A Mailserver on Ubuntu 16.04: Postfix, Dovecot, I

By Reason
May 7th, 2016
Permalink

This long post contains a recipe for building a reasonably secure Ubuntu 1
Services, using Postfix, Dovecot, and MySQL, with anti-spam packages in
AntiVirus, SpamAssassin, and Postgrey. Local users are virtual ratl
Administration of users and domains is achieved through the Postfix A
provided by Roundcube.

This is an updated version of earlier Ubuntu 12.04 and Ubuntu 14.04 r
people graciously helped to fix bugs and make improvements in the origir
issue here please do let me know.

Introduction

Building a Linux mailserver from scratch to your own liking is a painful p
one of the few folk who do that day in and day out - there is no way
generally consists of a range of different packages that separately handle
mail, and spam-related tasks: they must all talk to one another correctly,
configuration documentation, and there is no one obvious best practice f
to store user data, or how to glue the various different components tog
different viable setups for moving mail between Postfix and Dovecot, t
assembly tends to be unforgiving on matters such as file ownership and
specific processes, and tiny errors in esoteric configuration files. Unless y
end result will likely be either insecure or otherwise subtly non-functiona
the best of bad outcomes.

You must also pay attention to adding support for the dominant tools
suppression industry, such as Sender Policy Framework (SPF) and Domai
else risk mail from your server being flagged for the spam folder or in
providers. This is a broad subject and a changing ecosystem worthy of a l

There are a number of fairly up to date recipes for creating mailservers c
is an Ubuntu recipe by Ivar Abrahamsen, which gives you Postfix for SM
to store account information, virtual user mail directories, and an array o
effective when working in concert. It's a good set of documents, as the
producing a secure mailserver as the end result. In the past I have made
as a basis for my mail servers, and recommend it.

There are also a great many partial recipes and out of date guides that
than a help - especially when it comes to Dovecot, which has changed
versions. The configuration is completely different, and so are many
binaries. When I chose to migrate my servers from Courier to Dovecot ba
find a good all-in-one-place guide, and hence the existence of the first ir
then I've added new versions as Ubuntu updates its long term support rel

You should at least skim the whole of this post before setting forth to follo
avoid unpleasant surprises, and there are some notes at the end on alter
are worth reading before you get started.

Outlining the Goal

The end result of following this guide will be a secure mail server for
following software packages:

- Postfix: sends and receives mail via the SMTP protocol. It will c
mailservers if the mail is sent by an authenticated user, but anyone
for local delivery.
- Dovecot: a POP and IMAP server that manages local mail directorie
and download their mail. It also handles user authentication.
- Postgrey: greylists incoming mail, requiring unfamiliar deliverers tc
resend. This is one of the better tools for cutting down on spam.
- amavisd-new: a manager for organizing various antivirus and spam cl
- Clam AntiVirus: a virus detection suite.
- SpamAssassin: for sniffing out spam in emails.
- Postfix Admin: a web front end for administering mail users and doma
- Roundcube: a webmail interface for users.

The server will accept plain text or encrypted SMTP and POP/IMAP conne
will not allow user authentication without encryption. It will pass through
mail sent by local users, removing identifying information from the origina

Using Amazon Web Services

This post is written assuming the use of Amazon Web Services to host a
any hosting service can be used. Very little of the material here is concer
So if you are working with another service, just skip over the AWS-spec
equivalent operations in the service that you have chosen to use. Not
services are equal when it comes to how email providers view mail comin
it is pointless to even try hosting any significant mail service at Dig
blacklisted by many services, and you will have no leverage with any of th
to trying to ensure delivery.

If not setting up in AWS, at the very least ensure that you firewall you
outset. In many services a new server or virtual instance is completely ex
want to lock it down immediately with something like Uncomplicated F
below, replacing MY_IP_ADDRESS with your IP adddress:

```
sudo su
apt-get install ufw
ufw enable
ufw allow from MY_IP_ADDRESS
```

After you have completed setup then you can open up your server
communicate with the rest of the world.

Use of example.com and mail.example.com

Throughout the instructions in this post example.com is used as the dom
domain has the hostname mail.example.com. So wherever you see thes
with your chosen domain and mail server hostname.

Fire up an Ubuntu 16.04 AWS Instance with a Suitable

Start up a new server instance. At the time of writing, Ubuntu 16.04 is
the quick start menu for launching a new instance. Mail servers don't ge
you aren't in the business of email: 2G of RAM is enough for the recipe
only because ClamAV and Amavis are memory hogs. Thus while micro in
larger instance types should be more than enough to support a personal r
server, or the throughput of a mailing list of a few thousand members.

Firewall settings in AWS are managed through assignment of Security
create one before starting the server. The Security Group should allow
address to these ports:

- 25 (SMTP)
- 80 (HTTP)
- 110 (POP3)
- 143 (IMAP)
- 443 (HTTPS)
- 465 (SMTPS)
- 993 (IMAPS)
- 995 (POP3S)

The above is in addition to whatever rules you might have for SSH acces
idea to leave that open to the world, so lock it down to the IP address ra
idea to restrict all inbound traffic to the server to your own IP addresses
adjust the rules to allow traffic from the rest of the world after you're cert
shipshape.

Some Basic Configuration

The baseline Ubuntu instance is lacking in near every package you migh
fairly close to scratch. You'll log in as the "ubuntu" user and then switch t
"sudo su" command; most of what you need to do requires root access:

```
sudo su
```

You must set up an Elastic IP to give your server a permanent IP address.
have its own strange-looking hostname, so changing to the domain the se
the list.

```
hostname mail.example.com
```

Now set the contents of /etc/hostname to be the hostname:

```
echo "mail.example.com" > /etc/hostname
```

And add your hostname to the first line of /etc/hosts:

```
127.0.0.1 mail.example.com localhost

# There will be IPv6 configuration below the first line, but l
...
```

Now you will want to regenerate the server's default self-signed SSL c
domain name. You may have purchased an SSL certificate for your mail s
and completely secure to run a mail server using a self-signed certificate
warning screens when using webmail hosted on the server and warnin
connecting via POP, IMAP, or SMTP.

```
apt-get install --assume-yes ssl-cert
make-ssl-cert generate-default-snakeoil --force-overwrite
```

Now Build a LAMP Web Server

The list of goals here includes webmail and a web-based administrative
as a starting point you will need to set up a LAMP web server, which st
system, Apache as the webserver, MySQL as the database, and PHP as th
there is a shortcut to install all of the basic LAMP packages, so start by u
installing those packages. Notice the "^" character at the end of the comm

```
apt-get update
apt-get upgrade --assume-yes
apt-get install --assume-yes lamp-server^
```

During this installation process you will be asked to enter and confirm a pa
Choose something sensible and wait for the remaining installations to con
adding an array of necessary or useful additional packages for PHP, such a
parser, and GD image processing support. Note that the old standby o
longer necessary, as this functionality is a part of core PHP now. At this p
your own taste and to support any other applications you might want to ru

```
apt-get install --assume-yes \
  php7.0-mcrypt \
  php7.0-curl \
  php7.0-gd \
  php7.0-mbstring \
  php-apcu \
  php-xml-parser
```

Configure PHP

The default configuration settings for PHP and the additional packag
/etc/php/7.0/apache2/php.ini and /etc/php/7.0/mods-available respec
casual usage. So unless you have something complicated or high-power
change anything.

Use OpenSSL to Create a Unique Diffie-Helman Group

Good security is requiring ever more work on the part of system admin
more recent attacks on SSL is known as Logjam, and defending agains

non-standard addition to application SSL settings. Creating your own
saving it to a configuration file is the first step:

```
openssl dhparam -out /etc/ssl/private/dhparams.pem 2048
chmod 600 /etc/ssl/private/dhparams.pem
```

Configure Apache

The expected end result for the Apache webserver is that it will serve a si
web applications: (a) Roundcube for webmail, and (b) Postfix Admin hi
HTTP requests will be redirected to use HTTPS, as there is no good reas
any of applications that will reside on the server.

Firstly configure the following lines in /etc/apache2/conf-enabled/
information that Apache gives out in its response headers:

```
#
# ServerTokens
# This directive configures what you return as the Server HTT
# Header. The default is 'Full' which sends information about
# and compiled in modules.
# Set to one of:  Full | OS | Minimal | Minor | Major | Prod
# where Full conveys the most information, and Prod the least
#
ServerTokens Prod

#
# Optionally add a line containing the server version and vir
# name to server-generated pages (internal error documents, F
# listings, mod_status and mod_info output etc., but not CGI
# documents or custom error documents).
# Set to "EMail" to also include a mailto: link to the Server
# Set to one of:  On | Off | EMail
#
ServerSignature Off
```

Make sure that mod_rewrite, mod_ssl, a few other useful modules, and
enabled - you'll need these line items to be able to force visitors to use HT

```
a2enmod deflate expires headers rewrite ssl
a2ensite default-ssl
```

Edit these lines in /etc/apache2/mods-available/ssl.conf to ensure that pr
are not used:

```
# Aiming for perfect forward secrecy where possible, and prot
# attacks such as Logjam. See:
# https://weakdh.org/sysadmin.html
# https://hynek.me/articles/hardening-your-web-servers-ssl-ci
SSLCipherSuite ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128
SSLHonorCipherOrder on

#   The protocols to enable.
#   Available values: all, SSLv3, TLSv1, TLSv1.1, TLSv1.2
#   SSL v2  is no longer supported
SSLProtocol all -SSLv2 -SSLv3
```

The default site configuration in /etc/apache2/sites-available/000-defa
something like this for the sake of simplicity:

```
<VirtualHost *:80>
  ServerName mail.example.com
  ServerAdmin webmaster@localhost

  DocumentRoot /var/www/html
  <Directory "/var/www/html">
    Options FollowSymLinks
    AllowOverride All
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/error.log

  # Possible values include: debug, info, notice, warn, error
  # alert, emerg.
  LogLevel warn

  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

But of course your taste and needs may vary. Keeping the same simple ap
SSL configuration in /etc/apache2/sites-available/default-ssl.conf can be s

```
<IfModule mod_ssl.c>
  <VirtualHost _default_:443>
    ServerName mail.example.com
    ServerAdmin webmaster@localhost

    # Set the HTTP Strict Transport Security (HSTS) header to
    # HTTPS for 1 Year, including subdomains, and allow this
    # added to the preload list.
    Header always set Strict-Transport-Security "max-age=3153

    # Prevent clickjacking by controlling who can put the sit
    # frame. Only needed for text/html, but doesn't hurt to b
    # generally.
    Header set X-Frame-Options "SAMEORIGIN"

    # Prevent mime based attacks by telling browsers that sup
    # to use the declared mime type regardless of what the co
    # like.
    Header set X-Content-Type-Options "nosniff"

    DocumentRoot /var/www/html
    <Directory "/var/www/html">
      Options FollowSymLinks
      AllowOverride All
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log

    # Possible values include: debug, info, notice, warn, err
    # alert, emerg.
    LogLevel warn

    CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined

    #   SSL Engine Switch:
    #   Enable/Disable SSL for this virtual host.
    SSLEngine on
    #

    # ... more default SSL configuration ...

    # You will probably need to change this next Directory di
    # in order to match the earlier one.
    <Directory "/var/www/html">
      SSLOptions +StdEnvVars
    </Directory>

    # ... yet more default SSL configuration ...
```

If you are using a purchased rather than self-signed SSL certificate, the
certificate bundle from the issuer. If you don't want to pay for a valid SSL
a look at setting up with Let's Encrypt or a similar free certificate auth
wildcard certificate for *.example.com, a less costly certificate coveri
www.example.com and mail.example.com, or you may have separate ce
you care about. Place the relevant certificate, private key, and CA ce
locations:

- /etc/ssl/private/example.com.key
- /etc/ssl/certs/example.com.crt
- /etc/ssl/certs/ca-bundle.crt

The key must not be password protected, and it must be locked down s
read it:

```
chmod 600 /etc/ssl/private/example.com.key
```

Now change these lines in /etc/apache2/sites-enabled/default-ssl.conf:

```
#   A self-signed (snakeoil) certificate can be created by in
#   the ssl-cert package. See
#   /usr/share/doc/apache2/README.Debian.gz for more info.
#   If both key and certificate are stored in the same file,
#   SSLCertificateFile directive is needed.
SSLCertificateFile    /etc/ssl/certs/example.com.crt
SSLCertificateKeyFile /etc/ssl/private/example.com.key

#   Server Certificate Chain:
#   Point SSLCertificateChainFile at a file containing the
#   concatenation of PEM encoded CA certificates which form t
#   certificate chain for the server certificate. Alternative
#   the referenced file can be the same as SSLCertificateFile
#   when the CA certificates are directly appended to the ser
#   certificate for convinience.
SSLCertificateChainFile /etc/ssl/certs/ca-bundle.crt
```

To push visitors to HTTPS, put something similar to the following snippet i

```
RewriteEngine On

# Redirect all HTTP traffic to HTTPS.
RewriteCond %{HTTPS} !=on
RewriteRule ^/?(.*) https://%{SERVER_NAME}/$1 [R,L]
```

Make Use of that Diffie-Helman Group

The configuration needed to use the Diffie-Helman group in /etc/ssl/pri
the version of Apache. For the version used in Ubuntu 16.04, add or edit

available/ssl.conf:

```
# Protect against Logjam attacks. See: https://weakdh.org
SSLOpenSSLConfCmd DHParameters "/etc/ssl/private/dhparams.pem"
```

Now restart Apache to pick up the changes, after which you should be
homepage and see that you are automatically redirected to HTTPS.

```
service apache2 restart
```

Install the Mailserver Packages

Now we're ready to start in on the harder stuff. As for the LAMP server,
the basic packages for a mail server. Again, note the "^" at the end of the

```
apt-get install --assume-yes mail-server^
```

When Postfix installs, you will be asked to choose a general type of mail
Internet site. You will be asked for the system mail name, which is the h
mail.example.com. When Dovecot installs you will be asked whethe
certificate. Here answer no, as you will be adjusting the Dovecot configur
default snakeoil certificate or, ideally, your purchased SSL certificate.

What this set of packages provides to you is pretty much just the bar
mailserver that manages its users as straightforward Unix users and does
data. That is not the goal here, so we need the rest of the cast, such a
Dovecot, and a coterie of spam-mashing packages:

```
apt-get install --assume-yes \
  postfix-mysql \
  dovecot-mysql \
  postgrey \
  amavis \
  clamav \
  clamav-daemon \
  spamassassin \
  libdbi-perl \
  libdbd-mysql-perl \
  php7.0-imap \
  postfix-policyd-spf-python
```

The libdbi-perl and libdbd-mysql-perl packages are required by Amavis. Th
provides support for POP3 as well as the IMAP protocol, and will be need
the possible options for PHP webmail applications. The postfix-policyd-spf-
of Sender Policy Framework (SPF) on incoming mail to block spam. You w
point to have php7.0-imap running and ready:

```
service apache2 restart
```

Next you'll want to install a few optional packages that extend the abilities
packages by allowing greater inspection of attached files:

```
apt-get install --assume-yes \
  pyzor \
  razor \
  arj \
  cabextract \
  lzop \
  nomarch \
  p7zip-full \
  ripole \
  rpm2cpio \
  tnef \
  unzip \
  unrar-free \
  zip \
  zoo
```

Configure MySQL

A few alterations to the default MySQL configuration in /etc/mysql/mysc
Add the following:

```
# This removes NO_ZERO_IN_DATE and NO_ZERO_DATE, which cause p
# Postfix Admin code, from strict mode.
sql_mode=ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,ERROR_FOR_DIVI
```

Then restart the MySQL server:

```
service mysql restart
```

Create a Mail Database and User in MySQL

Log in to MySQL as the root user, entering the password you set earlier:

```
mysql -uroot -p
```

Now set up a database and user for the mail software. This database
accounts and mail domains, using schema set up by the Postfix Admin
better password than the example used below, but note that it will be ente
files. If an attacker gains shell access to the server, it doesn't matter how
from unauthorized access to the database is largely provided by the fact t
by the firewall, but it never hurts to have good passwords in place reg
layers of defense is always a good strategy.

```
create database mail;
grant all on mail.* to 'mail'@'localhost' identified by 'mailp
```

Install Postfix Admin 3.0.2 and the MySQL Schema

Postfix Admin is installed as follows. To start things off, download the pa
it, move it into a subdirectory of your webroot, and change ownership to t

```
wget http://downloads.sourceforge.net/project/postfixadmin/pos
tar -xf postfixadmin-3.0.2.tar.gz
rm -f postfixadmin-3.0.2.tar.gz
mv postfixadmin-3.0.2 /var/www/html/postfixadmin
chown -R www-data:www-data /var/www/html/postfixadmin
```

Next up is an interesting sort of a two-phase setup process. Fir
/var/www/html/postfixadmin/config.local.php :

```
touch /var/www/html/postfixadmin/config.local.php
chown www-data:www-data /var/www/html/postfixadmin/config.loca
```

Entries made in /var/www/html/postfixadmin/config.local.php will overr
/var/www/html/postfixadmin/config.inc.php. Add the following lines:

```
<?php
// Configuration options here override those in config.inc.ph

// You have to set $CONF['configured'] = true; before the
// application will run.
$CONF['configured'] = true;

// Postfix Admin Path
// Set the location of your Postfix Admin installation here.
// YOU MUST ENTER THE COMPLETE URL e.g. http://domain.tld/pos
$CONF['postfix_admin_url'] = 'https://mail.example.com/postfi

// Database connection details.
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'mail';
$CONF['database_password'] = 'mailpassword';
$CONF['database_name'] = 'mail';

// Site Admin
// Define the Site Admin's email address below.
// This will be used to send emails from to create mailboxes
// from Send Email / Broadcast message pages.
// Leave blank to send email from the logged-in Admin's Email
$CONF['admin_email'] = 'admin@example.com';

// Mail Server
// Hostname (FQDN) of your mail server.
// This is used to send email to Postfix in order to create m
$CONF['smtp_server'] = 'localhost';
$CONF['smtp_port'] = '25';

// Encrypt
// In what way do you want the passwords to be crypted?
// md5crypt = internal postfix admin md5
$CONF['encrypt'] = 'md5crypt';

// Default Aliases
// The default aliases that need to be created for all domain
$CONF['default_aliases'] = array (
    'abuse' => 'admin@example.com',
    'hostmaster' => 'admin@example.com',
    'postmaster' => 'admin@example.com',
    'webmaster' => 'admin@example.com'
);

// Footer
// Below information will be on all pages.
// If you don't want the footer information to appear set thi
$CONF['show_footer_text'] = 'YES';
$CONF['footer_text'] = 'Return to mail.example.com';
$CONF['footer_link'] = 'https://mail.example.com';

// Mailboxes
// If you want to store the mailboxes per domain set this to
// Examples:
//   YES: /usr/local/virtual/domain.tld/username@domain.tld
//   NO:  /usr/local/virtual/username@domain.tld
$CONF['domain_path'] = 'NO';
// If you don't want to have the domain in your mailbox set t
// Examples:
//   YES: /usr/local/virtual/domain.tld/username@domain.tld
//   NO:  /usr/local/virtual/domain.tld/username
// Note: If $CONF['domain_path'] is set to NO, this setting w
$CONF['domain_in_mailbox'] = 'YES';

// Specify '' for Dovecot and 'INBOX.' for Courier.
$CONF['create_mailbox_subdirs_prefix']='';
```

Note that the last items above are only for the purposes of defining hov
they don't set system paths for mailboxes. The actual system paths to
defined in the Dovecot configuration outlined in a later section of th
configuraton options relating to optional functionality that can be use
exploring the documentation and the main configuration file /var/www/htı

Next open up a web browser and visit your mail server at:

```
https://mail.example.com/postfixadmin/setup.php
```

Postfix Admin will automatically create its database schema at this point
page to choose a setup password and generate a hash of that pa
configuration file /var/www/html/postfixadmin/config.local.php and save i

```
// In order to setup Postfixadmin, you MUST specify a hashed p
// To create the hash, visit setup.php in a browser and type a
// on submission it will be echoed out to you as a hashed valu
$CONF['setup_password'] = '...a long hash string...';
```

Then return to the setup page. You can now use the password you sele
administrator account.

You should close off access to http://mail.example.com/postfixadmin/s
Create a file /var/www/html/postfixadmin/.htaccess and put the following

```
<Files "setup.php">
deny from all
</Files>
```

Create the Domain and Accounts in Postfix Admin

Now navigate to the main Postfix Admin login page:

```
https://mail.example.com/postfixadmin/
```

Log in as the newly created administrator account, and then choose th
option in order to create the example.com domain. Make sure that you
and note that you will probably want to change the default limits to allow
aliases. Next navigate to Domain List -> Domain List and click on the ı
From that page you can then add mail users (Add mailbox) and aliases (
schema, but it won't do anything else at this point as none of the other
configured to use the database.

Make sure that you create a mailbox for admin@example.com since your
that address. While the alias addresses do tend to receive spam, it is a ve
legitimate mail arriving at these addresses, as it can alert you to proble
personal mail server, but very important if you are in the business of se
good deliverability.

Postfix Admin does have another useful function during this lengthy setu
mail to local users through the web interface, which is helpful when testin
down errors.

Create a User to Handle Virtual Mail Directories

Virtual mail users are those that do not exist as Unix system users. They
methods of authentication or mail delivery and don't have home directori
things here: mail users are defined in the database created by Postfix
system users. Mail will be kept in subfolders per domain and acc
admin@example.com will have a mail directory of /var/vmail/exampl
directories will be owned by a single user called vmail, and Dovecot wil
create and update mail files.

```
useradd -r -u 150 -g mail -d /var/vmail -s /sbin/nologin -c "V
mkdir /var/vmail
chmod 770 /var/vmail
chown vmail:mail /var/vmail
```

Note that the user and virtual mail directory folder are using the "mail" g
users in that group to modify the contents.

Configure Dovecot

Dovecot will manage IMAP and POP3 connections, local mail directories, a
handed off from Postfix. It will also manage authentication for SMTP co
having two separate authentication systems when Dovecot can handle bo
across a number of files in /etc/dovecot and subfolders thereof, an
intimidating, it is all laid out fairly logically. The first thing to do is ensure
data in the database created by Postfix Admin, so you will
/etc/dovecot/dovecot-sql.conf.ext such that it uses the MySQL database c

```
# Database driver: mysql, pgsql, sqlite
driver = mysql
```

```
# Examples:
#   connect = host=192.168.1.1 dbname=users
#   connect = host=sql.example.com dbname=virtual user=virtual
#   connect = /etc/dovecot/authdb.sqlite
#
connect = host=localhost dbname=mail user=mail password=mailpa
```

```
# Default password scheme.
#
# List of supported schemes is in
# http://wiki2.dovecot.org/Authentication/PasswordSchemes
#
default_pass_scheme = MD5-CRYPT
```

```
# Define the query to obtain a user password.
#
# Note that uid 150 is the "vmail" user and gid 8 is the "mail
#
password_query = \
  SELECT username as user, password, '/var/vmail/%d/%n' as use
  'maildir:/var/vmail/%d/%n' as userdb_mail, 150 as userdb_uid
  FROM mailbox WHERE username = '%u' AND active = '1'
```

```
# Define the query to obtain user information.
#
# Note that uid 150 is the "vmail" user and gid 8 is the "mail
#
user_query = \
  SELECT '/var/vmail/%d/%n' as home, 'maildir:/var/vmail/%d/%n
  150 AS uid, 8 AS gid, concat('dirsize:storage=', quota) AS q
  FROM mailbox WHERE username = '%u' AND active = '1'
```

Then change the controlling definitions in /etc/dovecot/conf.d/10-auth.c
the SQL configuration files. While you are there, you should also make sur
disabled unless the connection is encrypted or local:

```
# Disable LOGIN command and all other plaintext authentication
# SSL/TLS is used (LOGINDISABLED capability). Note that if the
# matches the local IP (ie. you're connecting from the same co
# connection is considered secure and plaintext authentication
disable_plaintext_auth = yes
```

```
# Space separated list of wanted authentication mechanisms:
#   plain login digest-md5 cram-md5 ntlm rpa apop anonymous gs
#   gss-spnego
# NOTE: See also disable_plaintext_auth setting.
auth_mechanisms = plain login
```

```
##
## Password and user databases
##

#
# Password database is used to verify user's password (and no
# You can have multiple passdbs and userdbs. This is useful i
# allow both system users (/etc/passwd) and virtual users to
# duplicating the system users into virtual database.
#
# <doc/wiki/PasswordDatabase.txt>
#
# User database specifies where mails are located and what us
# own them. For single-UID configuration use "static" userdb.
#
# <doc/wiki/UserDatabase.txt>

#!include auth-deny.conf.ext
#!include auth-master.conf.ext

#!include auth-system.conf.ext
# Use the SQL database configuration for authentication rathe
# any of these others.
!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 up, you must tell Dovecot where to put the virtual user mail directo changes in /etc/dovecot/conf.d/10-mail.conf:

```
# Location for users' mailboxes. The default is empty, which
# tries to find the mailboxes automatically. This won't work
# doesn't yet have any mail, so you should explicitly tell Do
# location.
#
# If you're using mbox, giving a path to the INBOX file (eg.
# isn't enough. You'll also need to tell Dovecot where the ot
# kept. This is called the "root mail directory", and it must
# path given in the mail_location setting.
#
# There are a few special variables you can use, eg.:
#
#   %u - username
#   %n - user part in user@domain, same as %u if there's no d
#   %d - domain part in user@domain, empty if there's no doma
#   %h - home directory
#
# See doc/wiki/Variables.txt for full list. Some examples:
#
#   mail_location = maildir:~/Maildir
#   mail_location = mbox:~/mail:INBOX=/var/mail/%u
#   mail_location = mbox:/var/mail/%d/%1n/%n:INDEX=/var/index
#
# <doc/wiki/MailLocation.txt>
#
mail_location = maildir:/var/vmail/%d/%n
```

```
# System user and group used to access mails. If you use multi
# can override these by returning uid or gid fields. You can u
# or names. <doc/wiki/UserIds.txt>
mail_uid = vmail
mail_gid = mail
```

```
# Valid UID range for users, defaults to 500 and above. This i
# to make sure that users can't log in as daemons or other sys
# Note that denying root logins is hardcoded to dovecot binary
# be done even if first_valid_uid is set to 0.
#
# Use the vmail user uid here.
first_valid_uid = 150
last_valid_uid = 150
```

Let Dovecot know about either your snakeoil or purchased certific /etc/dovecot/conf.d/10-ssl.conf. Remember to include your CA certificate the certificate issuer:

```
# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>
ssl = yes

# PEM encoded X.509 SSL/TLS certificate and private key. They
# dropping root privileges, so keep the key file unreadable b
# root. Included doc/mkcert.sh can be used to easily generate
# certificate, just make sure to update the domains in doveco
#
# The generated snakeoil certificate:
#ssl_cert = </etc/ssl/certs/ssl-cert-snakeoil.pem
```

```
#ssl_key = </etc/ssl/private/ssl-cert-snakeoil.key
# Purchased certificate:
ssl_cert = </etc/ssl/certs/example.com.crt
ssl_key = </etc/ssl/private/example.com.key

# If key file is password protected, give the password here.
# give it when starting dovecot with -p parameter. Since this
# world-readable, you may want to place this setting instead
# root owned 0600 file by using ssl_key_password = <path.
#ssl_key_password =

# PEM encoded trusted certificate authority. Set this only if
# ssl_verify_client_cert=yes. The file should contain the CA
# followed by the matching CRL(s). (e.g. ssl_ca = </etc/ssl/c
ssl_ca = </etc/ssl/certs/ca-bundle.crt
```

You must also update the following lines in /etc/dovecot/conf.d/10-ssl
protocols that are no longer secure are not used:

```
# DH parameters length to use. In light of Logjam, has to be
# See: https://weakdh.org/sysadmin.html
ssl_dh_parameters_length = 2048

# SSL protocols to use. Don't use the no-longer secure protoc
ssl_protocols = !SSLv2 !SSLv3

# SSL ciphers to use. See:
# https://weakdh.org/sysadmin.html
# https://hynek.me/articles/hardening-your-web-servers-ssl-ci
ssl_cipher_list = ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES

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

Next, edit these lines in /etc/dovecot/conf.d/10-master.conf to add the Po

```
service auth {
  # auth_socket_path points to this userdb socket by default.
  # used by dovecot-lda, doveadm, possibly imap process, etc.
  # full permissions to this socket are able to get a list of
  # get the results of everyone's userdb lookups.
  #
  # The default 0666 mode allows anyone to connect to the soc
  # userdb lookups will succeed only if the userdb returns an
  # matches the caller process's UID. Also if caller's uid or
  # socket's uid or gid the lookup succeeds. Anything else ca
  #
  # To give the caller full permissions to lookup all users,
  # something else than 0666 and Dovecot lets the kernel enfo
  # permissions (e.g. 0777 allows everyone full permissions).
  unix_listener auth-userdb {
    mode = 0666
    user = vmail
    group = mail
  }

  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    # Assuming the default Postfix user and group
    user = postfix
    group = postfix
  }
```

You may have to explicitly set a postmaster address in /etc/dovecot
"Invalid settings: postmaster_address setting not given" showing up in th
that. Make sure that a suitable alias or mailbox exists for your chosen pos

```
# Address to use when sending rejection mails.
# Default is postmaster@<your domain>.
postmaster_address = postmaster@example.com
```

You must now ensure that the Dovecot configuration is accessible to both

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

A final note on Dovecot: it only creates a user's mail directory when mai
user. So creating a user in Postfix Admin will not result in the immediate c
/var/vmail, and that's just fine.

Configure Amavis, ClamAV, and SpamAssassin

Before configuring Postfix, we may as well take a short detour into confi
Their default configuration is close to what most people will need, and too
many of the optional additional packages you may have installed. If you
knowledge, you can of course spend a fair amount of time here crafting in
this is a quick and straightforward process, however. Note that here
relating to integration with Postfix - e.g. additions to the /etc/postfix/
section of this post.

First add Amavis and ClamAV users to one another's groups to enable the

```
adduser clamav amavis
adduser amavis clamav
```

This also requires editing the following lines in /etc/clamav/clamd.conf:

```
# Needed to allow things to work with Amavis, when both amavis
# users are added to one another's groups.
AllowSupplementaryGroups true
```

Then turn on Amavis by editing /etc/amavis/conf.d/15-content_filter_mo
default, so uncomment the @bypass... lines:

```
use strict;

# You can modify this file to re-enable SPAM checking through
# and to re-enable antivirus checking.

#
# Default antivirus checking mode
# Please note, that anti-virus checking is DISABLED by
# default.
# If You wish to enable it, please uncomment the following li

@bypass_virus_checks_maps = (
   \%bypass_virus_checks, \@bypass_virus_checks_acl, \$bypass

#
# Default SPAM checking mode
# Please note, that anti-spam checking is DISABLED by
# default.
# If You wish to enable it, please uncomment the following li

@bypass_spam_checks_maps = (
   \%bypass_spam_checks, \@bypass_spam_checks_acl, \$bypass_s

1;  # ensure a defined return
```

Now enable SpamAssassin by editing these lines in /etc/default/spamassa

```
# Change to one to enable spamd
ENABLED=1
```

```
# Cronjob
# Set to anything but 0 to enable the cron job to automaticall
# spamassassin's rules on a nightly basis
CRON=1
```

SpamAssassin under Amavis will only check mail that's determined to be
are a couple of ways to tell Amavis which mails are for local delivery, but
database set up by Postfix Admin. Edit /etc/amavis/conf.d/50-user to look

```
use strict;

#
# Place your configuration directives here.  They will overri
# earlier files.
#
# See /usr/share/doc/amavisd-new/ for documentation and examp
# the directives you can use in this file
#

# Three concurrent processes. This should fit into the RAM av
# AWS micro instance. This has to match the number of process
# for Amavis in /etc/postfix/master.cf.
$max_servers  = 3;

# Add spam info headers if at or above that level - this ensu
# are always added.
$sa_tag_level_deflt  = -9999;

# Check the database to see if mail is for local delivery, an
# should be spam checked.
@lookup_sql_dsn = (
    ['DBI:mysql:database=mail;host=127.0.0.1;port=3306',
     'mail',
     'mailpassword']);
$sql_select_policy = 'SELECT domain from domain WHERE CONCAT(

# Uncomment to bump up the log level when testing.
# $log_level = 2;

#------------ Do not modify anything below this line --------
1;  # ensure a defined return
```

Next make sure the ClamAV database is up to date by running freshclam.

```
freshclam
```

You will have to restart these processes to pick up the new configuration:

```
service clamav-daemon restart
service amavis restart
service spamassassin restart
```

Configure Postfix

Postfix handles incoming mail via the SMTP protocol, and its configuratio
to integrate with the various other packages we have installed so far. At
hand off incoming mail to the spam and virus checkers before passing it c
communicate with Dovecot in order to authenticate virtual users who are
to send mail.

Firstly you must create a set of files that describe for Postfix where to
domains. Note that the "hosts" directive in these files must be exactly th
the MySQL server configuration. If one side says "localhost" and the othe
may find that Postfix cannot connect to MySQL - strange but true. Here ar

/etc/postfix/mysql_virtual_alias_domainaliases_maps.cf

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
query = SELECT goto FROM alias,alias_domain
  WHERE alias_domain.alias_domain = '%d'
  AND alias.address=concat('%u', '@', alias_domain.target_doma
  AND alias.active = 1
```

/etc/postfix/mysql_virtual_alias_maps.cf

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
table = alias
select_field = goto
where_field = address
additional_conditions = and active = '1'
```

/etc/postfix/mysql_virtual_domains_maps.cf

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

/etc/postfix/mysql_virtual_mailbox_domainaliases_maps.cf

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
query = SELECT maildir FROM mailbox, alias_domain
  WHERE alias_domain.alias_domain = '%d'
  AND mailbox.username=concat('%u', '@', alias_domain.target_d
  AND mailbox.active = 1
```

/etc/postfix/mysql_virtual_mailbox_maps.cf

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
table = mailbox
select_field = CONCAT(domain, '/', local_part)
where_field = username
additional_conditions = and active = '1'
```

/etc/postfix/mysql_virtual_sender_login_maps.cf

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
query = SELECT goto FROM alias WHERE address='%s'
```

Now create the file /etc/postfix/header_checks, which will contain sor
headers when relaying mail. This improves privacy for the sending users
original IP address and mail software identifiers, for example. This file
Postfix configuration:

```
/^Received:/                IGNORE
/^User-Agent:/              IGNORE
/^X-Mailer:/                IGNORE
/^X-Originating-IP:/        IGNORE
/^x-cr-[a-z]*:/             IGNORE
/^Thread-Index:/            IGNORE
```

The following is the complete main Postfix configuration file at /etc/postf
number of complex choices and options on how mail is relayed and how
the scope of this post to explain each and every choice of best practic
detail. I strongly suggest that you spend some time reading up on Postfix
is easy to fall down and produce a suboptimal or faulty mailserver.

```
# See /usr/share/postfix/main.cf.dist for a commented, more

# The first text sent to a connecting process.
smtpd_banner = $myhostname ESMTP $mail_name
biff = no
# appending .domain is the MUA's job.
append_dot_mydomain = no
readme_directory = no

# --------------------------------
# SASL parameters
# --------------------------------

# Use Dovecot to authenticate.
smtpd_sasl_type = dovecot
# Referring to /var/spool/postfix/private/auth
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
broken_sasl_auth_clients = yes
smtpd_sasl_security_options = noanonymous
```

```
smtpd_sasl_local_domain =
smtpd_sasl_authenticated_header = yes

# --------------------------------
# TLS parameters
# --------------------------------

# The default snakeoil certificate. Comment if using a purch
# SSL certificate.
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key

# Uncomment if using a purchased SSL certificate.
# smtpd_tls_cert_file=/etc/ssl/certs/example.com.crt
# smtpd_tls_key_file=/etc/ssl/private/example.com.key

# The snakeoil self-signed certificate has no need for a CA
# if you are using your own SSL certificate, then you probab
# a CA certificate bundle from your provider. The path to th
# here.
# smtpd_tls_CAfile=/etc/ssl/certs/ca-bundle.crt

# Ensure we're not using no-longer-secure protocols.
smtpd_tls_mandatory_protocols=!SSLv2,!SSLv3

smtp_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
#smtpd_tls_session_cache_database = btree:${data_directory}/
#smtp_tls_session_cache_database = btree:${data_directory}/s

# Note that forcing use of TLS is going to cause breakage -
# don't offer it and so delivery will fail, both incoming an
# unfortunate given what various governmental agencies are u
#
# Enable (but don't force) all incoming smtp connections to
smtpd_tls_security_level = may
# Enable (but don't force) all outgoing smtp connections to
smtp_tls_security_level = may

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-do
# information on enabling SSL in the smtp client.

# --------------------------------
# TLS Updates relating to Logjam SSL attacks.
# See: https://weakdh.org/sysadmin.html
# --------------------------------

smtpd_tls_exclude_ciphers = aNULL, eNULL, EXPORT, DES, RC4,
smtpd_tls_dh1024_param_file = /etc/ssl/private/dhparams.pem

# --------------------------------
# SMTPD parameters
# --------------------------------

# Uncomment the next line to generate "delayed mail" warning
#delay_warning_time = 4h
# will it be a permanent error or temporary
unknown_local_recipient_reject_code = 450
# how long to keep message on queue before return as failed.
maximal_queue_lifetime = 7d
# max and min time in seconds between retries if connection
minimal_backoff_time = 1000s
maximal_backoff_time = 8000s
# how long to wait when servers connect before receiving res
smtp_helo_timeout = 60s
# how many address can be used in one message.
# effective stopper to mass spammers, accidental copy in who
# but may restrict intentional mail shots.
smtpd_recipient_limit = 16
# how many error before back off.
smtpd_soft_error_limit = 3
# how many max errors before blocking it.
smtpd_hard_error_limit = 12

# This next set are important for determining who can send m
# to other servers. It is very important to get this right -
# an open relay that allows unauthenticated sending of mail
#
# You are encouraged to read up on what exactly each of thes

# Requirements for the HELO statement
smtpd_helo_restrictions = permit_mynetworks, warn_if_reject
# Requirements for the sender details. Note that the order m
# E.g. see http://jimsun.linxnet.com/misc/restriction_order_
smtpd_sender_restrictions = permit_mynetworks, reject_authen
# Requirements for the connecting server
smtpd_client_restrictions = reject_rbl_client sbl.spamhaus.o
# Requirement for the recipient address. Note that the entry
# "check_policy_service inet:127.0.0.1:10023" enables Postgr
smtpd_recipient_restrictions = reject_unauth_pipelining, per
smtpd_data_restrictions = reject_unauth_pipelining
# This is a new option as of Postfix 2.10, and is required i
# smtpd_recipient_restrictions for things to work properly i
smtpd_relay_restrictions = reject_unauth_pipelining, permit_

# require proper helo at connections
smtpd_helo_required = yes
# waste spammers time before rejecting them
smtpd_delay_reject = yes
disable_vrfy_command = yes

# ----------------------------------
# General host and delivery info
# ----------------------------------

myhostname = mail.example.com
myorigin = /etc/hostname
# Some people see issues when setting mydestination explicit
# subdomain, while leaving it empty generally doesn't hurt.
# mydestination = mail.example.com, localhost
mydestination =
# If you have a separate web server that sends outgoing mail
# mailserver, you may want to add its IP address to the spac
# mynetworks, e.g. as 10.10.10.10/32.
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
mynetworks_style = host

# This specifies where the virtual mailbox folders will be l
virtual_mailbox_base = /var/vmail
# This is for the mailbox location for each user. The domain
# map allows us to make use of Postfix Admin's domain alias
virtual_mailbox_maps = mysql:/etc/postfix/mysql_virtual_mail
# and their user id
virtual_uid_maps = static:150
# and group id
virtual_gid_maps = static:8
# This is for aliases. The domainaliases map allows us to ma
# use of Postfix Admin's domain alias feature.
virtual_alias_maps = mysql:/etc/postfix/mysql_virtual_alias_
# This is for domain lookups.
virtual_mailbox_domains = mysql:/etc/postfix/mysql_virtual_d
# Used in conjunction with reject_authenticated_sender_login
# verify that the sender is sending with their own address,
# of the aliases mapped to that address.
smtpd_sender_login_maps = mysql:/etc/postfix/mysql_virtual_s

# --------------------------------
# Integration with other packages
# ---------------------------------------

# Tell postfix to hand off mail to the definition for doveco
virtual_transport = dovecot
dovecot_destination_recipient_limit = 1

# Use amavis for virus and spam scanning
content_filter = amavis:[127.0.0.1]:10024

# Settings for checking SPF to cut down spam.
policy-spf_time_limit = 3600s

# --------------------------------
# Header manipulation
# --------------------------------------

# Getting rid of unwanted headers. See: https://posluns.com/
header_checks = regexp:/etc/postfix/header_checks
# getting rid of x-original-to
enable_original_recipient = no
```

To be clear, if you are using a purchased SSL certificate - and have a CA
- then you will have to alter these lines in /etc/postfix/main.cf:

```
# The default snakeoil certificate. Comment if using a purcha
# SSL certificate.
# smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
# smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key

# Uncomment if using a purchased SSL certificate.
smtpd_tls_cert_file=/etc/ssl/certs/example.com.crt
smtpd_tls_key_file=/etc/ssl/private/example.com.key

# The snakeoil self-signed certificate has no need for a CA f
# if you are using your own SSL certificate, then you probabl
# a CA certificate bundle from your provider. The path to tha
# here.
smtpd_tls_CAfile=/etc/ssl/certs/ca-bundle.crt
```

You must also expand /etc/postfix/master.cf, and here is the entire file
the default material from the package install, such as commented options

```
#
# Postfix master process configuration file.  For details on
# of the file, see the master(5) manual page (command: "man
# on-line: http://www.postfix.org/master.5.html).
#
# Do not forget to execute "postfix reload" after editing th
#
# ==============================================================
# service type  private unpriv  chroot  wakeup  maxproc comm
#               (yes)   (yes)   (no)    (never) (100)
# ==============================================================
smtp      inet  n       -       y       -       -       smtp
#smtp      inet  n       -       y       -       1       pos
#smtpd     pass  -       -       y       -       -       smt
#dnsblog   unix  -       -       y       -       0       dns
#tlsproxy  unix  -       -       y       -       0       tls
```

```

# SMTP with TLS on port 587. Currently commented.
#submission inet n       -       y       -       -       smt
#  -o syslog_name=postfix/submission
#  -o smtpd_tls_security_level=encrypt
#  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_enforce_tls=yes
#  -o smtpd_client_restrictions=permit_sasl_authenticated,re
#  -o smtpd_sasl_tls_security_options=noanonymous

# SMTP over SSL on port 465.
smtps     inet  n       -       y       -       -       smtp
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_tls_auth_only=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated,rej
  -o smtpd_sasl_security_options=noanonymous,noplaintext
  -o smtpd_sasl_tls_security_options=noanonymous

#628       inet  n       -       y       -       -       qmq
pickup    unix  n       -       y       60      1       pick
cleanup   unix  n       -       y       -       0       clea
qmgr      unix  n       -       n       300     1       qmgr
#qmgr     unix  n       -       n       300     1       oqmg
tlsmgr    unix  -       -       y       1000?   1       tlsm
rewrite   unix  -       -       y       -       -       triv
bounce    unix  -       -       y       -       0       boun
defer     unix  -       -       y       -       0       boun
trace     unix  -       -       y       -       0       boun
verify    unix  -       -       y       -       1       veri
flush     unix  n       -       y       1000?   0       flus
proxymap  unix  -       -       n       -       -       prox
proxywrite unix -       -       n       -       1       prox
smtp      unix  -       -       y       -       -       smtp
relay     unix  -       -       y       -       -       smtp
#       -o smtp_helo_timeout=5 -o smtp_connect_timeout=5
showq     unix  n       -       y       -       -       show
error     unix  -       -       y       -       -       erro
retry     unix  -       -       y       -       -       erro
discard   unix  -       -       y       -       -       disc
local     unix  -       n       n       -       -       loca
virtual   unix  -       n       n       -       -       virt
lmtp      unix  -       -       y       -       -       lmtp
anvil     unix  -       -       y       -       1       anvi
scache    unix  -       -       y       -       1       scac
#
# ==========================================================
# Interfaces to non-Postfix software. Be sure to examine the
# pages of the non-Postfix software to find out what options
#
# Many of the following services use the Postfix pipe(8) del
# agent.  See the pipe(8) man page for information about ${r
# and other message envelope options.
# ==========================================================
#
# maildrop. See the Postfix MAILDROP_README file for details
# Also specify in main.cf: maildrop_destination_recipient_li
#
maildrop  unix  -       n       n       -       -       pipe
  flags=DRhu user=vmail argv=/usr/bin/maildrop -d ${recipien
#
# ==========================================================
#
# Recent Cyrus versions can use the existing "lmtp" master.c
#
# Specify in cyrus.conf:
#   lmtp    cmd="lmtpd -a" listen="localhost:lmtp" proto=tcp
#
```

```
# Specify in main.cf one or more of the following:
#  mailbox_transport = lmtp:inet:localhost
#  virtual_transport = lmtp:inet:localhost
#
# ==========================================================
#
# Cyrus 2.1.5 (Amos Gouaux)
# Also specify in main.cf: cyrus_destination_recipient_limit
#
#cyrus     unix  -       n       n       -       -       pip
#  user=cyrus argv=/cyrus/bin/deliver -e -r ${sender} -m ${e
#
# ==========================================================
# Old example of delivery via Cyrus.
#
#old-cyrus unix  -       n       n       -       -       pip
#  flags=R user=cyrus argv=/cyrus/bin/deliver -e -m ${extens
#
# ==========================================================
#
# See the Postfix UUCP_README file for configuration details
#
uucp      unix  -       n       n       -       -       pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nextho
#
# Other external delivery methods.
#
ifmail    unix  -       n       n       -       -       pipe
  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop (
bsmtp     unix  -       n       n       -       -       pipe
  flags=Fq. user=bsmtp argv=/usr/lib/bsmtp/bsmtp -t$nexthop
scalemail-backend unix  -   n   n   -   2   pipe
  flags=R user=scalemail argv=/usr/lib/scalemail/bin/scalema
mailman   unix  -       n       n       -       -       pipe
  flags=FR user=list argv=/usr/lib/mailman/bin/postfix-to-ma
  ${nexthop} ${user}

# The next two entries integrate with Amavis for anti-virus/
amavis      unix    -       -       y       -       3
  -o smtp_data_done_timeout=1200
  -o smtp_send_xforward_command=yes
  -o disable_dns_lookups=yes
  -o max_use=20
127.0.0.1:10025 inet    n       -       y       -       -
  -o content_filter=
  -o local_recipient_maps=
  -o relay_recipient_maps=
  -o smtpd_restriction_classes=
  -o smtpd_delay_reject=no
  -o smtpd_client_restrictions=permit_mynetworks,reject
  -o smtpd_helo_restrictions=
  -o smtpd_sender_restrictions=
  -o smtpd_recipient_restrictions=permit_mynetworks,reject
  -o smtpd_data_restrictions=reject_unauth_pipelining
  -o smtpd_end_of_data_restrictions=
  -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 smtpd_client_connection_count_limit=0
  -o smtpd_client_connection_rate_limit=0
  -o receive_override_options=no_header_body_checks,no_unkno

# Integration with Dovecot - hand mail over to it for local
# run the process under the vmail user and mail group.
dovecot      unix    -       n      n       -       -  pipe
  flags=DRhu user=vmail:mail argv=/usr/lib/dovecot/dovecot-l

# Integration with the SPF check package.
```

```
policy-spf  unix  -       n       n       -       -       sp
     user=nobody argv=/usr/bin/policyd-spf
```

Note that Amavis is restricted to three processes, which should be fine f
The processes are memory-heavy, so start low and add more only if you
see the notes in this guide for pointers on how to do that.

Restart Everything, and Test the Server

Restart all the necessary processes to pick up configuration changes:

```
service postfix restart
service spamassassin restart
service clamav-daemon restart
service amavis restart
service dovecot restart
```

Now start testing! Keep an eye on /var/log/mail.log for error messages
IMAP, send mail to an account created on the server, and send mail from
up the firewall to allow global access to the relevant ports before doir
Google is your friend when it comes to searching on specific error messag
of any specific problem.

AWS Mail Restrictions and Reverse DNS Lookup

Once configured, with IP address set and DNS records set up, you'll nee
put in place for your server and the AWS outgoing mail restrictions
standard customer service form. This doesn't take long, and it can actuall
necessary, prior to the server completion.

Install and Set up Monit for Monitoring

Monit is a very useful monitoring tool that helps rescue your server
through apt-get:

```
apt-get install --assume-yes monit
```

The following are a set of fairly trivial instructions that set monit to w
processes, but without issuing notifications or doing much more than res
Amavis configuration specifies a fairly infrequent check, as it is possible tc
where it refuses connections because you're sending mail too rapidly ar
number of concurrent connections per process (which is set at a low 128
restart Amavis at that point just makes things worse, boosting load and
queued and reattempted for any period while Amavis is truly down and wa

Create the following files in the Monit configuration directory.

/etc/monit/conf.d/amavis

```
check process amavisd with pidfile /var/run/amavis/amavisd.pid
  every 5 cycles
  group mail
  start program = "/usr/sbin/service amavis start"
  stop  program = "/usr/sbin/service amavis stop"
  if failed port 10024 protocol smtp then restart
  if 5 restarts within 25 cycles then timeout
```

/etc/monit/conf.d/apache2

```
check process apache2 with pidfile /var/run/apache2/apache2.p
  group www
  start program = "/usr/sbin/service apache2 start"
  stop program = "/usr/sbin/service apache2 stop"
  if failed host localhost port 80 protocol http
    with timeout 10 seconds
    then restart
  if failed host localhost port 443 type tcpssl protocol http
    with timeout 10 seconds
    then restart
  if 5 restarts within 5 cycles then timeout
```

/etc/monit/conf.d/dovecot

```
check process dovecot with pidfile /var/run/dovecot/master.pi
  group mail
  start program = "/usr/sbin/service dovecot start"
  stop program = "/usr/sbin/service dovecot stop"
  group mail
  # We'd like to use this line, but see:
  # http://serverfault.com/questions/610976/monit-failing-to-
  #if failed port 993 type tcpssl sslauto protocol imap for 5
  if failed port 993 for 5 cycles then restart
  if 5 restarts within 25 cycles then timeout
```

/etc/monit/conf.d/mysql

```
check process mysqld with pidfile /var/run/mysqld/mysqld.pid
  group database
  start program = "/usr/sbin/service mysql start"
  stop program = "/usr/sbin/service mysql stop"
  if failed host localhost port 3306 protocol mysql then resta
  if 5 restarts within 5 cycles then timeout
```

/etc/monit/conf.d/postfix

```
check process postfix with pidfile /var/spool/postfix/pid/mast
  group mail
  start program = "/usr/sbin/service postfix start"
  stop  program = "/usr/sbin/service postfix stop"
  if failed port 25 protocol smtp then restart
  if 5 restarts within 5 cycles then timeout
```

/etc/monit/conf.d/spamassassin

```
check process spamassassin with pidfile /var/run/spamassassin.
  group mail
  start program = "/usr/sbin/service spamassassin start"
  stop  program = "/usr/sbin/service spamassassin stop"
  if 5 restarts within 5 cycles then timeout
```

/etc/monit/conf.d/sshd

```
check process sshd with pidfile /var/run/sshd.pid
  start program "/usr/sbin/service ssh start"
  stop program "/usr/sbin/service ssh stop"
  if failed host 127.0.0.1 port 22 protocol ssh then restart
   if 5 restarts within 5 cycles then timeout
```

Now enable the local Monit HTTP interface by editing /etc/monit/monit
lines. This enables use of commands such as monit status. Run moni
available options.

```
## Monit has an embedded HTTP interface which can be used to
## services monitored and manage services from a web interfac
## interface is also required if you want to issue Monit comm
## command line, such as 'monit status' or 'monit restart ser
## for this is that the Monit client uses the HTTP interface
## commands to a running Monit daemon. See the Monit Wiki if
## enable SSL for the HTTP interface.
#
set httpd port 2812 and
    use address localhost  # only accept connection from loca
    allow localhost        # allow localhost to connect to th
#     allow admin:monit      # require user 'admin' with pass
```

Then restart Monit to pick up the new orders:

```
service monit restart
```

Monit offers options for notifications, a web console, restarting on high
other amenities, so you may want to add more to this very basic config
familiar with the application. One important item to note is that once att
Monit will no longer monitor that service, even after both service and Mo
Check to see which services are being monitored by running:

```
monit status
```

Then monitor a service with:

```
# monit monitor <name>, e.g.:
monit monitor mysqld
```

Install Roundcube for Webmail

Roundcube is a straightforward PHP webmail package: if all you need is
via a web interface then this is for you. There are other, more compl
options out there but you pay the price for that in the time taken to in
Roundcube is a much less onerous experience, but unfortunately the in
online on how to install Roundcube are, shall we say, somewhat confused
the wrong path if working from a package install on Ubuntu. Here inste
manage things.

Start by installing the necessary packages. The plugin packages aren't ess
them over to see what is available:

```
apt-get install --assume-yes \
  roundcube \
  roundcube-plugins \
  roundcube-plugins-extra \
  php-mail \
  php-mail-mimedecode \
  php-mime-type \
  php-mail-mime \
  php7.0-intl \
  php7.0-zip

pear install Net_IDNA2-0.1.1
```

In the package installation process you will be asked whether the installe
Answer Yes, then choose mysql as the database type. You'll be asked fo
so enter it. Then you will be asked to enter and confirm a password for
that will be created for you. The same comments on passwords apply her
for the root and mail user.

The php7.0-intl and php7.0-zip modules should in theory be enabled by c
problems getting PHP and Apache to realize that fact. Try enabling them
server, which is what finally worked for me:

```
phpenmod intl zip
```

Either way, restart Apache to pick up changes:

```
service apache2 restart
```

Edit the following lines in /etc/roundcube/config.inc.php to tell Ro
applications are running on the same machine as it is:

```
// The mail host chosen to perform the log-in.
// Leave blank to show a textbox at login, give a list of hos
// to display a pulldown menu or set one host as string.
// To use SSL/TLS connection, enter hostname with prefix ssl:
// Supported replacement variables:
// %n - hostname ($_SERVER['SERVER_NAME'])
// %t - hostname without the first part
// %d - domain (http hostname $_SERVER['HTTP_HOST'] without t
// %s - domain name after the '@' from e-mail address provide
// For example %n = mail.domain.tld, %t = domain.tld
// WARNING: After hostname change update of mail_host column
//          required to match old user data records with the
$config['default_host'] = 'localhost';

// SMTP server host (for sending mails).
// To use SSL/TLS connection, enter hostname with prefix ssl:
// If left blank, the PHP mail() function is used
// Supported replacement variables:
// %h - user's IMAP hostname
// %n - hostname ($_SERVER['SERVER_NAME'])
// %t - hostname without the first part
// %d - domain (http hostname $_SERVER['HTTP_HOST'] without t
// %z - IMAP domain (IMAP hostname without the first part)
// For example %n = mail.domain.tld, %t = domain.tld
$config['smtp_server'] = 'localhost';
```

As /etc/roundcube/config.inc.php overrides /etc/roundcube/defaults.inc.
number of the overall properties, you will have to add these lines to /
order to tell Roundcube to (a) redirect non-secure HTTP connections to I
for caching:

```
// enforce connections over https
// with this option enabled, all non-secure connections will
// set the port for the ssl connection as value of this optio
$config['force_https'] = true;

// Type of IMAP indexes cache. Supported values: 'db', 'apc'
$config['imap_cache'] = 'db';

// Backend to use for session storage. Can either be 'db' (de
$config['session_storage'] = 'db';
```

Ensure that you update the following configuration option in /etc/round
value for this installation:

```
// this key is used to encrypt the users imap password which i
// in the session record (and the client cookie if remember pa
// please provide a string of exactly 24 chars.
// YOUR KEY MUST BE DIFFERENT THAN THE SAMPLE VALUE FOR SECURI
$config['des_key'] = 'enter a unique value here';
```

At this point Roundcube is now installed and minimally configured, but it
webroot. The Roundcube webroot containing PHP files and va
/var/lib/roundcube, and the next step is to make that available to visitor
creating a symlink in the webroot:

```
ln -s /var/lib/roundcube /var/www/html/roundcube
```

Now redirect the default landing page for visitors to Roundcube, which
index page out of the way:

```
mv /var/www/html/index.html /var/www/html/index.bak.html
```

Then expand /var/www/html/.htaccess to include a rule to redirect just
Being this selective leaves the open the option of adding other
/var/www/html for whatever you might want to use them for, and preserv

```
RewriteEngine On

# Redirect all HTTP traffic to HTTPS.
RewriteCond %{HTTPS} !=on
RewriteRule ^/?(.*) https://%{SERVER_NAME}/$1 [R,L]

# Send / to /roundcube.
RewriteRule ^/?$ /roundcube [L]
```

You can now test Roundcube by visiting http://mail.example.com and log

Add Two Factor Authentication to Roundcube

Two factor authentication (2FA) is increasingly a good idea in this age of c
Adding it is entirely optional, but note that there is a decent 2FA plugin
2Steps Verification. The documentation in that repository covers installat
isn't hard to do at all, so it won't be repeated here.

Options to Help Ensure Deliverability

If you want a quiet life free from worries about whether your mail destination, then you should take the small amount of additional time t mail server. These modifications go a long way towards ensuring th triggering false positives from spam filters. It is becoming ever harder the reaches its recipients. Set up a mail server in the cloud that happens t some time in the past used by spammers, or engage in a serious discus spam, and you can wave goodbye to the certainty of delivery of your ma similar email providers. These entities maintain their own arcane anti-s and distinct from the open world of IP address blacklists. Resolving issu attention of anyone who might help is essentially impossible if you don't h

Thus everyone who builds their own mail server should set up the tw providers pay attention to when it comes to the decision tree for their Policy Framework (SPF) and DomainKeys Identified Mail (DKIM).

Set Up Sender Policy Framework (SPF)

SPF requires only that you add a TXT record to your DNS zone for the do on the tools provided by your domain registrar, or the tools you set up y own nameservers. If using a registrar's web interface to make DNS cha the option to enter a subdomain for the record. If you do, then leave that

This generic SPF TXT record authorizes mail originating from mail se identified by MX records and all other servers associated with your domai

```
"v=spf1 a mx -all"
```

Note that the double quotes are a necessary part of the SPF TXT record. than this are possible, as outlined in the SPF documentation. You can s examples in the wild by using the dig command. e.g.:

```
dig google.com txt
```

Server Setup For DomainKeys Identified Mail (DKIM)

Setting up DKIM is a little more involved than SPF. First install the necess

```
apt-get install --assume-yes opendkim opendkim-tools
```

Add the following to /etc/opendkim.conf:

```
Domain    example.com
KeyFile    /etc/postfix/dkim.key
Selector    dkim
SOCKET    inet:8891@localhost
```

Add the following to /etc/default/opendkim:

```
SOCKET="inet:8891@localhost"
```

Append a suitable DKIM configuration to /etc/postfix/main.cf:

```
# -------------------------------------
# DKIM
# -------------------------------------
milter_default_action = accept
milter_protocol = 6
smtpd_milters = inet:localhost:8891
non_smtpd_milters = inet:localhost:8891
```

Now you can generate a private key for signing outgoing mail. Note that
is the value given to Selector in /etc/opendkim.conf. This can be any
consistent about replacing dkim with your desired value everywhere i
command to generate the key and associated materials in the form
dkim.txt. The former is the RSA private key, while the latter contains the
into your DNS records.

```
opendkim-genkey -t -s dkim -d example.com
```

Move the key into place and grant suitable permissions, but don't forget t
backed up somewhere safe:

```
mv dkim.private /etc/postfix/dkim.key
chmod 660 /etc/postfix/dkim.key
chown root:opendkim /etc/postfix/dkim.key
```

You'll need to restart Postfix and OpenDKIM services to pick up the config
mail is signed using DKIM:

```
service opendkim start
service postfix restart
```

DNS Setup For DomainKeys Identified Mail (DKIM)

Next up is the DNS record setup. How you do this is again completely dep
or how it is managed for you - everyone's tools are different. Note that
create raw TXT records with specific subdomains, which will prevent you
If this is the case, then you will have to transfer your domain to a real re
the toys.

The file dkim.txt contains the following content, the full TXT record th
subdomain dkim._domainkey and a long set of encoded content as the
given to Selector in /etc/opendkim.conf.

```
dkim._domainkey IN  TXT ( "v=DKIM1; k=rsa; t=y; "
"p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC9rulKo58JIb5h+3MMEnY
```

When adding the DNS record you should omit the k=rsa; t=y; portion o
to the key format and that defaults to RSA. They second denotes that th
be included. Thus the value to add looks like this:

```
"v=DKIM1; p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC9rulKo58JIb
```

It is helpful to view examples in the wild for the purposes of compariso
should enter the value into your records. You can use the DKIM key che
Note that the dkim._domainkey is the subdomain in the following comma

```
dig dkim._domainkey.twitter.com txt
```

Monitor the OpenDKIM Service

Add a configuration file /etc/monit/conf.d/opendkim:

```
check process opendkim with pidfile /var/run/opendkim/opendkim
  group mail
  start program = "/etc/init.d/opendkim start"
  stop  program = "/etc/init.d/opendkim stop"
  if 5 restarts within 5 cycles then timeout
```

Then restart Monit:

```
service monit restart
```

Sharing a DKIM Key for Multiple Domains

If you are serving multiple domains from the same mail server, then the
should be:

```
Domain    *
KeyFile    /etc/postfix/dkim.key
Selector    dkim
SOCKET    inet:8891@localhost
```

This will work unless you are running a mailing list or similar, in which
containing addresses in domains other than the list domain. In that ca
ServerFault question and answer for more complex configurations that car

Testing the SPF and DKIM Configuration

The only certain test is to send mail from the server and inspect it. A dec
which will report on the validity of the SPF and DKIM additions to mail
DNS changes a chance to propagate before using it.

Notes on Postgrey

You'll find that Postgrey has its own idiosyncratic notion of what default co
should work. There are whitelist configuration files for
/etc/postgrey/whitelist_clients and /etc/postgrey/whitelist_recipients r
actually used by default. If you want to use them, you must either copy th

```
cp /etc/postgrey/whitelist_clients /etc/postfix/postgrey_white
cp /etc/postgrey/whitelist_recipients /etc/postfix/postgrey_wh
```

Or, alternatively, edit /etc/default/postgrey:

```
# postgrey startup options, created for Debian

# you may want to set
#   --delay=N   how long to greylist, seconds (default: 300)
#   --max-age=N delete old entries after N days (default: 35)
# see also the postgrey(8) manpage

POSTGREY_OPTS="--inet=10023"
POSTGREY_OPTS="$POSTGREY_OPTS --whitelist-clients=/etc/postgr
POSTGREY_OPTS="$POSTGREY_OPTS --whitelist-recipients=/etc/pos

# the --greylist-text commandline argument can not be easily
# POSTGREY_OPTS when it contains spaces.  So, insert your tex
#POSTGREY_TEXT="Your customized rejection message here"
```

Notes on Serving Multiple Domains

You can create multiple domains in Postfix Admin if so desired, under
Additional domains added in Postfix Admin can be aliased to existing do
Alias Domain, such that address@example1.com is always forwarded to
can stand as distinct domains with their own mailboxes, aliases, and so fo

Depending on your use case, you might also want to adjust some of the
accessing the site at mail.example1.com, mail.example2.com, and so
redirect to SSL to recognize all of the domains used.

Roundcube should work for multiple domains, but requires you to cre
Fortunately the instructions are clear.

The tricky part of the setup for multiple domains relates to SSL certific
DNS lookup for outgoing mail. Postfix requires one IP address per certifica
could be provisioned with multiple IP addresses and certificates, which
Alternatively, use a single IP address and multi-domain UCC certificate
wants the reverse DNS lookup to be consistent, however, and in a
addresses, the outgoing mail will all appear to be coming from the single
domain it is sent from. This is a problem. The simplest solution to work ar
is to assign the same mail.example.com MX record for all of the domains
are some examples that explain this setup if you look around online.

Notes on Setting Up as a Backup MX

A backup mail server will receive mail when the primary is offline, and the
it is available. For a personal mail server this is probably unnecessary, but
want to set up the server you are building as one of the backup mail serv
to this recipe are required. These are taken from a short guide that you m

1) Firstly create /etc/postfix/mysql_relay_domains_maps.cf containing th

```
user = mail
password = mailpassword
hosts = 127.0.0.1
dbname = mail
query = SELECT domain FROM domain WHERE domain='%s' AND backup
```

2) Then add the following to /etc/postfix/main.cf:

```
# This is a backup MX server, and this line tells Postfix
# where to send the mail.
```

```
relay_domains = proxy:mysql:/etc/postfix/mysql_relay_domains_m
```

3) In the domain configuration in Postfix Admin, check the "Mail server is

4) To enable delivery to the backup, the MX DNS records must include an primary mail server. E.g.:

```
name                     priority  ip-address
mail.example.com           10      172.10.10.10
mail-backup.example.com    20      172.10.10.11
```

5) Lastly, you will need to set up a script or process of change replicat entries for users and domains are the same in primary and backup mail MX flag.

Notes on Managing Quotas

If you've been following carefully, you will note that nothing has been s disk space quotas. It was not an important goal for the work that p instructions. As things stand the necessary fields for quota managment are not used, as (a) the quota module isn't enabled by default in Dovecot, to use quotas by default.

So if you want to enable disk quotas, you should first of all alter the Post /var/www/html/postfixadmin/config.inc.php by adding /var/www/html/postfixadmin/config.local.php :

```
// Quota
// When you want to enforce quota for your mailbox users set t
$CONF['quota'] = 'YES';
// You can either use '1024000' or '1048576'
$CONF['quota_multiplier'] = '1024000';
```

```
// Optional:
// Show used quotas from Dovecot dictionary backend in virtua
// mailbox listing.
// See: DOCUMENTATION/DOVECOT.txt
//      http://wiki2.dovecot.org/Quota/Dict
//
$CONF['used_quotas'] = 'YES';

// if you use dovecot >= 1.2, set this to yes.
// Note about dovecot config: table "quota" is for 1.0 & 1.1,
// table "quota2" is for dovecot 1.2 and newer
$CONF['new_quota_table'] = 'YES';
```

Next, you will want to enable and configure the quota and imap_quota manages quotas while the latter enables reporting on quotas via IMAP. Y following documentation for instructions on how to do this:

- Quota (Dovecot 2.*)
- Quota Configuration (Dovecot 2.*)

These configuration changes will be made in 10-mail.conf and 90-quota directory.

Bypassing Spam and Virus Checks for Local Mail

If you're in the business of sending out newsletters or frequent updates
completely control the content in those emails, then you probably dor
checks for those items. It's a pointless use of server processing cycles, a
the server if you are making it process the full range of checks on each an

To have amavisd-new skip the checks for mail originating from a known
from a web application on another server, etc), edit /etc/amavis/conf.d/5(

```
# Replace 111.111.111.111/32 with your desired list of client
# ranges which will bypass checks.
@mynetworks = qw( 127.0.0.0/8 [::1] 111.111.111.111/32 );

# Rules for clients defined in @mynetworks
$policy_bank{'MYNETS'} = {
  bypass_spam_checks_maps   => [1],  # don't spam-check inter
  bypass_banned_checks_maps => [1],  # don't banned-check int
  bypass_header_checks_maps => [1],  # don't header-check int
};
```

Replace 111.111.111.111/32 with whatever set of IP address ranges yo
checks. All mail arriving from those sources will fall into MYNETS for am
checking. If bypassing by IP address doesn't fit your needs, you can fin
users, destinations, or sources in a helpful, if dated guide to amavisd-new

Some Final Notes on Security

You'll note that there are a fair number of configuration files that contain
and webmail data in this server, and that includes PHP files sitting in th
dominant security concern: the mail users are virtual and only the server
in as a system user. On AWS the default setup is for SSH login to use k
only the ubuntu user has a key setup to allow login. You can also ea
selected IP addresses via the security group applied to the server. Further,
to ensure that no web visitor can directly view configuration files - a
includes, which covers the rare case where some error causes PHP files
text. MySQL access is from localhost only, in any case.

All in all the lowest bar from a traditional security perspective is probabl
runs a couple of complicated PHP web applications with database acces
involve a way to upload and execute an arbitrary PHP script or shell com
permissions, or various other XSS attacks allowing for session hijacking
just by getting into the databases, compromise of the webroot is com
functions of the server. Major PHP webmail applications have exhibited r
past years, but at some point you have to pick your software. On the who
with the output of established development communities whose memb
record of vulnerabilities found and fixed, and where there are a large
codebase.

These are all good reasons for setting up your webmail on a different ser
and Dovecot - something to bear in mind.

Of course being on AWS - or indeed any sort of easily available hosting in
in your front room - means that the US government has free acces
particularly feel up to the task, and you may never know a copy w

forthcoming evolutions in virtual hosting services will be some form
operations such that you can have the convenience of an AWS-style serv
it affords the present day panopticon-in-the-making.

Further, it is apparently the case that all email traffic between mail serv
governmental agencies. Unfortunately the present state of SMTP in the
servers do not implement the ability to pass emails over an encrypted
setup and enforce encryption for POP, IMAP, and webmail connections be
email traffic between mail servers is often plain text. Forcing your
connections with other servers will mean that a large fraction of your ema
rejected. Thus the configuration provided for Postfix in this post is for o
and received will be encrypted if the mail server on the other end of the c