - it works as a module loaded when needed. SA to hold renewed, used native utility **sa-the update** . Verify that the *etc / cron.d* have a file **sa-update** with the planned launch of the updater.

[!] Put **spamassassin 3.3.2** of **rpmforge-extras**Because pierces EPEL of version 3.3.1 has a congenital defect in the sa-update. The latest version **3.3.2** has spared this problem and is updated correctly

```
yum install spamassassin amavisd-new
```

Bit will correct */etc/mail/spamassassin/local.cf*

local.cf

But the configuration file **Amavis** ' and - the exact opposite touted configs Dovecot me. He is a regular perl script, which is also more poorly formatted. I make a change - rename files in .pl and edited with syntax highlighting; it's a little easier life.

amavisd.conf

Now let **the Postfix** 'in that he now uses **Amavis** for checking mail. Make a corresponding block in */etc/postfix/master.cf*

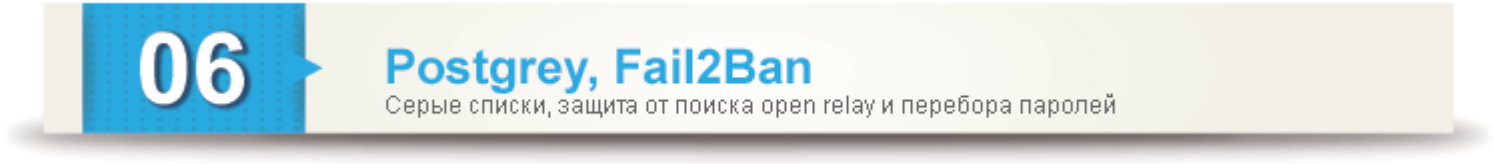
master.cf

and */etc/postfix/main.cf* add

```
# ==========================================================
# AMAVIS
# ==========================================================
content_filter=amavisfeed:[127.0.0.1]:10024
```

We are working on the following services:

```
chkconfig amavisd on && /etc/init.d/amavisd start
/etc/init.d/postfix restart

telnet 127.0.0.1 10024
# 220 [127.0.0.1] ESMTP amavisd-new service ready
```



On the effectiveness of Greylisting it has been written many times, so I just silently

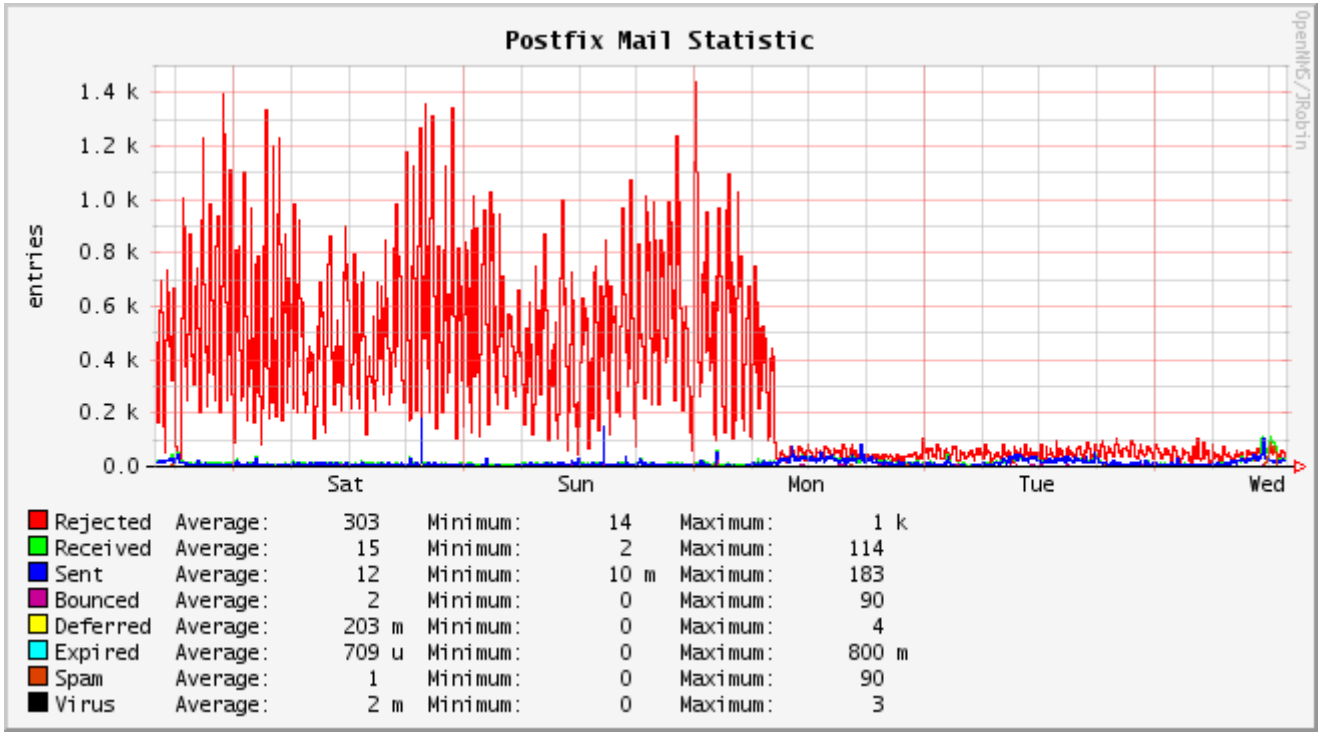
```
yum install postgrey
```

No further configuration is required - to prescribe it in */etc/postfix/main.cf*

```
smtpd_recipient_restrictions = ...
                               reject_unauth_destination,
                               check_policy_service unix:/var/spool/postfix/postgrey/socket,
                               ...
```

[!] Check_policy_service Directive should *definitely* go after reject_unauth_destination.
If you need to be excluded from scanning any server - edit */etc/postfix/postgrey_whitelist_clients.local* , and to exclude from the scan specific e-mail addresses on the local server - edit *postgrey_whitelist_recipients* . Comprehensive information is available on the wiki: wiki.centos.org/HowTos/postgrey

Pro **Fail2ban**separate conversation. To demonstrate its effectiveness, show the picture to the statistics of the mail server before and after you install the utility. The red line on the graph - unauthorized swotting in search of an open relay. Letters, of course, cut off, and the load is no longer created, but why bother to listen to this garbage. Therefore fail2ban install three rules significantly improve the appearance of the graph:




```
yum install fail2ban
```

Rules for the mail server /etc/fail2ban/jail.conf

If SSH is open only on the internal network, the first rule and can be removed. And yet, no additional movements - rules work "out of the box." Thank urbain for the reminder about protection against brute force smtp.

```
chkconfig fail2ban on && /etc/init.d/fail2ban start
```

07 Тестирование

Since we hooked Dovecot plugin *autocreate* , is to create domains and mailboxes enough to lead them to the database or through Postfixadmin making INSERT INTO console. When the first authentication or the first letter received directory structure will be created automatically.

Potestiruem pop3, imap, smtp in different poses

Potestiruem virus protection

08 Заключение

Now ~~the computer's power can be turned off~~ , you can start to work with your mail server, create domains, users, aliases, etc. Finally some general points and recommendations:

1. To work with the configuration files I use git, turning the / etc directory to the repository. This approach allows you to conveniently fumble configs between the employees of the technical department and the steps to control the configuration process. In dealing with such a complex system as a mail server, such a practice can greatly simplify life.
2. One of my tasks was to migration c courier to dovecot. And yes, it's real. Official documentation wiki2.dovecot.org/Migration/Courier provides nearly all the required information. In addition, it will recommend manually POP3 UIDL command on the old and the new server under the same user with a non-empty mailbox, comparing the results. They should be the same, otherwise the user's mail client will download all the mail from the box again.
   It is also possible for the end of the migration to the new server, delete unnecessary files and folders

```
3. find . -name "courier*" -delete
```

4. Do not forget about iptables - open only the required ports and hide from prying official eyes.
5. Bundled with CentOS logger rsyslog I never liked him, and I change it to the syslog-ng from EPEL. By the way, the direct participation of syslog-ng schedule is drawn from the sixth chapter - collecting and recording statistics is conducted by parsing the logs.
6. Table created vacation, but do not put dovecot Sieve plugin and "auto-reply" no. Therefore the table - just a reserve "for the future." Hyde and so giant turned.

Almost complete assembly configuration files I put on githab , so if necessary you can access the configuration file is not in pieces, and whole. In this article, files edited gradually, mimicking the same gradual adjustment of all systems. The vast majority of files were given guidelines and once the reader has not turned an idea that goes behind what and how to communicate. I tried to get rid of it, showing phasing settings. Perhaps someone so handy.

Once again, I stress that *this is the initial stage of installation*. Even if you do not consider monitoring the connection, which is highly desirable on the mail server, working more no end. Definitely need careful adjustment of anti-spam policy; If you plan to use additional relay servers, you will need revision requests; sure to be a need to adjust the parameters of restrictions, etc.

But otherwise this attempt at writing in the writing of large-scale management can be considered completed. The plans include the publication of the material is no longer for "beginners" and not on the e-mail server, but I decided to practice on cats. Who has read this far ... um ... I envy your patience, but split the publication into multiple parts it seemed inappropriate.

How To Install Mail Server With PostfixAdmin on CentOS 7

http://idroot.net/linux/install-mail-server-postfixadmin-centos-7/

In this tutorial we will show you how to install and configuration of Mail Server With PostfixAdmin on CentOS 7 server. For those of you who didn't know, Postfixadmin is a web frontend to the mysql or mariaDB database used by Postfix. With Postfixadmin we can easily manage from a web browser our Postfix service, adding and removing mail users and domains, mail aliases, disk quotas, etc.

This article assumes you have at least basic knowledge of linux, know how to use the shell, and most importantly, you host your site on your own VPS. The installation is quite simple and assumes you are running in the root account, if not you may need to add 'sudo' to the commands to get root privileges. I will show you through the step by step installation Seafile Secure Cloud Storage on a CentOS 7 server.

Install Mail Server With PostfixAdmin on CentOS 7

Step 1. First, let's start by ensuring your system is up-to-date.

```
1 yum clean all
2 yum -y update
```

Step 2. Install LAMP server.

A CentOS 7 LAMP stack server is required. If you do not have a LAMP installed, you can follow our guide here. Also install required PHP modules:

```
1 yum install php70w-cli php70w-gd php70w-xml php70w-curl php70w-mysql php70w-zip php70w-mbstring php70w-mcrypt
```

Step 3. Configuring MariaDB for PostfixAdmin.

By default, MariaDB is not hardened. You can secure MariaDB using the mysql_secure_installation script. You should read and below each step carefully which will set the root password, remove anonymous users, disallow remote root login, and remove the test database and access to secure MariaDB:

```
1 mysql_secure_installation
```

Configure it like this:

```
1 - Set root password? [Y/n] y
2 - Remove anonymous users? [Y/n] y
3 - Disallow root login remotely? [Y/n] y
4 - Remove test database and access to it? [Y/n] y
5 - Reload privilege tables now? [Y/n] y
```

Next we will need to log in to the MariaDB console and create a database for the PostfixAdmin. Run the following command:

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

This will prompt you for a password, so enter your MariaDB root password and hit Enter. Once you are logged in to your database server you need to create a database for PostfixAdmin installation:

```
1 CMariaDB [(none)]> CREATE DATABASE postfixadmin;
2 MariaDB [(none)]> GRANT ALL PRIVILEGES ON postfixadmin.* TO 'postfixadmin'@'localhost' IDENTIFIED BY 'strong_password';
3 MariaDB [(none)]> FLUSH PRIVILEGES;
4 MariaDB [(none)]> \q
```

Step 4. Installing PostfixAdmin.

The first thing to do is to go to PostfixAdmin's download page and download the latest stable version of PostfixAdmin:

```
1 wget -q -O - "https://downloads.sourceforge.net/project/postfixadmin/postfixadmin/postfixadmin-3.0.2/postfixadmin-3.0.2.tar.gz" | tar
  -xzf - -C /var/www/html
```

Open the mail configuration file:

```
1 nano /var/www/html/postfixadmin-3.0.2/config.inc.php
```

Edit the following values:

```
1 $CONF['configured'] = true;
2 $CONF['database_type'] = 'mysqli';
3 $CONF['database_host'] = 'localhost';
4 $CONF['database_user'] = 'postfixadmin';
5 $CONF['database_password'] = 'strong_password';
6 $CONF['database_name'] = 'postfixadmin';
7 $CONF['domain_path'] = 'NO';
8 $CONF['domain_in_mailbox'] = 'YES';
```

Now you need to assign the ownership of the files and folders to Apache's user and group. To do so, the command is:

```
1 chown -R apache: /var/www/html/postfixadmin-3.0.2
```

To populate the database go to https://Your_IP_Address/postfixadmin-3.0.2/setup.php and you should see something like below:

```
1 Testing database connection - OK - mysqli://postfixadmin:xxxxx@localhost/postfixadmin
2 Everything seems fine... attempting to create/update database structure
```

Create a new admin user:

```
1 bash /var/www/html/postfixadmin-3.0.2/scripts/postfixadmin-cli admin add admin@your_domain_idroot.net --password
  strong_password22 --password2 strong_password22 --superadmin 1 --active 1
```

Step 5. Installing and configure Postfix.

Install postfix with the following command:

```
1 yum -y install postfix
```

Once the installation is completed, we need to create configuration files:

```
1 mkdir -p /etc/postfix/sql/
```

```
1 nano /etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query  = SELECT goto FROM alias,alias_domain WHERE alias_domain.alias_domain = '%d' and alias.address = CONCAT('@',
  alias_domain.target_domain) AND alias.active = 1 AND alias_domain.active='1'
```

```
1 nano /etc/postfix/sql/mysql_virtual_alias_domain_mailbox_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
```

```
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query = SELECT maildir FROM mailbox,alias_domain WHERE alias_domain.alias_domain = '%d' and mailbox.username =
  CONCAT('%u', '@', alias_domain.target_domain) AND mailbox.active = 1 AND alias_domain.active='1'
```

```
1 nano /etc/postfix/sql/mysql_virtual_alias_domain_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query = SELECT goto FROM alias,alias_domain WHERE alias_domain.alias_domain = '%d' and alias.address = CONCAT('%u',
  '@', alias_domain.target_domain) AND alias.active = 1 AND alias_domain.active='1'
```

```
1 nano /etc/postfix/sql/mysql_virtual_alias_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query = SELECT goto FROM alias WHERE address='%s' AND active = '1'
6 #expansion_limit = 100
```

```
1 nano /etc/postfix/sql/mysql_virtual_domains_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query          = SELECT domain FROM domain WHERE domain='%s' AND active = '1'
6 #query         = SELECT domain FROM domain WHERE domain='%s'
7 #optional query to use when relaying for backup MX
```

```
8 #query           = SELECT domain FROM domain WHERE domain='%s' AND backupmx = '0' AND active = '1'
9 #expansion_limit = 100
```

```
1 nano /etc/postfix/sql/mysql_virtual_mailbox_limit_maps.cf
```

Edit the following values:

```
1 user = postfixadmin
2 password = strong_password
3 hosts = localhost
4 dbname = postfixadmin
5 query = SELECT quota FROM mailbox WHERE username='%s' AND active = '1'
```

```
1 nano /etc/postfix/sql/mysql_virtual_mailbox_maps.cf
```

Edit the following values:

```
user = postfixadmin
password = strong_password
hosts = localhost
dbname = postfixadmin
query          = SELECT maildir FROM mailbox WHERE username='%s' AND active = '1'
#expansion_limit = 100
```

Next, edit the main.cf file:

```
postconf -e "myhostname = $(hostname -f)"

postconf -e "virtual_mailbox_domains = proxy:mysql:/etc/postfix/sql/mysql_virtual_domains_maps.cf"
postconf -e "virtual_alias_maps = proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_maps.cf,
proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_maps.cf,
proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf"
postconf -e "virtual_mailbox_maps = proxy:mysql:/etc/postfix/sql/mysql_virtual_mailbox_maps.cf,
proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_mailbox_maps.cf"
postconf -e "smtpd_tls_cert_file = /etc/pki/tls/certs/localhost.crt"
postconf -e "smtpd_tls_key_file = /etc/pki/tls/private/localhost.key"
postconf -e "smtpd_use_tls = yes"
postconf -e "smtpd_tls_auth_only = yes"
postconf -e "smtpd_sasl_type = dovecot"
postconf -e "smtpd_sasl_path = private/auth"
postconf -e "smtpd_sasl_auth_enable = yes"
postconf -e "smtpd_recipient_restrictions = permit_sasl_authenticated, permit_mynetworks, reject_unauth_destination"
```

```
postconf -e "mydestination = localhost"
postconf -e "mynetworks = 127.0.0.0/8"
postconf -e "inet_protocols = ipv4"
postconf -e "inet_interfaces = all"
postconf -e "virtual_transport = lmtp:unix:private/dovecot-lmtp"
```

Open the master.cf file, find submission inet n and smtps inet n sections and edit as follows:

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

Edit the following values:

```
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
# -o smtpd_reject_unlisted_recipient=no
# -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
```

```
11    -o milter_macro_daemon_name=ORIGINATING
12 smtps     inet  n       -       n       -       -       smtpd
13    -o syslog_name=postfix/smtps
14 #  -o smtpd_tls_wrappermode=yes
15    -o smtpd_sasl_auth_enable=yes
16 #  -o smtpd_reject_unlisted_recipient=no
17 #  -o smtpd_client_restrictions=$mua_client_restrictions
18 #  -o smtpd_helo_restrictions=$mua_helo_restrictions
19 #  -o smtpd_sender_restrictions=$mua_sender_restrictions
20 #  -o smtpd_recipient_restrictions=
21    -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
22    -o milter_macro_daemon_name=ORIGINATING
```

Finally, enable the postfix service:

```
1  systemctl enable postfix
2  systemctl restart postfix
```

Step 6. Installing and Configure Dovecot.

Install dovecot using the command bellow:

```
1  yum install dovecot dovecot-mysql
```

Open the /etc/dovecot/conf.d/10-mail.conf file:

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

Change the following values:

```
1 mail_location = maildir:/var/vmail/%d/%n
2 mail_privileged_group = mail
3 mail_uid = vmail
4 mail_gid = mail
5 first_valid_uid = 150
6 last_valid_uid = 150
```

Open the /etc/dovecot/conf.d/10-auth.conf file:

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

Change the following values:

```
1 auth_mechanisms = plain login
2 #!include auth-system.conf.ext
3 !include auth-sql.conf.ext
```

Create a new dovecot-sql.conf.ext file:

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

Edit the following values:

```
1 driver = mysql
2 connect = host=localhost dbname=postfixadmin user=postfixadmin password=strong_password
3 default_pass_scheme = MD5-CRYPT
4 password_query = SELECT username as user, password, '/var/vmail/%d/%n' as userdb_home, 'maildir:/var/vmail/%d/%n' as
  userdb_mail, 150 as userdb_uid, 8 as userdb_gid FROM mailbox WHERE username = '%u' AND active = '1'
5 user_query = SELECT '/var/vmail/%d/%u' as home, 'maildir:/var/vmail/%d/%u' as mail, 150 AS uid, 8 AS gid,
  concat('dirsize:storage=',  quota) AS quota FROM mailbox WHERE username = '%u' AND active = '1'
```

In the /etc/dovecot/conf.d/10-ssl.conf file enable SSL support:

```
1 ssl = yes
```

Open the /etc/dovecot/conf.d/15-lda.conf file and set the postmaster_address email address:

```
1 postmaster_address = postmaster@your_domain_name.com
```

Open the /etc/dovecot/conf.d/10-master.conf file, find the service lmtp section and change it to:

```
1 service lmtp {
2   unix_listener /var/spool/postfix/private/dovecot-lmtp {
3     mode = 0600
4     user = postfix
5     group = postfix
6   }
7 }
```

Find the service auth section and change it to:

```
service auth {
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user = postfix
    group = postfix
  }
  unix_listener auth-userdb {
    mode = 0600
    user = vmail
    #group = vmail
  }
  user = dovecot
}
```

Change the service auth-worker section to the following:

```
service auth-worker {
  user = vmail
}
```

Now you need to assign the ownership of the files and folders. To do so, the command is:

```
1 chown -R vmail:dovecot /etc/dovecot
2 chmod -R o-rwx /etc/dovecot
```

Finally, enable and restart the dovecot service:

```
1 systemctl enable dovecot
2 systemctl restart dovecot
```

Step 7. Installing and configure Spamassassin.

Install spamassassin using the command bellow:

```
1 yum -y install spamassassin
```

Create a spamassassin system user:

```
1 groupadd spamd
```

```
2 useradd -g spamd -s /bin/false -d /var/log/spamassassin spamd
3 chown spamd:spamd /var/log/spamassassin
```

Next, configure Postfix to use SpamAssassin:

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

Change values:

```
1 smtp      inet  n       -       n       -       -       smtpd
```

With:

```
1 smtp      inet  n       -       n       -       -       smtpd -o content_filter=spamassassin
```

Add the following line at the end of the file:

```
1 systemctl enable spamassassin
```

```
2 systemctl restart spamassassin
```

Finally, restart the postfix service:

```
1 systemctl restart postfix
```

Step 8. Accessing PostfixAdmin.

If everything is set up correctly now you should be able to log in to your PostfixAdmin backend by going to http://Your_IP_Address/postfixadmin-3.0.2.2 and create your first virtual domain and mailbox.

Congratulation's! You have successfully installed PostfixAdmin. Thanks for using this tutorial for installing Mail Server With PostfixAdmin on CentOS 7 system. For additional help or useful information, we recommend you to check the official PostfixAdmin web site.

Configure mail server on Centos 7 with Postfix, Dovecot, Apache, postfixadmin and Roundcube.

http://blog.ceae.info/configure-mail-server-on-centos-7-with-postfix-dovecot-apache-postfixadmin-and-roundcube/

We starting from Centos 7 Infrastructure Server with Mail Server from Installer.
We have now:

1 – We have public IP
2 – Revers DNS for this IP
3 – Domain already bought.

Step 0. Preparing with minimal aplication to install:

```
yum -y install wget whois nc vim gpm ppp rp-pppoe dialog logwatch telnet nmap mutt
yum -y install epel-release
yum -y update
yum -y install perl-MailTools perl-MIME-EncWords perl-Email-Valid perl-Test-Pod dovecot dovecot-mysql  dovecot-pigeonhole
perl-Mail-Sender perl-Log-Log4perl imapsync offlineimap amavisd-new clamav perl-Razor-Agent mariadb-server opendkim vim wget
crypto-utils mod_ssl.x86_64 php php-mysql php-fpm  clamav-update php-imap.x86_64 NetworkManager-tui mailx lrzip lzop lz4 arj  unzoo
cabextract p7zip fail2ban php-mcrypt.x86_64
systemctl stop rpcbind
systemctl disable rpcbind
```

Step 1. Setup your hostname server.

```
hostnamectl set-hostname mail.your-domain.tk
```

Edit you /etc/hosts to look like this

```
[root@mail ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
```

```
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.1.47 mail.your-domain.com mail
```

Step 2. Test postfix local delivery
We create 2 users for local delivery test.

```
useradd -d /home/john -M -N -s /sbin/nologin john
useradd -d /home/mark -M -N -s /sbin/nologin mark
```

Now we will send a local mail.

```
echo Hello | mail -s test john@localhost
```

and will check if mail has been delivered

```
tail -f /var/log/maillog
Oct 21 14:55:58 localhost postfix/local[2916]: 770201440486: to=<john@localhost.your-domain.com>, orig_to=<john@localhost>,
relay=local, delay=0.19, delays=0.13/0.02/0/0.04, dsn=2.0.0, status=sent (delivered to mailbox)
```

if works go to next step

Step 3. Setup MariaDB.
Config next `/etc/my.cnf.d/server.cnf` file like this:

```
# this is read by the standalone daemon and embedded servers
[server]
```

innodb_file_per_table
innodb_file_format = Barracuda

```
# this is only for the mysqld standalone daemon
[mysqld]
```

Enable mariadb service

```
systemctl enable mariadb.service
```

Start mariadb database server

```
systemctl start mariadb.service
```

Secure mariadb installation

```
mysql_secure_installation
```

Step 4. Configure Clam Antivirus

We need to configure how clam refreshes his database

```
vim /etc/sysconfig/freshclam
```

comment or remove last line

```
#  FRESHCLAM_DELAY=disabled-warn    # REMOVE ME
```

we will make a edit clamav config file

```
vim /etc/freshclam.conf
```

comment or remove line with words “example”.

and finally update your viruses database.

```
freshclam
```

Step 5. Configure basic settings in spamassasin

Enable spamassasin service

```
systemctl start spamassassin.service
systemctl status spamassassin.service
systemctl enable spamassassin.service
```

update spamassasin definitions

```
sa-update
```

Step 6. Integrate spamassasin and clamav with amavisd.

First install some app:

```
yum -y install clamav clamav-devel clamav-server clamd
```

We need to provide some config files.

```
cp /usr/share/doc/clamav-server-0.99.2/clamd.sysconfig /etc/sysconfig/clamd.amavisd
```

We need to adapt config file to our actual configuration.

```
vim /etc/sysconfig/clamd.amavisd
```

and add to last line

```
CLAMD_CONFIGFILE=/etc/clamd.d/amavisd.conf
CLAMD_SOCKET=/var/run/clamd.amavisd/clamd.sock
```

We will create a couple of new files

```
vim /etc/tmpfiles.d/clamd.amavisd.conf
```

add this content

```
d /var/run/clamd.amavisd 0755 amavis amavis -
```

Edit next file

```
vim /usr/lib/systemd/system/clamd@.service
```

with this content

```
[Unit]
Description = clamd scanner (%i) daemon
After = syslog.target nss-lookup.target network.target
[Service]
Type = simple
ExecStart = /usr/sbin/clamd -c /etc/clamd.d/%i.conf --foreground=yes
Restart = on-failure
PrivateTmp = true
[Install]
WantedBy=multi-user.target
```

Now we can enable clamd@amavisd service

```
systemctl start clamd@amavisd
systemctl enable clamd@amavisd
systemctl status clamd@amavisd
```

Configure amavisd service

```
vim /etc/amavisd/amavisd.conf
```

At line 16 set number of amavisd childrens.

More childres uses more ram but delivers more mail at once, one amavisd children consumes near 30% of cpu in a low end server, be careful if you receive a lot of mails at once can be a big punch in your cpu have too many childrens.

```
$max_servers = <number>
```

line 20 set $mydomain

```
$mydomain = 'ceae.info';
```

line 152 aprox set your hostname

```
$myhostname= 'mail.ceae.info';
```

Start service Amavisd

```
systemctl start amavisd.service
```

Enable service

```
systemctl enable amavisd.service
```

Step 7. Enable Apache and minim config.

yum -y install php php-devel php-gd php-imap php-ldap php-mysql php-odbc php-pear php-xml php-xmlrpc php-mbstring php-mcrypt php-mssql php-snmp php-soap php-tidy curl curl-devel perl-libwww-perl ImageMagick libxml2 libxml2-devel mod_fcgid php-cli httpd-devel spamassassin unzip bzip2 unrar perl-DBD-mysql

systemctl start httpd.service
systemctl enable httpd.service
Edit config file
vim /etc/httpd/conf/httpd.conf
at line 86 edit with your admin email
 # ServerAdmin root@localhost

at line 152 should be
AllowOverride None
AllowOverride All

Config php
vim /etc/php.ini
at line 763 edit like this
;cgi.fix_pathinfo=1
cgi.fix_pathinfo=0

at line 877 edit like this
;date.timezone =
date.timezone = Europe/Berlin

And now restart apache
systemctl restart httpd.service
Step 8. Setup Postfixadmin
Download Postxiadmin
Page is *https://sourceforge.net/projects/postfixadmin/files/postfixadmin/* and download the latest.

```
wget
http://downloads.sourceforge.net/project/postfixadmin/postfixadmin/postfixadmin-3.0/postfixadmin-3.0.tar.gz?r=https%3A%2F%2Fsour
ceforge.net%2Fprojects%2Fpostfixadmin%2Ffiles%2Fpostfixadmin%2Fpostfixadmin-3.0%2F&ts=1479731076&use_mirror=netix
```

Move file to tar.gz file

```
mv
postfixadmin-3.0.tar.gz\?r\=https\:%2F%2Fsourceforge.net%2Fprojects%2Fpostfixadmin%2Ffiles%2Fpostfixadmin%2Fpostfixadmin-3.0%2F
postfixadmin-3.0.tar.gz
```

Extract folder

```
tar -zxvf postfixadmin-3.0.tar.gz -C /var/www/html/
cd /var/www/html/
chown -R root.apache postfixadmin-3.0/
 ln -s postfixadmin-3.0/ postfixadmin
```

We need to create postfix user database:

```
mysql -u root -p
MariaDB [(none)]> create database postfix;
Query OK, 1 row affected (0.00 sec)
MariaDB [(none)]> GRANT all on postfix.* to 'postfix'@'localhost' identified by 'yourPASSword';
Query OK, 0 rows affected (0.00 sec)
```

Now we have database, edit config file:
vim /var/www/html/postfixadmin/config.inc.php

```
$CONF['configured'] = true;
$CONF['setup_password'] = 'YOUR-STRONG-PASSWORD';
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'postfix';
$CONF['database_password'] = 'yourPASSword';
$CONF['database_name'] = 'postfix';
$CONF['show_password'] = 'YES';
$CONF['page_size'] = '30';
$CONF['default_aliases'] = array (
'abuse' => 'abuse@ceae.info',
'hostmaster' => 'hostmaster@ceae.info',
'postmaster' => 'postmaster@ceae.info',
'webmaster' => 'webmaster@ceae.info'
);
$CONF['domain_path'] = 'NO';
$CONF['domain_in_mailbox'] = 'YES';
$CONF['maildir_name_hook'] = 'NO';
$CONF['transport'] = 'YES';
$CONF['vacation'] = 'YES';
$CONF['vacation_domain'] = 'autoreply.ceae.info';
$CONF['vacation_control'] ='YES';
```

If your domain do not exist, activate this

```
$CONF['emailcheck_resolve_domain']='NO';
```

Now got to browser and type *http://your-ip/postfixadmin/setup.php* and setup your admin password.

Now login and create new domain and email *http://YOUR-IP-server/postfixadmin/login.php*

Step 10 Setup Dovecot.

Now we enable IMAP and POP3 service.

vim /etc/dovecot/dovecot-sql.conf.ext

```
# The mysqld.sock socket may be in different locations in different systems
driver = mysql
##
connect = host=localhost dbname=postfix user=postfix password=yourpassword
#
# Default password scheme.
# depends on your $CONF['encrypt'] setting:
# md5crypt  -> MD5-CRYPT
# md5       -> PLAIN-MD5
# cleartext -> PLAIN
default_pass_scheme = MD5-CRYPT
# Query to retrieve password. user can be used to retrieve username in other
# # formats also.
password_query = SELECT username AS user,password FROM mailbox WHERE username = '%u' AND active='1'
# Query to retrieve user information.
## user_query = SELECT maildir, 1001 AS uid, 1001 AS gid FROM mailbox WHERE username = '%u' AND active='1'
user_query = SELECT CONCAT('/var/spool/vmail/', domain,'/', maildir) AS home, CONCAT('maildir:/var/spool/vmail/',domain,'/',
maildir) AS mail, 5000 AS uid, 12 AS gid, concat('dict:storage=',CAST(ROUND(quota / 1024) AS CHAR), '::proxy::quota') AS quota,
CONCAT('*:storage=',CAST(quota AS CHAR), 'B') AS quota_rule FROM mailbox WHERE username = '%u' AND active = '1'
```

Now we edit next file:

vim /etc/dovecot/conf.d/dovecot-mysql-quota.conf.ext

```
connect = host=localhost dbname=postfix user=postfix password=yourpassword
map {
  pattern = priv/quota/storage
  table = quota2
  username_field = username
  value_field = bytes
}
map {
  pattern = priv/quota/messages
  table = quota2
  username_field = username
  value_field = messages
}
```

vim /etc/dovecot/dovecot.conf
line 24 tells dovecot what protocols should serve
protocols = imap pop3

line 31 is what interfaces where dovecot will be listening
*listen = *, ::*

line 44 welcome message,
#login_greeting = Dovecot ready.
login_greeting = Server OK.

line 69 defines behavior when reboot dovecot service
shutdown_clients = yes

Edit custom logging
vim /etc/dovecot/conf.d/10-logging.conf
line 8 log file
log_path = /var/log/dovecot.log

line 32 logging verbose password for debuging
#auth_verbose_passwords = no
auth_verbose_passwords = plain

line 41 enable debug password
#auth_debug_passwords = no
auth_debug_passwords = yes

Restart dovecot
systemctl restart dovecot.service

Create user for delivery internal and log.
```
useradd -r -u 5000 -g mail -d /var/spool/vmail -s /sbin/nologin -c "Virtual mailbox" vmail
```

Create folder vmail
mkdir /var/spool/vmail

change owner of log file
chown vmail /var/log/dovecot.log

Create logrotate for dovecot
vim /etc/logrotate.d/dovecot
/var/log/dovecot.log {
missingok
notifempty
delaycompress
sharedscripts
postrotate
/bin/kill -USR1 `cat /var/run/dovecot/master.pid 2>/dev/null` 2> /dev/null || true
endscript
}

Config authenticated user
vim /etc/dovecot/conf.d/10-auth.conf
line 10 disable plain test on
disable_plaintext_auth = yes

Auth mecanism
auth_mechanisms = plain login cram-md5

and databate type user setup
#!include auth-system.conf.ext
!include auth-sql.conf.ext

Setup SSL
SSL protocols
ssl_protocols = !SSLv2 !SSLv3

SSL ciphers to use
ssl_cipher_list =
EECDH+ECDSA+AESGCM:EECDH+aRSA+AESGCM:EECDH+ECDSA+SHA384:EECDH+ECDSA+SHA256:EECDH+aRSA+SHA384:EECDH+aRSA+SHA256:EECDH+aRSA+RC4:EECDH:EDH+aRSA:!aNULL:!eNULL:!LOW:!3DES:!MD5:!EXP:!PSK:!SRP:!DSS:!RC4

Prefer the server's order of ciphers over client's.
ssl_prefer_server_ciphers = yes

Now dovecot needs to know what protocols will serve and how
vim /etc/dovecot/conf.d/10-master.conf

```
service imap-login {
  inet_listener imap {
    port = 143
  }
  inet_listener imaps {
    port = 993
    ssl = yes
  }
service pop3-login {
  inet_listener pop3 {
    port = 110
  }
  inet_listener pop3s {
    port = 995
    ssl = yes
  }
}

service auth {
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user = vmail
    group = mail
  }
```

Enable sieve in dovecot
vim /etc/dovecot/conf.d/15-lda.conf

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

end for today

----------XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXx----------

Postfix mail server on CentOS 7 with virtual domains, management system, web access and many others

https://www.dmosk.ru/instruktions.php?object=postfix-centos

Terms on the topic: Postfix , POP3 , SMTP , IMAP , MariaDB , CentOS , PostfixAdmin , Dovecot , Roundcube

In this manual, you configure a full-fledged mail server. List of all features and capabilities:

- Mail system based on Postfix;
- Support for virtual domains;
- Storing mail on the server;
- Connection to mailboxes by POP3 and IMAP (Dovecot);
- Encryption support;
- Storing a part of the settings in MariaDB;
- Protection from SPAM and viruses;
- Access to mail using the web interface (Roundcube);
- The ability to manage mailboxes using PostfixAdmin.

Content

1. Presetting the system

I remind you, this instruction is written under the Linux system CentOS version 7.

Setting the correct name to the server is an important step, since most antispam systems perform checks, referring to the server by name in anticipation of a response.

```
relay.dmosk.ru
```

** You must specify the FQDN name that will be available from the WAN. In this example,* ***relay.dmosk.ru is*** *specified .*

After we enter the following command:

Install service packs (they will be needed during server configuration):

* ***ntp*** *for the ability to synchronize the time on the server;* ***wget*** *- client for downloading files.*

Set the time zone (in this example, Moscow time):

Synchronize time:

We update the system:

We open the ports in advance on the firewall :

** where we will open the following ports:*

- ***25*** *- standard SMTP via STARTTLS;*

- ***80*** *- HTTP for Postfixadmin and Roundcube portals;*
- ***110*** *- standard POP3 via STARTTLS;*
- ***143*** *- standard IMAP via STARTTLS;*
- ***443*** *- secure HTTPS for Postfixadmin and Roundcube portals;*
- ***465*** *- Secure SMTP over SSL / TLS;*
- ***587*** *- Secure SMTP through STARTTLS;*
- ***993*** *- Secure IMAP over SSL / TLS;*
- ***995*** *- Secure POP3 over SSL / TLS.*

2. Configuring the web server: NGINX + PHP + MariaDB

The PostfixAdmin control system operates as a web application developed in PHP , and stores information in the database. In our example, we will use the web server on NGINX , and the database will be MariaDB.

NGINX installation

Add the repository with the required package:

```
[nginx]
name = nginx repo
baseurl = http: //nginx.org/packages/centos/$releasever/$basearch/
gpgcheck = 0
enabled = 1
```

Install nginx:

We allow autostart service and start it:

We check the working capacity of the web server by accessing it in the browser using the IP address . If you see the title "Welcome to nginx!", NGINX is configured correctly.

PHP + PHP-FPM + NGINX

Install php and php-fpm :

Configuring NGINX:

```
server {
    listen 80 default_server;
    set $ root_path / usr / share / nginx / html;

    location / {
```

```
        root $ root_path;
        index index.php index.hml;
    }

    location ~ \ .php $ {
        fastcgi_pass unix: /var/run/php-fpm/php5-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $ root_path $ fastcgi_script_name;
        include fastcgi_params;
        fastcgi_param DOCUMENT_ROOT $ root_path;
    }
}
```

** where **/ usr / share / nginx / html** - directory for posting the Postfix management portal.*

Configure PHP-FPM:

```
listen = /var/run/php-fpm/php5-fpm.sock
```

** here we changed the line **127.0.0.1:9000** .*

Run the services:

** If during the restart of nginx the nginx error pops up* ***: [emerg] a duplicate default server*** *, it is necessary to find the virtual domain setting, which also specifies the option* ***default_server*** *- the option should be removed. Or you can configure another virtual domain yourself.*

For verification, create an index file in the site directory with the following content:

```
<? php phpinfo (); ?>
```

Open the site in the browser by its IP address. On the opened page we should see the detailed information on php:

MariaDB

Install the database server with the following command:

Enable the autostart service and run it:

Set the password for the user sql root:

3. Installing and Configuring PostfixAdmin

Install additional components for PHP:

To apply installed packages, restart the script handler:

Download PostfixAdmin:

In the directory of nginx sites we create a directory for postfixadmin and unpack the archive into it:

We set the rights to the directory:

** despite the fact that we use the nginx web server, php-fpm by default, is run from the apache user.*

Create the postfix database and the account in mariadb:

** where **postfix** is the name of the database.*

** where **postfix** is the account name; **postfix123** - password; **localhost** only allows connection from the local server.*

Exit the MariaDB command shell:

Open the configuration file postfixadmin:

And we edit the following:

```
$ CONF ['configured'] = true;
$ CONF ['default_language'] = 'ru';
$ CONF ['database_password'] = 'postfix123';
$ CONF ['emailcheck_resolve_domain'] = 'NO';
```

Launch the browser and enter the address http: // <server IP address> /postfixadmin/setup.php

The process of verifying the configuration and installing the PostfixAdmin portal will begin. After it's finished, enter the password twice and generate a hash:

After reloading the page, we copy the hash:

Open the configuration file:

We find the line:

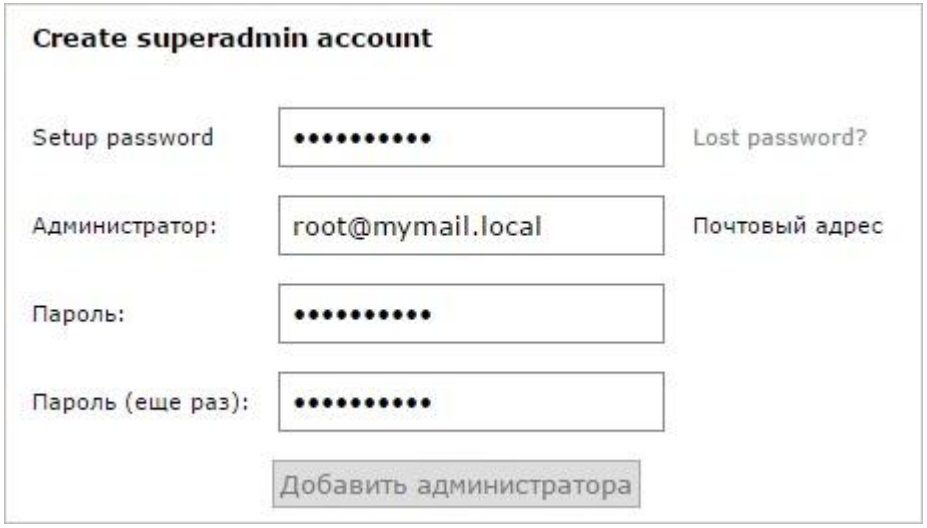
```
$ CONF ['setup_password'] = 'changeme';
```

And we change it to:

```
$ CONF ['setup_password'] = '7a8e14 ... c26';
```

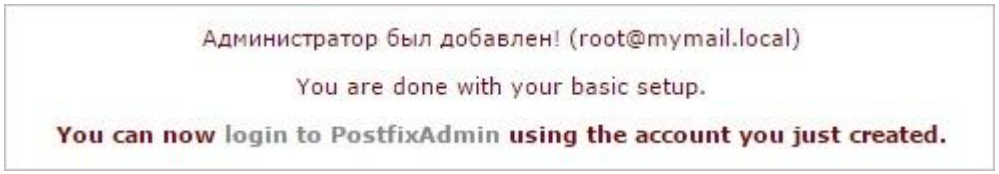
** where **'7a8e14 ... c26'** is the copied hash.*

After, on the same page where the hash is shown, add the superuser PostfixAdmin:



** where **Setup password** is the password that we entered on the previous page; **Password** - the new password for the account you are creating.*

As a result, we will see the following:



And we go in the browser to the page http: // <IP-address of the server> / postfixadmin /

We enter login and password for the created user.

Done.

4. Configuring Postfix

By default, Postfix is already installed in CentOS 7. But if the server does not exist without it, let's perform the installation with a simple command:

We create an account from which we will work with the virtual mailbox directory:

** First we create a group of **vmail** and guid **1024** , after - the user **vmail** with uid **1024** and home directory / **home / mail** . Note that on some systems, the group ID and user ID 1024 can be used. In this case, you must create another one, and in this manual, replace all 1024 with an alternative one.*

Now open the configuration file of the mail server for editing:

And we edit the following lines:

```
myorigin = $ mydomain
```

** This setting specifies which domain to substitute to the sender, if it is not specified in the FROM header.*

```
mydestination = $ myhostname, localhost. $ mydomain, localhost, $ mydomain
```

** specify for which domains we receive incoming mail.*

```
local_recipient_maps = unix: passwd.byname $ alias_maps
```

** indicate where to get the list of local users.*

```
mynetworks = 127.0.0.0/8
```

** Permit to send messages to the local server.*

```
alias_maps = hash: / etc / aliases
```

** indicate where to get the list of aliases.*

** You need to make sure that postfix will listen on all necessary interfaces, in this case, at all.*

Now at the end of the configuration file we will add the following:

```
virtual_mailbox_base = / home / mail
virtual_alias_maps = proxy: mysql: /etc/postfix/mysql_virtual_alias_maps.cf
virtual_mailbox_domains = proxy: mysql: /etc/postfix/mysql_virtual_domains_maps.cf
virtual_mailbox_maps = proxy: mysql: /etc/postfix/mysql_virtual_mailbox_maps.cf
virtual_minimum_uid = 1024
virtual_uid_maps = static: 1024
virtual_gid_maps = static: 1024
virtual_transport =
dovecot dovecot_destination_recipient_limit = 1
smtpd_sasl_auth_enable = yes
```

```
smtpd_sasl_exceptions_networks = $ mynetworks
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_sasl_type =
dovecot smtpd_sasl_path = private / auth
smtpd_tls_cert_file = /etc/ssl/mail/public.pem
smtpd_tls_key_file = /etc/ssl/mail/private.key
smtpd_use_tls = yes
smtpd_tls_auth_only = yes
smtpd_helo_required = yes
```

** where:*

- ***virtual_mailbox_base*** *- the basic way to store mailboxes on a UNIX system.*
- ***virtual_alias_maps*** *- format and path for storing aliases for virtual users.*
- ***virtual_mailbox_domains*** *- the format and path for storing virtual user domains.*
- ***virtual_mailbox_maps*** *- format and path for storing mailboxes for virtual users.*
- ***virtual_minimum_uid*** *- from which number to assign IDs to users.*
- ***virtual_uid_maps*** *- the identifier of the user from which messages are written.*
- ***virtual_gid_maps*** *- identifier of the group from which messages are written.*
- ***virtual_transport*** *- specifies the message delivery system.*
- ***dovecot_destination_recipient_limit*** *- the transfer of messages from Postfix to Dovecot is performed according to the specified number (in our example, 1 piece).*
- ***smtpd_sasl_auth_enable*** *- allows sasl authentication.*
- ***smtpd_sasl_exceptions_networks*** *- excluding networks from using encryption.*
- ***smtpd_sasl_security_options*** *- additional options for setting sasl.*
- ***broken_sasl_auth_clients*** *- this option is prescribed for MS Outlook clients.*

- ***smtpd_sasl_type*** *- indicates the type of authentication.*
- ***smtpd_sasl_path*** *- the path to the temporary information exchange files with Dovecot. It specifies either an absolute path or a relative queue_directory.*
- ***smtpd_tls_cert_file*** *- the full path to the public certificate.*
- ***smtpd_tls_key_file*** *- the full path to the private certificate.*
- ***smtpd_use_tls*** *- indicates to clients that TLS support is available.*
- ***smtpd_tls_auth_only*** *- use only TLS.*
- ***smtpd_helo_required*** *- request to start the session with a greeting.*

Create a file with the settings for accessing the database with aliases:

```
user = postfix
password = postfix123
hosts = localhost
dbname = postfix
query = SELECT goto FROM alias WHERE address = '% s' AND active = '1'
```

** where **user** and **password** - login and password to connect to MySQL; **hosts** - the name of the database server (in this case, the local server); **dbname** - the name of the database; **query** - a query template for data.*

Create a file with instructions for obtaining data on virtual domains:

```
user = postfix
password = postfix123
hosts = localhost
dbname = postfix
```

```
query = SELECT domain FROM domain WHERE domain = '% u'
```

And a file with mailboxes:

```
user = postfix
password = postfix123
hosts = localhost
dbname = postfix
query = SELECT CONCAT (domain, '/', maildir) FROM mailbox WHERE username = '% s' AND active = '1'
```

Open the file master.cf and add to the very end:

```
submission inet n - n - - smtpd
    -o smtpd_tls_security_level = may
    -o smtpd_sasl_auth_enable = yes
    -o smtpd_sasl_type = dovecot
    -o smtpd_sasl_path = / var / spool / postfix / private / auth
    -o smtpd_sasl_security_options = noanonymous
    -o smtpd_sasl_local_domain = $ myhostname
smtps inet n - n - - smtpd
    -o syslog_name = postfix / smtps
    -o smtpd_tls_wrappermode = yes
    -o smtpd_sasl_auth_enable = yes
    -o smtpd_client_restrictions = permit_sasl_authenticated, reject
dovecot unix - n n - - pipe
    flags = DRhu user = vmail: vmail argv = / usr / libexec / dovecot / deliver -d $ {recipient}
```

** Need to make sure that the contents of the file are no other options uncommented for* ***submission,*** **smtps** *and* ***dovecot*** *(by default, they are not). In this case, we configured postfix on ports 25, 465 and 587.*

Restart postfix:

5. Configuring Dovecot

Install Dovecot with the component to work with the DBMS:

We configure the way messages are stored:

```
mail_location = maildir: / home / mail /% d /% u /
first_valid_gid = 1024
```

** In this example, the messages will be stored in the advanced* ***maildir*** *format .*

Configure the listener for authentication:

```
service auth {
  unix_listener / var / spool / postfix / private / auth {
    mode = 0666
    user = postfix
    group = postfix
  }
  unix_listener auth-userdb {
```

```
    mode = 0600
    user = vmail
    group = vmail
  }
}
```

** Please note that / **var** / **spool** / **postfix** / **private** / **auth** is the same **private** / **auth** that was registered by us in postfix.*

Configure authentication in Dovecot:

```
#! include auth-system.conf.ext
! include auth-sql.conf.ext
```

** in this case, we just comment on the usual authentication and remove the comment for using sql-auttification.*

Configure the use of encryption:

```
ssl = required
ssl_cert = </etc/ssl/mail/public.pem
ssl_key = </etc/ssl/mail/private.key
```

** This setting will tell dovecot to require clients to use encryption.*

Let's configure automatic creation of directories when the user first connects to the mailbox:

```
lda_mailbox_autocreate = yes
```

We configure the connection to our database:

```
passdb {
  ...
  args = /etc/dovecot/sql.conf
}
userdb {
  ...
  args = /etc/dovecot/sql.conf
}
```

** In this example, we have specified a file that will contain the settings for retrieving users and passwords from the database.*

Create a file with the settings for working with mysql:

```
driver = mysql
connect = host = localhost dbname = postfix user = postfix password = postfix123
default_pass_scheme = MD5-CRYPT
password_query = SELECT password FROM mailbox WHERE username = '% u'
user_query = SELECT maildir, 1024 AS uid, 1024 AS gid FROM mailbox WHERE username = '% u'
user_query = SELECT CONCAT ('/ home / mail /', LCASE (`domain`), '/', LCASE (` maildir`)), 1024 AS uid,
1024 AS gid FROM mailbox WHERE username = '% u'
```

And, finally, configure the protocols and interface, which will listen to dovecot:

```
protocols = imap imaps pop3 pop3s
listen = *
```

** By default, dovecot also listens on ipv6 (**listen = *, ::)** . If the server does not use the 6th version of the TCP / IP protocol, errors will appear in the dovecot logs:*

master: Error: service (imap-login): listen (::, 143) failed

: service (imap-login): listen (::, 993) failed: Address

Generate security certificates
Create a catalog in which to place the certificates:

And we will generate them with the following command:

** the certificate is generated for **1461** days, **subj** keys can be arbitrary, **CN** must be specified in accordance with the name of the server by which we will connect to the mail.*

Run the dovecot:

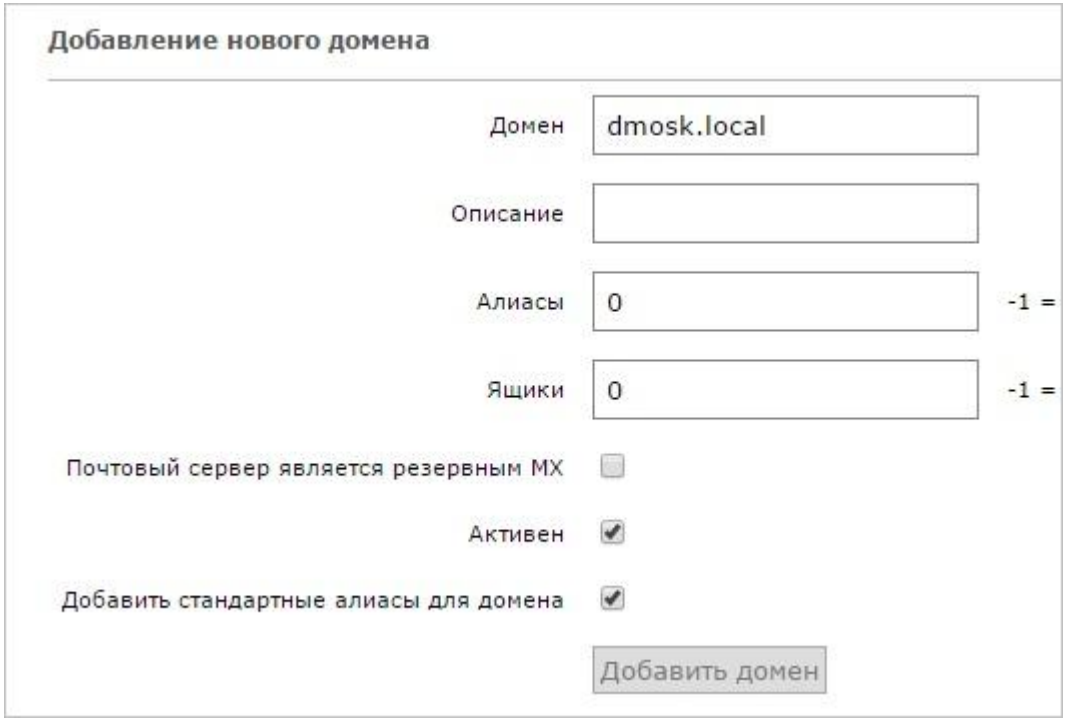
6. Create the first mailbox and check the server operation

In the browser, enter the path to Postfixadmin in the address bar - http: // <server IP address> / postfixadmin /.

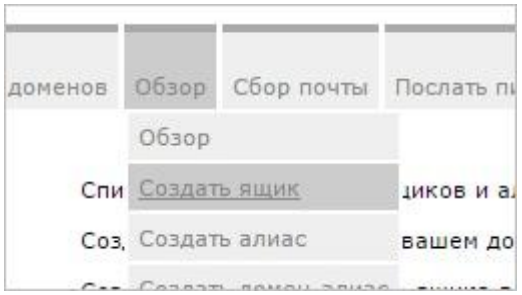
We enter login and password from the administrative account that we created in step 3. We will see the account management page.

Let's move on to the Domain List - New Domain:

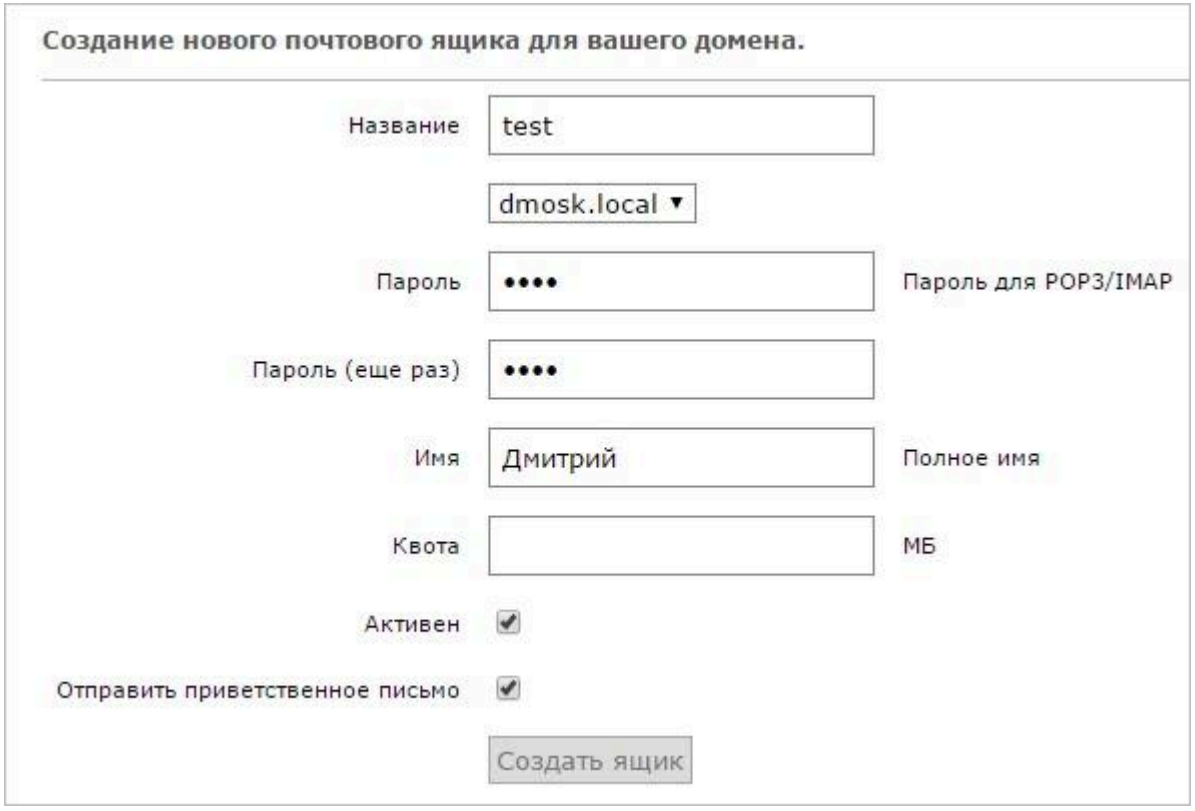
Fill out the forms and click on Add Domain:



Now go to Overview - Create a box:



Enter the data for the new user and click on Create a box:



Now you can connect to the server using any email program, for example, Mozilla Thunderbird.

Connection parameters:

- **Server** : the name of the server or its IP address (not desirable, since the certificate is issued by domain name).
- **IMAP** : 143 STARTTLS or 993 SSL / TLS
- **POP3** : 110 STARTTLS or 995 SSL / TLS
- **SMTP** : 25 STARTTLS or 465 SSL / TLS or 587 STARTTLS

7. Install and configure Roundcube Webmail

On the official site go to [the download page of](#) Roundcube. See the link to the latest stable version of the product:

Use the link to download the program archive:

Create a directory where the portal files will be located:

And unpack the downloaded archive:

Copy the config template:

And we open it for editing:

```
$ config ['db_dsnw'] = 'mysql: // roundcube: roundcube123 @ localhost / roundcubemail';
$ config ['enable_installer'] = true;
```

** We edit the first line, and add the second one. In the first line of the* **roundcube: roundcube123** - login and password for accessing the database; **localhost** - the database server; **roundcubemail** - the name of the database.

We set the apache owner to the portal folder:

Create a base in MariaDB for the roundcubemail:

And load the data into the created database:

Install the components necessary for the work of Roundcube:

Let's configure php:

```
date.timezone = "Europe / Moscow"
```
Reboot php-fpm:

Now open the browser and go to http: // <server IP address> / webmail / installer /. At the very bottom click on the **Next** button . If the button is inactive, check that there are no errors (**NOT OK**).

We check that all items are in the **OK** state .

After removing the folder with the installation scripts:

And go to the browser at http: // <server IP address> / webmail /.

8. Protect yourself from viruses

Installing and Configuring ClamAV
Install the antivirus:

Configuring postfix:

```
content_filter = scan: [127.0.0.1]: 10025
receive_override_options = no_address_mappings
```
** where **content_filter** points to an application that will scan messages; **receive_override_options**allows you to see the original email addresses of emails with viruses.*

Now edit master.cf:

We add the following:

```
scan unix - - n - 16 smtp
  -o smtp_send_xforward_command = yes
  -o smtp_enforce_tls = no

127.0.0.1:10026 inet n - n - 16 smtpd
  -o content_filter =
  -o receive_override_options = no_unknown_recipient_checks, no_header_body_checks
  -o smtpd_helo_restrictions =
  -o smtpd_client_restrictions =
  - o smtpd_sender_restrictions =
  -o smtpd_recipient_restrictions = permit_mynetworks, reject
  -o mynetworks_style = host
  -o smtpd_authorized_xforward_hosts = 127.0.0.0 / 8
```

Restart postfix:

Configure clamsmtpd:

```
ClamAddress: /var/run/clamd.scan/clamd.sock
TempDirectory: /var/run/clamd.scan
```

** Where **ClamAddress is** pointing to the path to the socket file - it must match the path in the configuration file for clam scan; **TempDirectory** - the path for storing temporary files.*

Editing the configuration file for clam scan:

```
PidFile /var/run/clamd.scan/clamd.pid
LocalSocket /var/run/clamd.scan/clamd.sock
User clamsmtp
```

** where **PidFile** is the path for the service pid file; **LocalSocket** - the path to the socket file for interaction with clamsmtp; **User** - the user from which clamd will be launched.*

Let's edit the owner on the directory for the socket file:

Now let's run the antivirus and run it:

Update

Open the config freshclam and put a comment opposite Example:

```
#Example
```

We resolve and start the service:

Run the update:

To configure the automatic update, we edit cron:

```
15 3 * * * / bin / freshclam
```

** In this example, every day at **03:15 the** clamav update process will be started.*

Checking

To check, we send a message with the following content:

X5O! P% @ AP [4 \ PZX54 (P ^) 7CC) 7} $ EICAR-STANDARD-ANTIVIRUS-TEST-FILE! $ H + H *

The letter should not reach.

9. Fighting with SPAM

Checking content with Spamassassin

Install spamassassin

Edit master.cf:

For smtp, add the following option:

```
smtp inet n - n - - smtpd
```

```
    -o content_filter = spamassassin
```

And add the following:

```
spamassassin unix - n n - - pipe
  flags = R user = spamd argv = / usr / bin / spamc -u spamd -e / usr / sbin / sendmail -f $ sender $
recipient
```

Updating spamassassin:

Let's start it and start the service:

Restart postfix:

For automatic updates, add the following to cron:

```
30 3 * * * / bin / sa-update
```

** The update will occur every day at **03:30** .*

To check the operation of the content antispam, send a message with the following content:

XJS * C4JDBQADN1.NSBN3 * 2IDNEN * GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL * C.34X

Antispam tools Postfix

MTA Postfix has its own mechanism for checking the headers of incoming messages. The rules are placed in 6 sections, which are processed in the following order:

client -> helo -> sender -> relay -> recipient -> data

And so, to configure the antispam in the **main.cf** configuration file, add:

```
= smtpd_client_restrictions
        permit_mynetworks
        permit_sasl_authenticated
        reject_unauth_pipelining
        permit

smtpd_helo_restrictions =
        permit

smtpd_sender_restrictions =
        permit_mynetworks
        permit_sasl_authenticated
        reject_non_fqdn_sender
        reject_unknown_sender_domain
        permit

smtpd_relay_restrictions =
        permit

smtpd_recipient_restrictions =
        permit_mynetworks
```

```
        permit_sasl_authenticated
        reject_non_fqdn_recipient
        reject_unauth_destination
        reject_unknown_recipient_domain
        reject_unverified_recipient
        permit

smtpd_data_restrictions =
        permit

smtpd_end_of_data_restrictions =
        permit
```

** these are more or less mild rules. They can be used for the first time while testing the server.*

To strengthen the protection we add:

```
smtpd_recipient_restrictions =
        ...
        reject_unknown_client_hostname
        reject_invalid_helo_hostname
        reject_non_fqdn_helo_hostname
        reject_unknown_helo_hostname
        reject_rbl_client bl.spamcop.net
        reject_rbl_client cbl.abuseat.org
        reject_rbl_client dul.ru
        reject_rbl_client dnsbl.abuse.ch
```

```
        permit
```

** where:*

- ***reject_unknown_client_hostname*** *- checks the presence of the PRT record of the sender and the availability of the working A-record in accordance with the PTR.*
- ***reject_invalid_helo_hostname*** *- Checks the HELO greeting syntax.*
- ***reject_non_fqdn_helo_hostname*** *- requires the correct FQDN name during the HELO greeting.*
- ***reject_unknown_helo_hostname*** *- forbids to be represented by names for which there is no A-record or MX.*
- ***reject_rbl_client*** *- checks the presence of the sender in blacklists.*

After making all edits, you need to restart Postfix:

The server is configured - you can use it.

Servers # UNIX # NGINX # MySQL

Install postfix + dovecot auth + tls + mysql + postfixadmin + postgrey + spamassassin and clamav on Centos 7

https://z0z0.me/install-postfix-dovecot-auth-tls-mysql-postfixadmin-postgrey-spamassassin-and-clamav-on-centos-7/

12 DECEMBER 2014 on mysql password, mariadb, postfix, dovecot, dovecot auth, tls, postfixadmin, spamassassin, postgrey

In this post we will install a mail server using virtual users with authentication using dovecot and ssl. After we are able to successfully send and retrieve e-mails we will securing the server with postgrey, spamassassin and clamav antivirus.

We will be using in this example a mysql backend. We will make sure that we have all the related packages installed. Will achieve that using yum:

```
#yum -y install postfix dovecot dovecot-mysql mariadb-server
```

When that's done we need to edit the `/etc/postfix/main.cf` to match your needed configuration:

```
alias_database = hash:/etc/aliases
alias_maps = hash:/etc/aliases
broken_sasl_auth_clients = yes
command_directory = /usr/sbin
config_directory = /etc/postfix
daemon_directory = /usr/libexec/postfix
data_directory = /var/lib/postfix
debug_peer_level = 2
debugger_command = PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin$daemon_directory/$process_name $process_id &
sleep 5
disable_vrfy_command = yes
html_directory = no
inet_interfaces = all
inet_protocols = all
mail_owner = postfix
mailq_path = /usr/bin/mailq.postfix
manpage_directory = /usr/share/man
```

```
milter_default_action = accept
mydestination = $myhostname, localhost.$mydomain, localhost
mydomain = example.com
myhostname = mail.example.com
myorigin = $myhostname
newaliases_path = /usr/bin/newaliases.postfix
queue_directory = /var/spool/postfix
readme_directory = /usr/share/doc/postfix-2.10.1/README_FILES
sample_directory = /usr/share/doc/postfix-2.10.1/samples
sendmail_path = /usr/sbin/sendmail.postfix
setgid_group = postdrop
smtpd_delay_reject = yes
smtpd_error_sleep_time = 1s
smtpd_hard_error_limit = 20
smtpd_helo_required = yes
smtpd_helo_restrictions = permit_mynetworks,
                      reject_non_fqdn_hostname,
                      reject_invalid_hostname,
                      permit
smtpd_recipient_restrictions = permit_sasl_authenticated,
                      permit_mynetworks,
                      check_policy_service unix:postgrey/socket,
                      reject_invalid_hostname,
                      reject_non_fqdn_hostname,
                      reject_unauth_destination,
                      reject_rbl_client list.dsbl.org,
                      reject_rbl_client sbl.spamhaus.org,
                      reject_rbl_client cbl.abuseat.org,
                      reject_rbl_client dul.dnsbl.sorbs.net,
                      permit
smtpd_sasl_auth_enable = yes
smtpd_sasl_path = private/auth
smtpd_sasl_security_options = noanonymous
```

```
smtpd_sasl_type = dovecot
smtpd_soft_error_limit = 10
smtpd_tls_CAfile = /etc/pki/tls/cert.pem
smtpd_tls_auth_only = yes
smtpd_tls_cert_file = /etc/pki/tls/certs/mail.example.com.crt
smtpd_tls_key_file = /etc/pki/tls/private/mail.example.com.key
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_security_level = may
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
unknown_local_recipient_reject_code = 550
virtual_alias_maps = mysql:/etc/postfix/sql/mysql_virtual_alias_maps.cf
virtual_gid_maps = static:12
virtual_mailbox_base = /home/vmail
virtual_mailbox_domains = mysql:/etc/postfix/sql/mysql_virtual_domains_maps.cf
virtual_mailbox_limit = 51200000
virtual_mailbox_maps = mysql:/etc/postfix/sql/mysql_virtual_mailbox_maps.cf
virtual_minimum_uid = 8
virtual_transport = virtual
virtual_uid_maps = static:8
virtual_maildir_limit_message=Sorry, Your maildir has overdrawn your diskspace quota, please free some space of your
mailbox and try again.
virtual_mailbox_limit_maps=mysql:/etc/postfix/sql/mysql_virtual_mailbox_limit_maps.cf
virtual_mailbox_limit_override=yes
virtual_overquota_bounce=yes
virtual_create_maildirsize=yes
smptd_tls_session_cache_database=btree:/var/spool/postfix/smtpd_tls_cache
virtual_mailbox_extended=yes
smtpd_tls_note_starttls_offer=yes
```

You need to add create the following files:

/etc/postfix/sql/mysql_relay_domains_maps.cf

```
user = postfix
password = someotherpass
hosts = 127.0.0.1
dbname = mail
table = domain
select_field = domain
additional_conditions = and backupmx = '1'
```

/etc/postfix/sql/mysql_virtual_alias_maps.cf

```
user = postfix
password = someotherpass
hosts = 127.0.0.1
dbname = mail
table = alias
select_field = goto
where_field = address
```

/etc/postfix/sql/mysql_virtual_mailbox_limit_maps.cf

```
user = postfix
password = someotherpass
hosts = 127.0.0.1
dbname = virtual_mail
```

```
table = mailbox
select_field = quota
where_field = username
#additional_conditions = and active = '1'
```

/etc/postfix/sql/mysql_virtual_mailbox_maps.cf

```
user = postfix
password = someotherpass
hosts = 127.0.0.1
dbname = mail
table = mailbox
select_field = maildir
where_field = username
#additional_conditions = and active = '1'
```

/etc/postfix/sql/mysql_virtual_domains_maps.cf

```
user = postfix
password = someotherpass
hosts = 127.0.0.1
dbname = mail
table = domain
select_field = domain
where_field = domain
additional_conditions = and backupmx = '0' and active = '1'
```

To configure the smtps and submission for local delivery change the following in `/etc/postfix/master.cf`

```
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated,reject

smtps     inet  n       -       n       -       -       smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated, reject
  -o milter_macro_daemon_name=ORIGINATING
```

Next we start the mysql database and create a database for our installation. To start the database please run the following command:

```
#systemctl start mariadb.service
```

Make sure that your database will start upon startup:

```
#systemctl enable mariadb.service
```

Now it's time to create our database and the proper users with the needed privileges:

```
#mysql -u root
MariaDB> create database mail;
MariaDB> grant all privileges on mail.* to 'postfixadmin'@'localhost' identified by 'somepass';
MariaDB> grant select on mail.* to 'postfix'@'localhost' identified by 'someotherpass';
MariaDB> flush privileges;
```

We have created two users with different privileges. First the postfixadmin is the user which would be used by the webgui called postfixadmin which need write privileges into the database because you will be adding the domains, users and aliases which will be holded by this installation via it. The other user is postfix, which will be used by the postfix and dovecot to query the users from the database. These queries does not write to the database therefore this user will only have select right granted.

Now it's time to install nginx and the postfixadmin. First you need to add a new repo for nginx cause we want to use the latest version available. For that create a new file `/etc/yum.repos.d/nginx.repo` with the following value in it.

```
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/mainline/centos/7/$basearch/
gpgcheck=0
enabled=1
```

Now we need to add the epel repo so that we can install the full list of needed packages.

```
#yum -y install http://dl.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-2.noarch.rpm
```

Now we can install all the needed packages. To do so you will need to run the command:

```
#yum -y install php php-fpm php-common php-pdo php-mbstring php-imap php-cli php-mysql nginx
```

Edit the `/etc/php-fpm.d/www.conf` and change the following entries:

```
;listen = 127.0.0.1:9000
listen = /var/run/php-fpm.sock
...
listen.owner = nginx
listen.group = nginx
listen.mode = 0666
...
user = nginx
group = nginx
```

Download the postfixadmin from here.

Unarchive the downloaded package and move it to `/var/www/html`:

```
#tar -xzvf postfixadmin-2.91.tar.gz
#mv postfixadmin-2.91 /var/www/html/postfixadmin
```

Now it's time to set up the nginx and start the services. Edit the `/etc/nginx/conf.d/default.conf` and change it to look like this:

```
server {
  listen       80;
  server_name  0.0.0.0;

  error_log /var/log/nginx/error.log debug;

  root   /var/www/html/postfixadmin;
  index  index.php index.html index.htm;

  location ~ \.php$ {
    fastcgi_pass   unix:/var/run/php-fpm.sock;
    fastcgi_index  index.php;
    fastcgi_param  SCRIPT_FILENAME      $document_root$fastcgi_script_name;
    include        fastcgi_params;
  }

}
```

Now go and edit the postfixadmin config file `/var/www/potfixadmin/html/config.inc.php`. Change the value false to true at the line:

```
$CONF['configured'] = true;
```

Edit the entries in the config file related to the database:

```
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'postfixadmin';
$CONF['database_password'] = 'somepassword';
```

```
$CONF['database_name'] = 'mail';
```

Save the changes. Disable your selinux by editing the config file and change the SELINUX value from enforcing to permissive or disabled. To apply the change you will need to reboot your server. You will have to add a rule to the firewall to open the port 80 for web access. To do that you need to run the following command:

```
#firewall-cmd --zone=public --add-service=http --permanent
```

Then restart the firewall and start the php-fpm and nginx services:

```
#systemctl restart firewalld
#systemctl start php-fpm
#systemctl start nginx
```

Now you can open a browser and type the IP address of your system to get to the postfixadmin page.

It's time to create the self signed certificate for your domain in this example mail.example.com.

```
#genkey --days 3650 mail.example.com
```

This certificate is valid for 10 years.

Now it's time to configure the dovecot. First create a file called `/etc/dovecot/dovecot-sql.conf.ext`

```
driver = mysql
connect = host=localhost dbname=mail user=postfix password=P0stf1x
default_pass_scheme = CRAM-MD5
user_query = SELECT '/home/vmail/%n@%d' AS home, 8 AS uid, 12 AS gid FROM mailbox WHERE username = '%u'
password_query = SELECT password FROM mailbox WHERE username = '%u'
```

Next file to edit `/etc/dovecot/conf.d/10-auth.conf`

Change the values to match the following:

```
auth_mechanisms = plain login cram-md5
#!include auth-deny.conf.ext
#!include auth-master.conf.ext

#!include auth-system.conf.ext
!include auth-sql.conf.ext
#!include auth-ldap.conf.ext
#!include auth-passwdfile.conf.ext
#!include auth-checkpassword.conf.ext
#!include auth-vpopmail.conf.ext
#!include auth-static.conf.ext
```

Next file to change is `/etc/dovecot/conf.d/10-mail.conf`:

```
mail_location = maildir:/home/vmail/%d/%n/:INDEX=/home/vmail/%d/%n/indexes
mail_uid = 8
```

```
mail_gid = 12
first_valid_uid = 8
last_valid_uid = 8
first_valid_gid = 12
last_valid_gid = 12
```

Next file to change is `/etc/dovecot/conf.d/10-master.conf`:

```
service imap-login {
    inet_listener imap {
        port = 143
    }
    inet_listener imaps {
        port = 993
        ssl = yes
    }
}

service pop3-login {
     inet_listener pop3 {
         #port = 110
    }
    inet_listener pop3s {
        #port = 995
        #ssl = yes
    }
}

service auth {
  unix_listener /var/spool/postfix/private/auth {
      mode = 0666
```

```
    user = mail
    group = mail
  }
}
```

Next file to change `/etc/dovecot/conf.d/10-ssl.conf`:

```
ssl = required
ssl_cert = </etc/pki/tls/certs/mail.example.com.crt
ssl_key = </etc/pki/tls/private/mail.example.com.key
```

Next file on line: `/etc/dovecot/conf.d/auth-sql.conf.ext`:

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

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

Note: There are only posted the values which need to be changed in the configuration files everything else should be left as it was on the config file by default.

Now before we start the services we would need to open the ports in the firewall. To do so run the following commands:

```
#firewall-cmd --zone=public --add-service=smtp --permanent
#firewall-cmd --zone=public --add-service=smtp --permanent
#firewall-cmd --zone=public --add-service=smtps --permanent
#firewall-cmd --zone=public --add-port=587/tcp --permanent
#firewall-cmd --zone=public --add-service=imaps --permanent
#firewall-cmd --zone=public --add-service=imap --permanent
#firewall-cmd --zone=public --add-service=pop3s --permanent
```

After we have added the ports we need to restart the firewall so that the rules would get applied.

```
#systemctl restart firewalld
```

Now we are ready to start up the service and enable them on boot.

```
#systemctl enable postfix
#systemctl restart postfix
#systemctl enable dovecot
#systemctl restart dovecot
```

We test the smtp by telneting into the domain on port 25

```
#telnet localhost 25
Connected to localhost.
```

```
Escape character is '^]'.
220 mail.example.com ESMTP Postfix
ehlo me
250-mail.example.com
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
```

First need to install the required software. For this we use yum:

```
#yum install -y postgrey spamassassin spamass-milter-postfix spamass-milter clamav-filesystem clamav-server clamav-update clamav-milter-systemd clamav-data clamav-server-systemd clamav-scanner-systemd clamav clamav-milter clamav-lib clamav-scanner
```

Once that is done it's time to configure the software. Let's start with the spamassassin. Fist edit the `/etc/mail/spamassassin/local.cf` match your configuration with this:

```
required_hits 5.0
report_safe 0
required_score 5
#rewrite_header Subject *** SPAM ***
remove_header ham Status
```

```
remove_header ham Level
```

Next file to configure is `/etc/sysconfig/spamassassin`:

```
# Options to spamd
SPAMDOPTIONS="-d -c -m5 -H"
```

and `/etc/sysconfig/spamass-milter`:

```
EXTRA_FLAGS="-i 127.0.0.1 -m -r -1 -I "
```

and the last one for spamassassin is: `/etc/sysconfig/spamass-milter-postfix`:

```
SOCKET="/run/spamass-milter/postfix/sock"
SOCKET_OPTIONS="-g postfix"
```

Next software to configure is clamav. First will edit the `/etc/clamd.d/scan.conf`

First thing to do is to comment or remove the Example in the cofiguration file. These are the configuration options which should be uncommented:

```
LogFile /var/log/clamd.scan
LogFileMaxSize 0
LogTime yes
```

```
LogSyslog yes
LogFacility LOG_MAIL
PidFile /var/run/clamd.scan/clamd.pid
TemporaryDirectory /var/tmp
DatabaseDirectory /var/lib/clamav
OfficialDatabaseOnly no
LocalSocket /var/run/clamd.scan/clamd.sock
LocalSocketGroup clamscan
LocalSocketMode 666
FixStaleSocket yes
TCPSocket 3310
TCPAddr 127.0.0.1
SelfCheck 600
User clamscan
AllowSupplementaryGroups yes
StatsEnabled yes
```

Next file to configure is `/etc/mail/clamav-milter.conf`:

```
MilterSocket /var/run/clamav-milter/clamav-milter.socket
MilterSocket inet:7357
MilterSocketMode 777
FixStaleSocket yes
AllowSupplementaryGroups yes
PidFile /var/run/clamav-milter/clamav-milter.pid
TemporaryDirectory /var/tmp
ClamdSocket unix:/var/run/clamd.scan/clamd.sock
OnClean Accept
OnInfected Reject
OnFail Defer
RejectMsg Rejecting harmful email: %v found.
```

```
AddHeader Add
VirusAction /usr/local/bin/my_infected_message_handler
LogFileMaxSize 0
LogTime yes
LogSyslog yes
LogFacility LOG_MAIL
```

Now let's try and configure postgrey. For this you need to edit `/etc/sysconfig/postgrey`:

```
POSTGREY_OPTS="--delay=60"
```

Last piece of software to install and configure is the opendkim to do this you need to run the following command:

```
#yum install -y libopendkim opendkim
```

Let's start to configure `/etc/opendkim.conf`:

```
PidFile /var/run/opendkim/opendkim.pid
Mode    sv
Syslog  yes
SyslogSuccess   yes
UserID  opendkim:opendkim
Socket  inet:8891@localhost
Umask   002
Canonicalization    relaxed/simple
MinimumKeyBits 1024
KeyTable    refile:/etc/opendkim/KeyTable
```

```
SigningTable    refile:/etc/opendkim/SigningTable
ExternalIgnoreList  refile:/etc/opendkim/TrustedHosts
InternalHosts   refile:/etc/opendkim/TrustedHosts
```

Let's specify the TrustedHosts at `/etc/opendkim/TrustedHosts`:

```
127.0.0.1
mail.example.com
example.com
#192.168.1.0/24
```

This file is used to define the internal ignore list and external ignore list. All emails which are orriginating from these hosts or domains will be automatically trusted and signed.

Now we create the keytable on `/etc/opendkim/KeyTable`:

```
default._domainkey.example.com example.com:default:/etc/opendkim/keys/default.private
```

Next we create the signing table at `/etc/opendkim/SigningTable`:

This file is used to declare the domains/email addresses and their selector

```
*@example.com default._domainkey.example.com
```

Now it's time to create the private and public keys. First change the directory to the `/etc/opendkim/keys` and create a folder in there named as your domain for which you want to create the keys and fix the ownership of the folder:

```
#cd /etc/opendkim/keys
#mkdir example.com
#cd example.com
#chown -R opendkim.opendkim /etc/opendkim/keys/example.com
```

Now let's generate the keys for the domain and set the proper ownership:

```
#opendkim-genkey -s default -d example.com
#chown -R opendkim. /etc/opendkim/keys/
```

The -s option would give the name of the selector and the -d would give the domain of the keys.

Now open the `/etc/opendkim/keys/example.com/mail.txt` and create a TXT entry on your domain DNS server according to what is in the file:

```
mail._domainkey IN TXT "v=DKIM1; k=rsa;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDFRSHhfOAoqsF5t8dmpzuYGQAIP3JoLRVLQ5r8SsI6SrS90VGAsAByJsERI
mLxV68D3b+iDUhaw+PmQ/9L4atgnp3cbqgNRD++lAOvloGe+I1GlN6iKYNm4Xr3IYPPvaZYlT0JgawRZKGffP0l0bCqJT/36ZCcI+S
hoq+l/H0LfwIDAQAB"
```

Now let's do the final changes in the postfix config file `/etc/postfix/main.cf`. Add the following entries to the file:

```
milter_default_action = accept
smtpd_milters = unix:/var/run/clamav-milter/clamav-milter.socket,
            unix:/var/run/spamass-milter/postfix/sock,
            inet:127.0.0.1:8891
```

Once that is done it's time to enable the services for startup and start them and restart the postfix.

```
#systemctl enable spamassassin.service
#systemctl start spamassassin.service
#systemctl enable spamass-milter.service
#systemctl enable spamass-milter-root.service
#systemctl start spamass-milter.service
#systemctl start spamass-milter-root.service
#systemctl enable clamd@scan.service
#systemctl start clamd@scan.service
#systemctl enable clamav-milter.service
#systemctl start clamav-milter.service
#systemctl enable postgrey.service
#systemctl start postgrey.service
#systemctl enable opendkim.service
#systemctl start opendkim.service
#systemctl restart postfix.service
```

AWS EC2 of Centos7 assume, it is a note of when you build Postfix, Dovecot, amavisd, in Postfixadmin.

https://qiita.com/kaoken/items/656df7b9a4e82fdf0163

Registration of domain, Elastic IP assignment, port 22, 25, 80, 110, 443, 465, 587, 995, assume that you have open.

Connection configuration of the mail server

type	port number	The type of encryption
SMTP	25	--
SMTPS	465	SSL/TLS
SMTP	587	STARTTLS

POP3	110	STARTTLS
POP3S	995	SSL/TLS

Each version at the time of construction

Apps such as	**version**
Apache	2.4.6
PHP	7.0.5
Mysql	5.6.30
Postfix	2.10.1
Dovecot	2.2.10
postfixadmin	2.93

This time, you will type to access the sub-domain in such a way as described below.

https:. // admin domain name / postfixadmin

The administrator of the creation and authority to deal with Web server

Mainly, phpAdmin, be used, for example, postfixadmin

Repository add

```
$ sudo groupadd -g 20000 webadmin
$ sudo useradd -g webadmin -u 20000 webadmin
$ sudo passwd webadmin
$ sudo -u webadmin mkdir -p /home/webadmin/www/public_html
$ sudo chown -R webadmin:webadmin /home/webadmin/www
```

webadmin users, so get used to in the sudo vmail user only "sudo -u vmail".

```
$ sudo vim /etc/sudoers.d/webadmin
```

Creating a /etc/sudoers.d/webadmin

```
Add in order to enable the sudo command in exec () in the # php.
Defaults:webadmin !requiretty
webadmin          ALL=(vmail)  NOPASSWD:ALL
```

su command is to handle only user that belongs to the wheel group.

```
$ sudo vim /etc/login.defs
```

Added to the end

Editing of /etc/login.defs

```
SU_WHEEL_ONLY yes
```

Editing of files

```
$ sudo vim /etc/pam.d/su
```

Remove the comment

Editing of /etc/pam.d/su

```
auth required /lib/security/pam_wheel.so use_uid
```

Construction of WEB server

If you already pre-built, you may skip.

Installation and service registration of

Repository add

I needed to put the php7. By default at the moment 5.4

Repository add

```
$ sudo yum install epel-release
$ sudo rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-7.rpm
```

Apache & PHP7

web server

```
$ sudo yum install -y gcc httpd httpd-devel mod_ssl libcap-devel
$ sudo yum install --enablerepo=remi,remi-php70 php php-mbstring php-mcrypt php-opcache php-mysqlnd php-gd php-devel php-imap php-gd php-pdo
php-pear
$ sudo systemctl enable httpd.service
```

Set of php

web server

```
$ Sudo vim /etc/php.ini
```

/etc/php.ini

```
date.timezone = 'Asia/Tokyo'
include_path = ".:/usr/share/php:/usr/share/pear"
```

Creating a mod_ruid2

For each virtual host, because I wanted to change the authority, use the mod_ruid2.

※ To enable mod_ruid2 is, SELinux please be disabled .

First of all, let's DL and decompress from the following site.

https://github.com/mind04/mod-ruid2

If you have not installed

```
$ sudo yum install wget unzip
```

DL and decompression and build

```
$ wget https://github.com/mind04/mod-ruid2/archive/master.zip
$ unzip master.zip
$ sudo apxs -i -c -l cap mod-ruid2-master/mod_ruid2.c
```

After building

```
----------------------------------------------------------------------
Libraries have been installed in:
   /usr/lib64/httpd/modules
```

```
If you ever happen to want to link against installed libraries
in a given directory, LIBDIR, you must either use libtool, and
specify the full pathname of the library, or use the `-LLIBDIR'
flag during linking and do at least one of the following:
   - add LIBDIR to the `LD_LIBRARY_PATH' environment variable
     during execution
   - add LIBDIR to the `LD_RUN_PATH' environment variable
     during linking
   - use the `-Wl,-rpath -Wl,LIBDIR' linker flag
   - have your system administrator add LIBDIR to `/etc/ld.so.conf'

See any operating system documentation about shared libraries for
more information, such as the ld(1) and ld.so(8) manual pages.
----------------------------------------------------------------------
chmod 755 /usr/lib64/httpd/modules/mod_ruid2.so
```

After the build, if such a message is Dere, it will be available.

Usually, but it will be written automatically in the httpd.conf, If you do not, please write the following things.

/etc/httpd/conf/httpd.conf

```
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule ruid2_module modules/mod_ruid2.so
```

Create a virtual host to new use in postfixadmin.

Creating a virtual host

```
$ sudo vim /etc/httpd/conf.d/admin.conf
```

/etc/httpd/conf.d/admin.conf

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerName admin.ドメイン名:443
    ServerAlias ドメイン名
    DocumentRoot /home/webadmin/www/public_html
    <Directory "/home/webadmin/www/public_html">
      Options -Indexes
      Require all granted
      Allow from all
      AllowOverride All
    </Directory>

    <IfModule mod_ruid2.c>
        RMode    config
        RUidGid  webadmin webadmin
        #RGroups  apachetmp
    </IfModule>

    SSLEngine on

    SSLCertificateFile    /etc/pki/tls/certs/localhost.crt
    SSLCertificateKeyFile /etc/pki/tls/private/localhost.key

    ErrorLog logs/admin_ssl_err
</VirtualHost>
</IfModule>
```

Please be set according to the individual environment. In addition, you might may be added the BASIC authentication.

By the way, in the / home / webadmin / www / public_html , you put the postfixadmin.

Besides, phpMyAdmin, etc., might be a good even put a web application for the administrator.

/ Home / in the webadmin / www / public_html, please upload the index.php below.

index.php

```
<?php
header('HTTP', true, 404);
?>
404 (Not Found)
```

Start-up of the web server

```
$ sudo systemctl start httpd.service
```

https:. // admin domain name /

After starting the web server, 404 (Not Found) Please make sure whether is displayed.

Construction of Mysql5.6

Installation and service registration of

Add Repository

I needed to put the php5.6. Defaults to 5.5 at the moment.

Add Repository

```
$ sudo yum -y install http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm
```

DB

```
$ sudo yum -y install mysql-community-server mysql mysql-devel mysql-server mysql-utilities
$ sudo systemctl enable mysqld.service
$ sudo systemctl start mysqld.service
```

DB

```
$ mysql -u root
```

Initial setting

Decided the root password, remove anonymous user.

DB

```
mysql> UPDATE mysql.user SET Password=PASSWORD('ルートパスワード') WHERE User='root';
mysql> DELETE FROM mysql.user WHERE User = '';
mysql> FLUSH PRIVILEGES;
mysql> EXIT;
```

phpMyAdmin no Susume

https://www.phpmyadmin.net/ than, download and unzip and

it is recommended that you put the phpMyAdmin to / home / webadmin / www / public_html .

In addition to setting a little, you can immediately use. Detailed explanation, please Gugutsu.

Be used in the e-mail server, create users and DB

It is already connected.

Creating a user and DB

```
mysql> CREATE USER 'postfix'@'localhost' IDENTIFIED BY 'postfixのパスワード';
mysql> CREATE DATABASE postfix CHARACTER SET utf8;
mysql> GRANT ALL ON postfix.* to postfix@localhost;
mysql> EXIT;
```

Construction of the mail server

Installation and service registration of

```
$ sudo yum install -y postfix dovecot dovecot-mysql
$ sudo systemctl enable postfix.service
$ sudo systemctl enable dovecot.service
```

Instead of cutting the MTA

In the initial state, Sendmail.Postfix So, the need to switch is not.

Creating Users and e-mail directory for the virtual domain

```
$ sudo groupadd -g 10000 vmail
$ sudo useradd -g vmail -u 10000 vmail /var/vmail
$ sudo mkdir -p -m 0700 /var/vmail/{vhosts,bin,deleted-vhosts}
$ sudo chown -R vmail:vmail /var/vmail
```

Create an encrypted file for smtpd

dh2048.pem and dh512.pem , the file Postfix SMTP server has the DH parameters used in EDH encryption.

cert.pem is, of PEM format, Postfix SMTP server RSA certificate file with the.

privkey.pem is, in PEM format, file with the Postfix SMTP server RSA private key.

```
$ sudo mkdir -p /etc/postfix/ssl/dhparams/
$ sudo openssl dhparam -out /etc/postfix/ssl/dhparams/dh2048.pem 2048
$ sudo openssl dhparam -out /etc/postfix/ssl/dhparams/dh512.pem 512
$ sudo mkdir -p /etc/postfix/ssl/selfsigned/
$ sudo openssl req -new -newkey rsa:4096 -days 3658 -sha256 -nodes -x509 \
      -subj "/C=JP/ST=Shizuoka/L=Shizuoka/O=Mailserver certificate/OU=Mail/CN=www.ドメイン名/emailAddress=admin@ドメイン名" \
      -keyout /etc/postfix/ssl/selfsigned/privkey.pem \
      -out /etc/postfix/ssl/selfsigned/cert.pem
```

-subj, please tailored to the individual environment.

Set of postfix

Query-building for access to the database

Create a directory

```
$ sudo mkdir -p /etc/postfix/mysql
```

virtual-alias-maps.cf

```
$ sudo vim /etc/postfix/mysql/virtual-alias-maps.cf
```

/etc/postfix/mysql/virtual-alias-maps.cf

```
hosts    = localhost
user     = postfix
password = postfix password
dbname   = postfix

query = SELECT goto FROM alias WHERE address='%s' AND active = 1
```

virtual-mailbox-domains.cf

virtual-mailbox-domains.cf

```
$ sudo vim /etc/postfix/mysql/virtual-mailbox-domains.cf
```

/etc/postfix/mysql/virtual-mailbox-domains.cf

```
hosts    = localhost
user     = postfix
password = postfix password
dbname   = postfix

query = SELECT domain FROM domain WHERE domain='%s' and backupmx = 0 and active = 1
```

virtual-mailbox-maps.cf

virtual-mailbox-maps.cf

```
$ sudo vim /etc/postfix/mysql/virtual-mailbox-maps.cf
```

/etc/postfix/mysql/virtual-mailbox-maps.cf

```
hosts    = localhost
user     = postfix
password = postfix password
dbname   = postfix

query = SELECT maildir FROM mailbox WHERE username='%s' AND active = 1
```

virtual-mailbox-limit-maps.cf

virtual-mailbox-limit-maps.cf

```
$ sudo vim /etc/postfix/mysql/virtual-mailbox-limit-maps.cf
```

/etc/postfix/mysql/virtual-mailbox-limit-maps.cf

```
user = postfix
password = postfix password
hosts = localhost
dbname = postfix
query = SELECT quota FROM mailbox WHERE username='%s' AND active = '1'
```

Edit the main.cf

Edit the main.cf

```
$ sudo vim /etc/postfix/main.cf
```

/etc/postfix/main.cf

```
myhostname = mail. domain name
mydomain = domain name
myorigin = $mydomain
inet_interfaces = all
inet_protocols = ipv4
mydestination = localhost localhost.$mydomain
mynetworks = 127.0.0.0/8
recipient_delimiter  = +
header_checks = regexp:/etc/postfix/header_checks
mime_header_checks = regexp:/etc/postfix/header_checks
smtpd_banner = ESMTP $mail_name
readme_directory = no
mailbox_command = procmail -a "$EXTENSION"

biff                 = no
append_dot_mydomain  = no
delay_warning_time   = 4h
disable_vrfy_command = yes
message_size_limit   = 51200000
mailbox_size_limit   = 102400000


#
# List of error classes that are reported to the postmaster
```

```
# If the way is sent to frequently, commented OK
#
notify_classes = resource, software
error_notice_recipient     = admin@ドメイン名


#
# Smtp
#
smtp_tls_loglevel             = 1
smtp_tls_security_level       = may
#smtp_tls_CAfile               =
smtp_tls_protocols            = !SSLv2, !SSLv3
smtp_tls_mandatory_protocols = !SSLv2, !SSLv3
smtp_tls_mandatory_ciphers    = high
smtp_tls_exclude_ciphers = Anull, eNULL, Export, DES, 3DES, RC2, RC4, MD5, PSK, SRP, DSS, AECDH, abandon
smtp_tls_note_starttls_offer = yes

#
# Smtpd
#
smtpd_tls_loglevel             = 1
smtpd_tls_auth_only            = yes
smtpd_tls_security_level       = may
smtpd_tls_received_header      = yes
smtpd_tls_protocols            = !SSLv2, !SSLv3
smtpd_tls_mandatory_protocols = !SSLv2, !SSLv3
smtpd_tls_mandatory_ciphers    = medium

#smtpd_tls_CAfile               = $smtp_tls_CAfile
smtpd_tls_cert_file            = /etc/postfix/ssl/selfsigned/cert.pem
smtpd_tls_key_file             = /etc/postfix/ssl/selfsigned/privkey.pem
smtpd_tls_dh1024_param_file    = /etc/postfix/ssl/dhparams/dh2048.pem
smtpd_tls_dh512_param_file     = /etc/postfix/ssl/dhparams/dh512.pem

tls_preempt_cipherlist = yes
tls_random_source      = dev:/dev/urandom
```

```
smtp_tls_session_cache_database  = btree:${data_directory}/smtp_scache
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
lmtp_tls_session_cache_database  = btree:${data_directory}/lmtp_scache


#
# SASL
#

smtpd_sasl_auth_enable          = yes
smtpd_sasl_type                 = dovecot
smtpd_sasl_path                 = private/auth
smtpd_sasl_security_options     = noanonymous
smtpd_sasl_tls_security_options = $smtpd_sasl_security_options
smtpd_sasl_local_domain         = $mydomain
smtpd_sasl_authenticated_header = yes

broken_sasl_auth_clients = yes

#
# Virtual mail box
#

virtual_uid_maps        = static:10000
virtual_gid_maps        = static:10000
virtual_minimum_uid     = 10000
virtual_mailbox_base    = /var/vmail/vhosts
virtual_transport       = lmtp:unix:private/dovecot-lmtp
virtual_mailbox_domains = mysql:/etc/postfix/mysql/virtual-mailbox-domains.cf
virtual_mailbox_maps    = mysql:/etc/postfix/mysql/virtual-mailbox-maps.cf
virtual_alias_maps      = mysql:/etc/postfix/mysql/virtual-alias-maps.cf

# System capacity limit
virtual_create_maildirsize = yes
virtual_mailbox_extended   = yes
virtual_mailbox_limit      = 102400000
virtual_mailbox_limit_maps = mysql:/etc/postfix/mysql/virtual-mailbox-limit-maps.cf
virtual_mailbox_limit_override = yes
```

```
virtual_maildir_limit_message = Sorry, the user's maildir has overdrawn his diskspace quota, please try again later.
virtual_overquota_bounce = yes


#
# Access restrictions
#

smtpd_recipient_restrictions =
     permit_mynetworks,
     permit_sasl_authenticated,
     reject_non_fqdn_recipient,
     reject_unauth_destination,
     reject_unknown_recipient_domain,
     reject_rbl_client all.rbl.jp,
     reject_rbl_client bl.spamcop.net,
     reject_rbl_client zen.spamhaus.org

smtpd_helo_restrictions =
     permit_mynetworks,
     permit_sasl_authenticated,
     reject_invalid_helo_hostname,
     reject_non_fqdn_helo_hostname

smtpd_client_restrictions =
     permit_mynetworks,
     permit_inet_interfaces,
     permit_sasl_authenticated

smtpd_sender_restrictions =
     reject_non_fqdn_sender,
     reject_unknown_sender_domain
```

mynetworks , please tailored to the individual environment.

Editing of header_checks

It sets the header check file.

header_checks

```
$ sudo vim /etc/postfix/header_checks
```

/etc/postfix/header_checks

```
/^Received:.*with ESMTPSA/  IGNORE
/^X-Originating-IP:/        IGNORE
/^X-Mailer:/                IGNORE
/^User-Agent:/              IGNORE
```

Added as needed

header_checks

```
$ Sudo folders / etc / postfix / header_checks
```

Edit the master.cf

header_checks

```
$ sudo vim /etc/postfix/master.cf
```

/etc/postfix/header_checks

```
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_dh1024_param_file=/etc/postfix/ssl/dhparams/dh2048.pem
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
#  -o milter_macro_daemon_name=ORIGINATING
465       inet  n       -       n       -       -       smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_reject_unlisted_recipient=no
  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
```

Set of Dovecot

Editing of dovecot.conf

Editing of dovecot.conf

```
$ sudo vim /etc/dovecot/dovecot.conf
```

/etc/dovecot/dovecot.conf

```
protocols = pop3 lmtp
listen = *
#mail_debug = yes
```

At first, it might be better to wait and see in the ** mail_debug = yes **.

Creating a dovecot-sql.conf.ext

Editing of dovecot.conf

```
$ sudo vim /etc/dovecot/dovecot-sql.conf.ext
```

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

```
driver   = mysql
connect  = host=localhost dbname=postfix user=postfix password=postfixパスワード
default_pass_scheme = SHA512-CRYPT
password_query = SELECT password FROM mailbox WHERE username = '%u' AND active='1'
user_query = SELECT CONCAT('/var/vmail/vhosts/', maildir) AS mail, 10000 AS uid, 10000 AS gid, CONCAT('*:bytes=', quota) AS quota_rule FROM mailbox
```

```
WHERE username = '%u' AND active='1'
```

10-auth.conf editing of

10-auth.conf editing of

```
$ sudo vim /etc/dovecot/conf.d/10-auth.conf
```

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

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

10-mail.conf editing of

10-mail.conf editing of

```
$ sudo vim /etc/dovecot/conf.d/10-mail.conf
```

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

```
mail_location = maildir:/var/vmail/vhosts/%d/%n/
maildir_stat_dirs=yes

mail_uid = 10000
mail_gid = 10000

first_valid_uid = 10000
```

```
last_valid_uid  = 10000

mail_privileged_group = vmail
```

10-master.conf editing of

10-mail.conf editing of

```
$ sudo vim /etc/dovecot/conf.d/10-master.conf
```

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

```
service pop3-login {
  inet_listener pop3 {
    port = 110
  }
  inet_listener pop3s {
    port = 995
    ssl = yes
  }
}

service lmtp {
  unix_listener /var/spool/postfix/private/dovecot-lmtp {
    mode = 0666
    user = postfix
    group = postfix
  }
}

service auth {
  unix_listener auth-userdb {
    mode = 0600
    user = vmail
```

```
    group = vmail
  }

  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user = postfix
    group = postfix
  }

  user = dovecot
}

service auth-worker {
  user = vmail
}
```

10-ssl.conf editing of

10-ssl.conf editing of

```
$ sudo vim /etc/dovecot/conf.d/10-ssl.conf
```

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

```
ssl_cert = </etc/postfix/ssl/selfsigned/cert.pem
ssl_key = </etc/postfix/ssl/selfsigned/privkey.pem
ssl_dh_parameters_length = 2048
ssl_cipher_list = EECDH+AES:EDH+AES+aRSA
ssl_prefer_server_ciphers = yes
```

Editing of 20-lmtp.conf

Editing of 20-lmtp.conf

```
$ sudo vim /etc/dovecot/conf.d/20-lmtp.conf
```

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

```
protocol lmtp {
  postmaster_address = postmaster@ドメイン名
  mail_plugins = $mail_plugins
}
```

Postfix & Dovecot 起動

```
$ sudo systemctl start postfix.service
$ sudo systemctl start dovecot.service
```

postfixadmin

Here is the version 2.93, but please warm download the latest version.

```
$ wget https://sourceforge.net/projects/postfixadmin/files/postfixadmin/postfixadmin-2.93/postfixadmin-2.93.tar.gz/download
$ mv ./download postfixadmin-2.93.tar.gz
$ tar xfvz postfixadmin-2.93.tar.gz
$ mv ./postfixadmin-2.93 /home/webadmin/www/public_html/postfixadmin
$ sudo mv /home/webadmin/www/public_html/postfixadmin/ADDITIONS/*.sh /var/vmail/bin
$ rm -R /home/webadmin/www/public_html/postfixadmin/ADDITIONS
$ sudo chown -R vmail:vmail /var/vmail/bin/
$ sudo sh -c "chmod 0700 /var/vmail/bin/*.sh"
```

Edit the config.inc.php

/home/webadmin/www/public_html/postfixadmin/config.inc.php

```
$CONF['configured'] = true;
$CONF['default_language'] = 'ja';
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'postfix';
$CONF['database_password'] = 'postfixパスワード';
$CONF['database_name'] = 'postfix';
```

```
$CONF['encrypt'] = 'dovecot:SHA512-CRYPT';
$CONF['dovecotpw'] = "doveadm pw -s SHA512-CRYPT";

$CONF['domain_path'] = 'YES';
$CONF['domain_in_mailbox'] = 'NO';

$CONF['mailbox_postcreation_script']='sudo -u vmail /var/vmail/bin/postfixadmin-mailbox-postcreation.sh';
$CONF['mailbox_postdeletion_script']='sudo -u vmail /var/vmail/bin/postfixadmin-mailbox-postdeletion.sh';
$CONF['domain_postdeletion_script'] ='sudo -u vmail /var/vmail/bin/postfixadmin-domain-postdeletion.sh';
```

The domain directory and mail box directory is created and deleted automatically, it supports in the presence of the shell.

Editing of postfixadmin-mailbox-postcreation.sh

/ Var / vmail / vhosts / domain name /, sur to create a mailbox directory.

```
$ sudo vim /var/vmail/bin/postfixadmin-mailbox-postcreation.sh
```

/var/vmail/bin/postfixadmin-mailbox-postcreation.sh

```
basedir=/var/vmail/vhosts
mkdir -p -m 0700 "${parent}"
#maildirmake "$maildir"
mkdir -p -m 0700 $maildir/{new,cur,tmp,.Drafts,.Junk,.Trash,.Sent}
```

postfixadmin-mailbox-postdeletion.sh の 編 集

/ Var / vmail / vhosts / domain name /, sur to remove the mailbox directory of.

```
$ sudo vim /var/vmail/bin/postfixadmin-mailbox-postdeletion.sh
```

/var/vmail/bin/postfixadmin-mailbox-postdeletion.sh

```
basedir=/var/vmail/vhosts
trashbase=/var/vmail/deleted-vhosts
```

postfixadmin-domain-postdeletion.sh の 編 集

/ var / vmail / vhosts sur that you want to delete the directory of domain names in the.

Has been deleted, domain directory, / Var / Vmail / Deleted-Vhosts will be moved with a date to.

```
$ sudo vim /var/vmail/bin/postfixadmin-domain-postdeletion.sh
```

/var/vmail/bin/postfixadmin-domain-postdeletion.sh

```
basedir=/var/vmail/vhosts
trashbase=/var/vmail/deleted-vhosts
```

Set building and later, so come out a lot with guggul, after will leave.

Connection test

Console 1

```
$ sudo tail -f /var/log/maillog
```

It is recommended that you test while looking at the log in two screen. In five of the following tests, especially if there is no problem, Postfix and Dovecot setting is finished.

SMTP port:25

Console 2

```
$ telnet ドメイン名 25
Trying 52.196.8.213...
Connected to ドメイン名.
Escape character is '^]'.
220 ドメイン名 ESMTP Postfix
ehlo localhost
... 省略
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

SMTP SSL/TLS port:465

Console 2

```
$ Openssl S_client -Connect domain name: 465 -Tlsextdebug
 CONNECTED (00000003)
... omitted
220 domain name Postfix ESMTP
Quit
221 2.0.0 Bye
Closed
```

SMTP STARTTLS port:587

Console 2

```
$ openssl s_client -connect ドメイン名:587 -starttls smtp -tlsextdebug
CONNECTED(00000003)
```

```
... 省略
250 DSN
quit
221 2.0.0 Bye
closed
```

POP3 STARTTLS port:110

Console 2

```
$ openssl s_client -connect ドメイン名:110 -starttls pop3 -tlsextdebug
CONNECTED(00000003)
... 省略
---
+OK Dovecot ready.
quit
closed
```

POP3 SSL/TLS pop3:995

Console 2

```
$ openssl s_client -connect ドメイン名:995 -tlsextdebug
CONNECTED(00000003)
... 省略
+OK Dovecot ready.
quit
closed
```

Finally

Now you can send and receive safe mail.

https:. // admin domain name / postfixadmin

mail software Thunderbird was using, was effortlessly able to send and receive e-mail in the automatic setting.

It will be at the end or more.

SOGo: installation on CentOS-MariaDB-Postfix-Dovecot-PostfixAdmin

https://wiki.kogite.fr/index.php/SOGo_:_installation_sur_CentOS-MariaDB-Postfix-Dovecot-PostfixAdmin

Summary [hide]

4.1 Installation
4.2 Configuration
4.2.1 Sogo
4.2.2 Apache
4.2.3 [TEMP] Add sender aliases to an account
System

install a minimal Centos6.x then:
yum update
yum install screen wget bash-completion man
Disable iptables and selinux
service iptables stop
Service ip6tables stop
chkconfig --del iptables
chkconfig --del ip6tables
setenforce 0
and in / etc / sysconfig / selinux edit:
SELINUX = disabled
Install MariaDB Replace_MySQL_par_MariaDB_sur_CentOS6
Declare additional deposits Dépôts_complémentaires
Install the necessary packages:
yum install --enablerepo = remi php-mysql php-mbstring php-imap
yum install postfix dovecot dovecot-mysql dovecot-pigeonhole cyrus-sasl-devel cyrus-sasl-sql
Postfix - PostfixAdmin

Check the latest version of PostfixAdmin at http://sourceforge.net/projects/postfixadmin/files/
wget http: // sourceforge.net / projects / postfixadmin / files / latest / download -O postfixadmin- 2.92 .tar.gz
tar -xvf postfixadmin- 2.92 .tar.gz
mv postfixadmin- 2.92 / usr / share / postfixadmin
chown -R apache.apache / usr / share / postfixadmin /
For vacation.pl:
yum install --enablerepoPerl-Perl-MailTools Perl-Perl-Mail-Perl-Perl-Perl-Perl-Perl-Perl-Perl-Perl
Configure the mail store (mailbox directory)
mkdir / home / vmail
chmod 770 / home / vmail
useradd -r -u 101 -g email -d / home / vmail -s / sbin / nologin -c "virtual mailbox" vmail

chown vmail: email / home / vmail
Setting Apache
vi / etc / httalias / mailadmin / usr / share / postfixadmin
< Directory> / usr / share / postfixadmin " >
  AllowOverride AuthConfig
</ Directory > pd / conf.d / postfixadmin.conf
httpd restart service
Creating the database
mysql -u root -p # Default root has no password
MariaDB> CREATE DATABASE postfix;
MariaDB> CREATE USER postfix @ localhost IDENTIFIED BY 'mysql_postfix_password';
MariaDB> GRANT ALL PRIVILEGES ON postfix. * TO postfix @ localhost;
Configuring Services

Configuring PostfixAdmin

cd / usr / share / postfixadmin
vi config.inc.php
modify or add:
$ CONF ['configured'] = true;
$ CONF ['setup_password'] = 'changeme'; // will be modified later
$ CONF ['default_language'] = 'fr';
$ CONF ['postfix_admin_url'] = '/ mailadmin';
$ CONF ['database_type'] = 'mysql';
$ CONF ['database_host'] = 'localhost';
$ CONF ['database_user'] = 'postfix';
$ CONF ['database_password'] = 'mysql_postfix_password';
$ CONF ['database_name'] = 'postfix';
$ CONF ['admin_email'] = 'postmaster@domain.fr'; // or empty

// if necessary
$ CONF ['aliases'] = '50';
$ CONF ['mailboxes'] = '50';
$ CONF ['maxquota'] = '100';

$ CONF ['domain_quota'] = 'YES';

```
$ CONF ['quota_multiplier'] = '1024000';
$ CONF ['transport'] = 'YES';
$ CONF ['transport_options'] = array (
  'virtual', // for virtual accounts
  'local', // for system accounts
  'relay' // for backup mx
);
$ CONF ['transport_default'] = 'virtual';
$ CONF ['vacation'] = 'YES';
$ CONF ['sendmail'] = 'NO';

$ CONF [ 'create_mailbox_subdirs'] = array ( 'Drafts', 'Spam', 'Sent', 'Trash');
$ CONF [ 'create_mailbox_subdirs_host'] = 'localhost';
$ CONF ['create_mailbox_subdirs_prefix'] = ;

$ CONF [ 'create_mailbox_subdirs_hostport'] = 143;
// $ CONF ['create_mailbox_subdirs_hostoptions'] = array ('notls');
$ CONF [ 'create_mailbox_subdirs_hostoptions'] = array ( 'novalidate-cert', 'norsh');
//
// END OF CONFIG FILE
//
```

Everything else is left in the file by default. Navigate to http: // <server URL> /mailadmin/setup.php
If all prerequisites are met, enter a password in "setup password" and validate. It gives a hash to enter in config.inc.php.
Then go back to setup.php, enter the password again in setup password, and create an admin (postmaster@domain.fr) + password.
You can then browse http: // <server URL> / mailadmin with this login
Configuring Postfix

vi / etc / postfix / main.cf
in relation to the fault, modify:
inet_interfaces = all
myhostname = sogo.domain.fr
mydomain = domain.fr
mynetworks = $ config_directory / mynetworks
relay_domains = proxy: mysql: /etc/postfix/mysql-relay_domains_maps.cf
recipient_delimiter = +
ADD at the end:

```
transport_maps = hash: / etc / postfix / transport
# virtual setup
virtual_alias_maps = proxy: mysql: /etc/postfix/mysql-virtual_alias_maps.cf,
                proxy: mysql: /etc/postfix/mysql-virtual_alias_alias_maps.cf,
                regexp: / etc / postfix / virtual_regexp
virtual_mailbox_base = / home / vmail
virtual_mailbox_domains = proxy: mysql: /etc/postfix/mysql-virtual_domains_maps.cf
virtual_alias_domains = proxy: mysql: /etc/postfix/mysql-virtual_alias_domains.cf
virtual_mailbox_maps = proxy: mysql: /etc/postfix/mysql-virtual_mailbox_maps.cf,
                 proxy: mysql: /etc/postfix/mysql-virtual_mailbox_alias_maps.cf
virtual_mailbox_limit_maps = proxy: mysql: /etc/postfix/mysql-virtual_mailbox_limit_maps.cf
virtual_minimum_uid = 101
virtual_uid_maps = static: 101
virtual_gid_maps = static: 12
virtual_transport = dovecot
dovecot_destination_recipient_limit = 1
mailbox_size_limit = 5120000000
smtp_host_lookup = dns, native

# authentication - "smtpd_sasl_auth_enable = no" to avoid open relay !!!
smtpd_sasl_auth_enable = no
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $ myhostname
broken_sasl_auth_clients = yes
smtpd_sasl_type = dovecot
smtpd_sasl_path = private / auth

# tls config
smtp_use_tls = yes
smtpd_use_tls = yes
smtpd_tls_security_level = may
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev: / dev / urandom
smtp_tls_session_cache_database = btree: $ data_directory / smtp_tls_session_cache
```

```
# Change mail.example.com. * To your host name
smtpd_tls_key_file = /etc/pki/tls/private/mail.example.com.key
smtpd_tls_cert_file = /etc/pki/tls/certs/mail.example.com.crt
# smtpd_tls_CAfile = /etc/pki/tls/root.crt

# rules restrictions
smtpd_client_restrictions =
smtpd_helo_restrictions =
smtpd_sender_restrictions =
smtpd_recipient_restrictions = permit_sasl_authenticated,
    permit_mynetworks,
    reject_unauth_destination,
    reject_non_fqdn_sender,
    reject_non_fqdn_recipient,
    reject_unknown_recipient_domain
# uncomment for realtime
#, reject_rbl_client zen.spamhaus.org
#, reject_rbl_client bl.spamcop.net
#, reject_rbl_client dnsbl.sorbs.net
smtpd_helo_required = yes
disable_vrfy_command = yes
smtpd_data_restrictions = reject_unauth_pipelining

# Other options
# email size limit ~ 20Meg
message_size_limit = 204800000
Generate TLS keys
Self-signed:
openssl req -new -outform PEM -out smtpd.cert -newkey rsa: 2048 -nodes -keyout smtpd.key -keyform PEM -days 3650 -x509
mv smtpd.cert / etc / pki / tls / certs / mail.example.com .crt
mv smtpd.key / etc / pki / tls / private / mail.example.com.key
OR if official certificate: merge the signed server certificate (crt) with the authority's intermediate certificate (in pem format)
cat mail.example.com.officiel.crt / etc / pki / tls / GandiStandardSSLCA.pem > mail.example.com.pem
Edit Postfix configuration files
vi / etc / postfix / master.cf
in relation to the fault, modify:
```

```
submission inet n - n - - smtpd
  -o smtpd_tls_security_level = encrypt
  -o smtpd_sasl_auth_enable = yes
  -o smtpd_client_restrictions = permit_sasl_authenticated, reject
  -o milter_macro_daemon_name = ORIGINATING
  -o syslog_name = postfix / submission
pickup fifo n - n 60 1 pickup
  -o content_filter =
  -o receive_override_options = no_header_body_checks
maildrop unix - nn - - pipe
  flags = DRhu user = vmail argv = / usr / local / bin / maildrop -d $ {recipient}
uucp unix - nn - - pipe
  flags = Fqhu user = uucp argv = uux -r -n -z -a $ sender - $ nexthop! rmail ($ recipient)
ifmail unix - nn - - pipe
  flags = F user = ftn argv = / usr / lib / ifmail / ifmail -r $ nexthop ($ recipient)
bsmtp unix - nn - - pipe
  flags = Fq. user = bsmtp argv = / usr / local / sbin / bsmtp -f $ sender $ nexthop $ recipient
ADD at the end
127.0.0.1:10025 inet n - y - - smtpd
  -o content_filter =
  -o smtpd_helo_restrictions =
  -o smtpd_sender_restrictions =
  -o smtpd_recipient_restrictions = permit_mynetworks, reject
  -o mynetworks = 127.0.0.0 / 8
  -o smtpd_error_sleep_time = 0
  -o smtpd_soft_error_limit = 1001
  -o smtpd_hard_error_limit = 1000
  -o receive_override_options = no_header_body_checks
  -o smtpd_bind_address = 127.0.0.1
  -o smtpd_helo_required = no
  -o smtpd_client_restrictions =
  -o smtpd_restriction_classes =
  -o disable_vrfy_command = no
  -o strict_rfc821_envelopes = yes
#
# Dovecot LDA (with management of recipient_delimiter)
```

dovecot unix - nn - - pipe
  flags = DRhu user = vmail: mail argv = / usr / libexec / dovecot / deliver -f {sender} -a $ {recipient} -d $ {user} @ $ {nexthop}
#
Vacation mail
vacation unix - nn - - pipe
  flags = Rq user = vacation argv = / var / spool / vacation / vacation.pl -f $ {sender} - $ {recipient}
Possibly, if amavis is installed:
smtp-amavis unix - - y - 2 smtp
  -o smtp_data_done_timeout = 1200
  -o disable_dns_lookups = yes
  -o smtp_send_xforward_command = yes
vi / etc / postfix / mynetworks
This specifies the list of subnets that Postfix considers as
"trusted" SMTP clients that have more privileges than "strangers".
#
In particular, "trusted" SMTP clients are allowed to relay mail
through Postfix.
#
Be sure to add your public ip address block if needed.
#
192.168.0.0/16
10.0.0.0/8
127.0.0.0/8
vi / etc / postfix / mysql-virtual_alias_maps.cf
< syntaxhighlight lang = bash >
 hosts = localhost
 user = postfix
 mysql_postfix_password
 dbname = postfix
 query SELECT goto FROM alias WHERE address = '% s' AND active = '1'

< syntaxhighlight lang = bash enclose = "div" >
vi / etc / postfix / mysql-virtual_domains_maps.cf
hosts = localhost
user = postfix
password = mysql_postfix_password

dbname = postfix
query = SELECT domain FROM domain WHERE domain = '% s' AND backupmx = '0' AND active = '1'
vi / etc / postfix / mysql-virtual_alias_alias_maps.cf
user = postfix
password = mysql_postfix_passwordmysql_postfix_password
hosts = 127.0.0.1
dbname = postfix
query = SELECT goto FROM alias, alias_domain WHERE alias_domain.alias_domain = '% d' AND alias.address = concat ('% u', '@', alias_domain.target_domain) AND alias.active = 1
vi / etc / postfix / mysql-virtual_alias_domains.cf
user = postfix
password = mysql_postfix_password
hosts = 127.0.0.1
dbname = postfix
query = SELECT alias_domain FROM alias_domain where alias_domain = '% s'
vi / etc / postfix / mysql-relay_domains_maps.cf
hosts = localhost
user = postfix
password = postfix
dbname = postfix
query = SELECT domain FROM domain WHERE domain = '% s' and backupmx = '1'
vi / etc / postfix / mysql-virtual_mailbox_maps.cf
hosts = localhost
user = postfix
password = postfix
dbname = postfix
query = SELECT maildir FROM mailbox WHERE username = '% s' AND active = '1'
vi / etc / postfix / mysql-virtual_mailbox_alias_maps.cf
user = postfix
password = mysql_postfix_password
dbname = postfix
query = SELECT maildir FROM mailbox, alias_domain WHERE alias_domain.alias_domain = '% d' AND mailbox.username = concat ('% u', '@', alias_domain.target_domain) AND mailbox.active = 1
vi / etc / postfix / mysql-virtual_mailbox_limit_maps.cf
hosts = localhost
user = postfix

password = mysql_postfix_password
dbname = postfix
query = SELECT quota FROM mailbox WHERE username = '% s' AND active = '1'
touch / etc / postfix / virtual_regexp
Email Address

useradd -r -d / var / spool / holiday -s / sbin / nologin -c "Virtual vacation" holiday
mkdir / var / spool / holiday
chmod 770 / var / spool / holiday
cp / usr / share / postfixadmin / VIRTUAL_VACATION / holiday .pl / var / spool / vacation /
chmod 755 / var / spool / vacation / vacation.pl
echo "autoreply.domain.fr vacation" > / etc / postfix / transportation
postmap / etc / postfix / transportation
chown -R vacation: vacation / var / spool / vacation
echo " 127.0.0.1 autoreply.domain.fr " >> / etc / hosts
mkdir / etc / postfixadmin# or: mkdir -p / etc / mail / postfixadmin, vacation.pl accepts both paths
vi / etc / postfixadmin / vacation.conf
# ========== begin configuration ==========
$ db_type = 'mysql';
$ db_username = 'postfix';
$ db_password = 'mysql_postfix_password';
$ db_name = 'postfix';
$ vacation_domain = 'autoreply.domain.fr';
$ custom_noreply_pattern = 1;
Custom vacation.pl: address exclusion
07/2015: Added an exlusion function and the corresponding configuration variables (can not be processed by the existing function custom_noreply because it also checks the recipient: therefore ourselves!).
diff /usr/share/postfixadmin/VIRTUAL_VACATION/vacation.pl /var/spool/vacation/vacation.pl
202a203,204
> our $ custom_exclude_pattern = 0;
> our $ exclude_pattern = 'some_address | some_domain';
642a645,655
>
> sub exclude_from_address {
> my ($ address) = @_;
> my $ logger = get_logger ();

>
> if (($ custom_exclude_pattern == 1 && $ address = ~ /^.*($exclude_pattern).*/i)) {
> $ logger-> debug ("sender $ address contains excluded pattern $ 1 - will not send vacation message");
> exit (0);
>
>
>
692a706
> $ from = exclude_from_address ($ from);
And in /etc/postfixadmin/vacation.conf add:
$ custom_exclude_pattern = 1;
$ exclude_pattern = 'exclude_domain.org|exclude_address@domain.org|exclude_sender';
Configuring Dovecot

cp / etc / dovecot / dovecot.conf / etc / dovecot / dovecot.conf.save
vi / etc / dovecot / dovecot.conf
edit :
protocols = imap pop3 lmtp
in the /etc/dovecot/conf.d directory, change:
10-auth.conf
auth_mechanisms = plain login
#! include auth-system.conf.ext
! include auth-sql.conf.ext
10-mail.conf
mail_plugins = quota imap_quota trash
mail_location = maildir: / home / vmail /% d /% n
first_valid_uid = 101
first_valid_gid = 12
mailbox_idle_check_interval = 30 secs
maildir_copy_with_hardlinks = yes
add :
plugin {
  trash = /etc/dovecot/trash.conf
}
and to enable folder sharing:
namespace {

```
 type = private
 separator = /
 prefix =
 #location defaults to mail_location.
 inbox = yes
}
namespace {
 type = shared
 separator = /
 prefix = Shared / %% n /
 #location = maildir: %% h / Maildir: INDEX = ~ / Maildir / shared / %% u
 location = maildir: / home / vmail / %% d / %% u: INDEX = / home / vmail /% d /% u / shared / %% u
 subscriptions = no
 list = children
}
```

NOTE : Mailbox sharing to mailbox (represented by the "type = shared" namespace) does not work at this time. Either this is not the proper configuration of namespace, or we have a problem with version 2.0.9-8 of dovecot on CentOS6. We continue to look for ...

In the meantime, public folders can be activated, which advantageously replace user-to-user shares:

add in addition of the 2 namespaces above:

```
namespace {
 type = public
 separator = /
 prefix = Public /
 location = maildir: / home / vmail / public
 subscriptions = no
}
```

Create / home / vmail / public directory, and in a dovecot-acl file with:

```
anyone itlr
```

In this directory "public" create subdirectories (one can not deposit mails directly in "public"). Change the owner of all directories and this file to "vmail: mail". Restart dovecot. A virtual folder "Shared Folders" should appear in mail clients (think about checking if this client needs to subscribe to the folders to view them).

10-master.conf # uncommenting:

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

```
  port = 993
  ssl = yes
}
service pop3-login {
  inet_listener pop3 {
    port = 110
}
inet_listener pop3s {
    port = 995
    ssl = yes
}
service imap {
  vsz_limit = 256M
}
service pop3 {
}
service auth {
  unix_listener auth-userdb {
    mode = 0666
   user = vmail
   group = mail
}
  unix_listener / var / spool / postfix / private / auth {
    mode = 0666
    user = postfix
    group = postfix
  }
}
service auth-worker {
}
service dict {
  unix_listener dict {
    mode = 0666
    user = vmail
    group = mail
  }
```

```
}
10-ssl.conf
ssl = yes
ssl_verify_client_cert = no
15-lda.conf
lda_mailbox_autocreate = yes
lda_mailbox_autosubscribe = yes
protocol lda {
  mail_plugins = sieve acl
  postmaster_address = postmaster@your-domain.tld
}
20-imap.conf
protocol imap {
  mail_plugins = $ mail_plugins acl imap_acl
  imap_client_workarounds = delay-newmail
}
20-lmtp.conf
lmtp_save_to_detail_mailbox = yes
protocol lmtp {
  mail_plugins = $ mail_plugins
}
20-managesieve.conf # uncomment:
service managesieve-login {
    inet_listener sieve {
    port = 4190
}
service_count = 1
    process_min_avail = 0
    vsz_limit = 64M
}
service managesieve {
}
protocol sieve {
   managesieve_max_line_length = 65536
   managesieve_implementation_string = Dovecot Pigeonhole
   managesieve_max_compile_errors = 5
```

}
20-pop3.conf
protocol pop3 {
  mail_plugins = $ mail_plugins
  pop3_client_workarounds = outlook-no-nuls oe-ns-eoh
}
90-acl.conf
plugin {
  acl = vfile
}
90-quota.conf (we manage the quotas to the user, with the "messages" and "bytes" fields of the database)
dict {
  quotadict = mysql: /etc/dovecot/dovecot-dict-quota.conf
}
plugin {
  quota = dict: user :: proxy :: quotadict
}
90-sieve.conf (management of mail files)
plugin {
  sieve = ~ / .dovecot.sieve
  sieve_global_path = /var/lib/dovecot/sieve/default.sieve
  sieve_dir = ~ / sieve
  sieve_global_dir = / var / lib / dovecot / sieve /
  sieve_max_script_size = 1M
}
Create files:
vi / etc / dovecot / dovecot-sql.conf.ext
driver = mysql
connect = host = localhost dbname = postfix user = postfix password = mysql_postfix_password
default_pass_scheme = MD5-CRYPT

password_query = SELECT username as user, password, concat ('/ home / vmail /', maildir) as userdb_home, concat ('maildir: / home / vmail /', maildir) as userdb_mail,
WHERE username = '% u' AND active = '1'

user_query = SELECT concat ('/ home / vmail /', maildir) as home, concat ('maildir: / home / vmail / ', quota) as quota_rule FROM mailbox WHERE username ='% u 'AND
active =' 1 '

vi / etc / dovecot / trash.conf
1 Spam
Uncomment if you want trash as well
2 Trash
vi / etc / dovecot / dovecot-dict-quota.conf
connect = host = localhost dbname = postfix user = postfix password = mysql_postfix_password
map {
  pattern = priv / quota / storage
  table = quota2
  username_field = username
  value_field = bytes
}
map {
  pattern = priv / quota / messages
  table = quota2
  username_field = username
  value_field = messages
}

NOTE: WITH THIS CONFIGURATION THE SHARING OF MAIL BOXES DOES NOT WORK! TO BE DEBUGGED.
Configuration Sieve

mkdir -p / var / lib / dovecot / sieve /
vi / var / lib / dovecot / sieve / globalfilter.sieve
require "fileinto";
  if exists "X-Spam-Flag" {
        if header: contains "X-Spam-Flag" "NO" {
        } else {
        fileinto "Spam";
        stop;
        }
  }
  if header: contains "subject" ["*** SPAM ***"] {
    fileinto "Spam";
    stop;
  }

touch / var / lib / dovecot / sieve / default.sieve
chmod 754 / var / lib / dovecot / sieve / default.sieve
chown -R vmail: Email / var / lib / dovecot
chkconfig dovecot is
the service postfix restart
the service dovecot restart
The other filters are saved in a file in ~ / sieve / xxxxx and a ~ / .dovecot.sieve link points to the file in that directory to activate it. (so there may be several filter files in the ~ / sieve directory, activated as needed).
The filters can be modified by hand, or by Thunderbird by loading the module "Sieve": https://github.com/thsmi/sieve/tree/master/nightly (at the bottom of the page, save the target of the link ... on sieve-0.2.3f.xpi, and install in TH modules) In message filters> Sieve settings, indicate:
sogo.domain.fr
port 4190
IMAP account
secure connection TRUE
Then in Message Filters> Sieve Filters Create a script (one asset at a time) with the rules.
End of configuration

Put the IP and the host name in / etc / hosts
In Postfixadmin, create a domain, and accounts.
SOGo

Installation

vi / etc / yum.repos.d / sogo.repo
[SOGo]
name = Inverse SOGo Repository
baseurl = http://inverse.ca/downloads/SOGo/RHEL6/$basearch
gpgcheck = 0
yum install --enablerepo = epo sogo sogo-ealarms-notify sogo-tool sudo memcached sope49-gdl1-mysql mod_ssl
service memcached start
chkconfig memcached on
Configuration

Create a MariaDB view on Postfix tables for SOGo
mysql -u root -p postfix
ASD_LOCAL_CONTROL AS_ELECT AS_ELECT AS_ELECT AS_ELECT AS_ELECT AS c_uid, username AS c_name, PASSWORD AS c_password, name AS c_cn, username AS mail,

domain FROM `mailbox`;
The rest of the tables will be created by SOGo.
Sogo
Inside the first embrace "{", uncomment and modify / add:
SOGoTimeZone = "Europe / Paris";
SOGoLanguage = French;
// SOGoMailMessageCheck = every_10_minutes;
// SOGoFirstDayOfWeek = 1;
// SOGoMailComposeMessageType = html;

# Obligatoitre, otherwise user shares to user do not work:
SOGoCalendarDefaultRoles = (
    PublicDAndTViewer
  // ConfidentialDAndTViewer
);

 SOGoSieveServer = sieve: //127.0.0.1: 4190;

// Send mail at different events (eg change of rights or creation):
 SOGoAppointmentSendEMailNotifications = YES;
 SOGoFoldersSendEMailNotifications = YES;
 SOGoACLsSendEMailNotifications = YES;

// Authorize users to change their password:
 SOGoPasswordChangeEnabled = YES;

// Sending mail to remind you of a task or an event:
 // SOGoEnableEMailAlarms = YES;

// Web Interface //
  SOGoForwardEnabled = YES;

  SOGoMemcachedHost = 127.0.0.1;

// These lines are important: they are the ones that force Sogo to create the corresponding tables
SOGoProfileURL = "mysql: // sogo: secret@127.0.0.1: 3306 / sogo / sogo_user_profile";

```
OCSFolderInfoURL = "mysql: // sogo: secret@127.0.0.1: 3306 / sogo / sogo_folder_info";
OCSSessionsFolderURL = "mysql: // sogo: secret@127.0.0.1: 3306 / sogo / sogo_sessions_folder";

// Authorize the connection in user@domain.tld:
SOGoEnableDomainBasedUID = YES;

// User Authentication - on a Postfix database - mysql
domains = {
    "domain.fr" =
        {
         SOGoMailDomain = "domain.com";
         SOGoDraftsFolderName = Drafts;
         SOGoUserSources = (
              {canAuthenticate = YES;
              displayName = "Domain Users";
              id = users1;
              isAddressBook = YES;
              type = sql;
              userPasswordAlgorithm = md5;
              LoginFieldNames = ("c_name");
              viewURL = "mysql: // sogo: secret@127.0.0.1: 3306 / sogo / sogo_users";
              IMAPLoginFieldName = "mail";
              DomainFieldName = "domain"; });
         };
    };
```

Create an "admin" admin, admin Sogo (SOGoSuperUsernames), which will share mail folders, agenda and global address book.
Apache
The sogo RPM created a file /etc/httpd/conf.d/SOGo.conf (example file where everything is commented). Edit "yourhostname" with the desired vhost name.
This file intercepts a URL of the type " https://myserver.domain.com/SOGo ".
Problem, with this configuration, if one accesses " https://myserver.domain.com ", one falls on the default page of Apache.
To redirect to SOGo, on CentOS edit etc / httpd / conf.d / welcome.conf:

```
<LocationMatch "^ / + $">
  Options -Indexes
# ErrorDocument 403 /error/noindex.html
  Redirect / https://sogo.systea.fr/SOGo
```

</ LocationMatch>
Then:
service httpd restart
service sogod restart
[TEMP] Add sending aliases to an account
Temporary solution until Sogo manages it himself. Edit the "mailbox" table:
mysql -u root -p postfix
MariaDB> alter table mailbox add mail_alias1 varchar (255) after username;
MariaDB> alter table mailbox add mail_alias2 varchar (255) after mail_alias1;
MariaDB> alter table mailbox add mail_alias3 varchar (255) after mail_alias2;
MariaDB> alter table mailbox add mail_alias4 varchar (255) after mail_alias3;
Change view:
MariaDB> DROP VIEW `sogo_users`;
Cdn, cd, cd, cd, cd, cd, cd, cd, cd, cd, cd, cd, cd, cd, cd, ;
In /etc/sogo/sogo.conf, add the line:
MailFieldNames = ("mail1", "mail2", "mail3", "mail4");
between "LoginFieldNames" and "viewURL" of the "domain" block (see Sogo configuration above).
Then, manually add the aliases:
MariaDB> update sogo_users set mail1 = 'first_alias@domain.com' where c_uid = 'user_name';
and restart Postfix to re-read his tables.
