Setup basic mailserver with Postfix + Dovecot + Sieve for Virtualmail

.. on ubunut 18.04 server

pastebin…

source

very good guide that explains alot

```
apt install pwgen
# usefull to create safe passwords
apt install mariadb-server
# we need this to store our accounts and domains database
apt install postfix postfix-mysql
```

choose to configure postfix as **Internet Site** and enter fully qualified domain name of the server, the name entered here must not match any email domains you want to handle later (since we are configuring our server to process virtual mailboxes and not canonical domains).

```
apt install apache2 php7.2 swaks mutt certbot dovecot-mysql dovecot-pop3d
dovecot-imapd dovecot-managesieved dovecot-lmtpd adminer ca-certificates

rm /etc/apache2/sites-enabled/*

cat > /etc/apache2/sites-available/webmail-http.conf <<EOF
<VirtualHost *:80>
  ServerName mail.yourdomain.ch
  DocumentRoot /var/www/webmail/pub
</VirtualHost>
EOF
a2ensite webmail-http
systemctl reload apache2
mkdir -p /var/www/webmail/pub
chown www-data.www-data /var/www/webmail/pub
echo "it works" > /var/www/webmail/pub/test.txt
```

now try http://mail.yourdomain.ch/test.txt in a web-browser, it should show “it works!”

```
certbot certonly --webroot --webroot-path /var/www/webmail/pub -d
mail.yourdomain.ch
cat > /etc/apache2/sites-available/webmail-https.conf <<EOF
<VirtualHost *:443>
 ServerName mail.yourdomain.ch
 DocumentRoot /var/www/webmail/pub
 SSLEngine on
 SSLCertificateFile /etc/letsencrypt/live/mail.yourdomain.ch/fullchain.pem
 SSLCertificateKeyFile /etc/letsencrypt/live/mail.yourdomain.ch/privkey.pem
 Alias /adminer /usr/share/adminer/adminer
</VirtualHost>
EOF
a2ensite webmail-https
a2enmod ssl
systemctl restart apache2
```

to auto-forward all non-https traffic except for the certbot renewal stuff to https, add this to your `/etc/apache2/sites-available/webmail-http.conf` file before the closing `VirtualHost` tag:

```
RewriteEngine On
RewriteCond %{REQUEST_URI} !.well-known/acme-challenge
RewriteRule ^(.*)$ https://%{SERVER_NAME}/ [R=301,L]
```

```
a2enmod rewrite
systemctl restart apache2
```

to make sure letsencrypt will restart all our servers once the ssh keys change, we need to add this:

```
echo -e "\npost-hook = service postfix restart ; service dovecot restart ; service apache2 restart" >> /etc/letsencrypt/cli.ini
```

setup mysql database:

create pw:

```
pwgen -s1 30 2
```

note down the two passwords and start the mysql client

```
mysql
```

in the mysql client console enter the following mysql commands:

```
  CREATE DATABASE mailserver;
  grant all on mailserver.* to 'mailadmin'@'localhost' identified by '<first password goes here>';
  grant select on mailserver.* to 'mailserver'@'127.0.0.1' identified by '<second password goes here>';
  USE mailserver;
  CREATE TABLE IF NOT EXISTS `domains` (
 `id` int(11) NOT NULL auto_increment,
 `name` varchar(50) NOT NULL,
 PRIMARY KEY (`id`)
 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

 CREATE TABLE IF NOT EXISTS `users` (
 `id` int(11) NOT NULL auto_increment,
 `domain_id` int(11) NOT NULL,
 `email` varchar(100) NOT NULL,
 `password` varchar(150) NOT NULL,
 `quota` int(11) NOT NULL DEFAULT 0,
 PRIMARY KEY (`id`),
 UNIQUE KEY `email` (`email`),
 FOREIGN KEY (domain_id) REFERENCES domains(id) ON DELETE CASCADE
 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

 CREATE TABLE IF NOT EXISTS `aliases` (
 `id` int(11) NOT NULL auto_increment,
 `domain_id` int(11) NOT NULL,
 `source` varchar(100) NOT NULL,
 `destination` varchar(100) NOT NULL,
 PRIMARY KEY (`id`),
 FOREIGN KEY (domain_id) REFERENCES domains(id) ON DELETE CASCADE
 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

exit
```

create a password hash and add a test user (using adminer). to create the hash i used

```
dovecot pw -s SHA256-CRYPT
```

now let's configure postfix:

```
cat > /etc/postfix/mysql-virtual-mailbox-domains.cf <<EOF
user = mailserver
password = <second password goes here>
hosts = 127.0.0.1
dbname = mailserver
query = SELECT 1 FROM domains WHERE name='%s'
EOF

postconf virtual_mailbox_domains=mysql:/etc/postfix/mysql-virtual-mailbox-domains.cf

#postconf virtual_mailbox_base=/var/vmail
#postconf virtual_uid_maps=static:5000
#postconf virtual_gid_maps=static:5000
```

postconf adds stuff to your main.cf file and also reloads postfix, so we can now test with our test domain if it works:

```
postmap -q yourdomain.ch mysql:/etc/postfix/mysql-virtual-mailbox-domains.cf
```

this shouwld now show 1 as an answer

now add virtual mailboxes:

```
cat > /etc/postfix/mysql-virtual-mailbox-maps.cf <<EOF
user = mailserver
password = <second password goes here>
hosts = 127.0.0.1
dbname = mailserver
query = SELECT 1 FROM users WHERE email='%s'
EOF

postconf virtual_mailbox_maps=mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf

postmap -q john@yourdomain.ch mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf


cat > /etc/postfix/mysql-virtual-alias-maps.cf <<EOF
user = mailserver
password = <second password goes here>
hosts = 127.0.0.1
dbname = mailserver
query = SELECT destination FROM aliases WHERE source='%s'
EOF

postconf virtual_alias_maps=mysql:/etc/postfix/mysql-virtual-alias-maps.cf

postmap -q forwarded@mydomain.ch mysql:/etc/postfix/mysql-virtual-alias-maps.cf
#this should return the target email address
```

next up is dovecot

create a user who will own all the mail stored on your server:

```
groupadd -g 5000 vmail
useradd -g vmail -u 5000 vmail -d /var/vmail -m
```

all the configs are stored in `/etc/dovecot/conf.d`

edit `10-auth.conf` and make sure that:

```
auth_mechanisms = plain login
```

and that the sql backend at the end of this file is the only enabled backend:

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

next is `auth-sql.conf.ext`

make sure the userdb section looks like this (it already did on ubuntu)

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

next is `10-mail.conf` set the mail location to

```
mail_location = maildir:~/Maildir
```

and enable the quota plugin

```
mail_plugins = quota
```

in the file `10-master.conf` edit the `service auth` section and uncomment the `Postfix smtp-auth` settings.. they should look like this:

```
  # Postfix smtp-auth
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user = postfix
    group = postfix
  }
```

further more, find the `service lmtp` section and edit it so it looks like this:

```
service lmtp {
  unix_listener /var/spool/postfix/private/dovecot-lmtp {
    group = postfix
    mode = 0600
    user = postfix
  }
}
```

restart dovecot

```
systemctl restart dovecot
```

next up is `10-ssl.conf`

```
ssl = required
```

and

```
ssl_cert = </etc/letseXwiN2Np9MI2DyEaBSHDAYPLTQsQkHAncrypt/live/mail.yourdomain.ch/fullchain.pem
ssl_key = </etc/letsencrypt/live/mail.yourdomain.ch/privkey.pem
```

```
cat >> /etc/dovecot/dovecot-sql.conf.ext <<EOF
driver = mysql
```

```
connect = host=127.0.0.1 dbname=mailserver user=mailserver password=<second password goes here>
user_query = SELECT email as user, \\
  concat('*:bytes=', quota) AS quota_rule, \\
  '/var/vmail/%d/%n' AS home, \\
  5000 AS uid, 5000 AS gid \\
  FROM users WHERE email='%u';
password_query = SELECT password FROM users WHERE email='%u'
EOF
```

```
cat >> /etc/dovecot/conf.d/90-quota.conf <<EOF

plugin {
  quota = maildir:User quota

  quota_status_success = DUNNO
  quota_status_nouser = DUNNO
  quota_status_overquota = "552 5.2.2 Mailbox is full and cannot receive any more emails"
}

service quota-status {
  executable = /usr/lib/dovecot/quota-status -p postfix
  unix_listener /var/spool/postfix/private/quota-status {
    user = postfix
  }
}

plugin {
   quota_warning = storage=95%% quota-warning 95 %u
   quota_warning2 = storage=80%% quota-warning 80 %u
   quota_warning3 = -storage=100%% quota-warning below %u
}
service quota-warning {
   executable = script /usr/local/bin/quota-warning.sh
   unix_listener quota-warning {
     group = dovecot
     mode = 0660
   }
 }
EOF
```

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

```
#!/bin/sh
PERCENT=$1
USER=$2
cat << EOF | /usr/lib/dovecot/dovecot-lda -d $USER -o "plugin/quota=maildir:User quota:noenforcing"
```

```
From: postmaster@webmail.example.org
Subject: Quota warning - $PERCENT% reached

Your mailbox can only store a limited amount of emails.
Currently it is $PERCENT% full. If you reach 100% then
new emails cannot be stored. Thanks for your understanding.
EOF
```

```
chmod 755 /usr/local/bin/quota-warning.sh

systemctl restart dovecot.service

postconf "smtpd_recipient_restrictions = reject_unauth_destination
check_policy_service unix:private/quota-status"

chown root:root /etc/dovecot/dovecot-sql.conf.ext
chmod go= /etc/dovecot/dovecot-sql.conf.ext
```

tell postfix to send email via lmtp to dovecot:

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

enable sieve plugin on lmtp protocol (this is where we get mails from postfix, so it's where we want to route them through sieve rules)

edit `/etc/dovecot/conf.d/20-lmtp.conf` and look for `mail_plugins` in the `protocol lmtp` section (at the very end in the default config).. edit the line like this:

```
mail_plugins = $mail_plugins sieve
```

pending issue i had to allow the world to read and write to `/var/run/dovecot/quota-warning` because the vmail user was used to access it and adding the vmail user to the dovecot group did not help to solve the issue.. not sure if this is really a good idea, i hope for some good feedback on this later on, so far i did:

```
chmod a+w /var/run/dovecot/quota-warning
chmod a+r /var/run/dovecot/quota-warning
```

now we can do some testing with swaks to send emails and `mutt` to read them over imap:

```
swaks --to myuser@yourdomain.ch --server mail.yourdomain.ch
mutt -f imaps://myuser@yourdomain.ch@mail.yourdomain.ch
```

install roundcube

i went with the latest roundcube because the versioni that came with ubuntu at the time of writing did not have the new elastic gui which i really wanted.

download the lastest verison of Roundcube and save the tar file to a temporary directory on your server

```
cd /tmp
wget
https://github.com/roundcube/roundcubemail/releases/download/1.4.1/roundcubemail-1.4.1-complete.tar.gz
```

```
cd /var/www/webmail
tar xvf /tmp/roundcubemail-1.4.1-complete.tar.gz
mv roundcubemail-1.4.1/{.,}* pub/
rmdir roundcubemail-1.4.1/
cd pub
```

install dependencies according to INSTALL instructions

```
apt install php7.2-json php7.2-xml php7.2-mbstring php7.2-zip php7.2-intl
php7.2-gd
```

further more i had to set the `date.timezone = Europe/Zurich` in the php.ini and reload apache to apply the changes

follow the INSTALL instructions.. for version 1.4.1 i did this:

```
  chown -R www-data.www-data logs temp
  pwgen -s1 30
```

take note of the password created with pwgen and start the mysql client

```
mysql
```

inside the mysql console run these commands to create the database:

```
CREATE DATABASE roundcubemail /*!40101 CHARACTER SET utf8 COLLATE
utf8_general_ci */;
GRANT ALL PRIVILEGES ON roundcubemail.* TO roundcube@localhost
    IDENTIFIED BY '<password goes here>';
quit
```

```
mysql roundcubemail < SQL/mysql.initial.sql
```

continue in the web-based installer found at https://mail.yourdomain.ch/installer

here are the things i've changed:

- imap port changed to 993
- imap server set to

  ```
  ssl://mail.yourdomain.ch
  ```

- smtp server set to

  ```
  tls://mail.yourdomain.ch
  ```

- set database password to the one you noted down before
- add at least `managesieve` and `password` plugin

move install directory out of www root.. we can always symlink to it later if we need the installer again

```
cd /var/www/webmail/pub/
mv installer ../
```

configure the plugins:

create a new password for a db admin user which we need for the password plugin:

```
pwgen -s1 30
```

create the db admin user for roundcube:

```
mysql
```

in the mysql console add the user:

```
grant all on mailserver.* to 'rcpasswd'@'localhost' identified by '<password
goes here>';
quit
```

now write our config for the password plugin:

```
cat > /var/www/webmail/pub/plugins/password/config.inc.php <<EOF
<?php
\$config['password_driver'] = 'sql';
\$config['password_confirm_current'] = true;
\$config['password_minimum_length'] = 7;
\$config['password_force_save'] = false;
\$config['password_algorithm'] = 'dovecot';
\$config['password_dovecotpw'] = '/usr/bin/doveadm pw -s SHA256-CRYPT';
\$config['password_dovecotpw_method'] = 'SHA256-CRYPT';
\$config['password_dovecotpw_with_method'] = true;
\$config['password_db_dsn'] = 'mysql://rcpasswd:<password goes
here>@localhost/mailserver';
\$config['password_query'] = "UPDATE users SET password=%D WHERE email=%u";
?>
EOF
```

the managesieve plugin is simpler, just provide the hostname of the sieve server and we're golden:

```
cat > /var/www/webmail/pub/plugins/managesieve/config.inc.php <<EOF
<?php
\$config['managesieve_host'] = 'localhost';
```

```
?>
EOF
```

client auto configuration

postponed.. tutorial

additional postfix settings

mail size limit

i think 10MB is just too small, so i allowed 30mb instead

```
postconf message_size_limit=31457280
```

regex based virtual aliases

i've added this as well to my config. however, I called the config file /etc/postfix/regex_aliases.map so i can create mappings for different domains and different purposes.

proxmox mail gateway

From:
http://wiki.psuter.ch/ - **pswiki**

Permanent link:
http://wiki.psuter.ch/doku.php?id=setup_basic_mailserver_with_postfix_dovecot_sieve&rev=1577918444

Last update: **01.01.2020 23:40**