Installing The Postfix MTA

This page is supplemental to main article: Creating a Virtual Mail Server with Postfix, Dovecot and MySQL

Postfix is a popular and secure mail transport agent, or MTA.

We will build and install Postfix using the build script from SlackBuilds.org or SBo. The steps shown here use versions current as of this writing, but you should use the latest version applicable to your Slackware version at the time you build it.

By default the SBo postfix script builds with dovecot for SASL, which we need, but without the database support we also need. You might want to check the SlackBuild script at the time you build for changes or other options.

We will assume that you are familiar with SlackBuilds and will provide only the essential steps for building postfix here. For more detailed information please visit the SBo How-To page.

If building for a Linode.com VPS you will either need to install the kernel-headers package, or preferably build on a local machine and then transfer the package to the VPS platform. You will also need to install the db48 package missing from the Linode.com Slackware.

The essential steps for building postfix with the required database support are (as root):

```
cd /tmp
wget http://slackbuilds.org/slackbuilds/14.1/network/postfix.tar.gz
tar -xvzf postfix.tar.gz
cd postfix
cat postfix.info
...
DOWNLOAD="http://postfix.cs.utah.edu/source/official/postfix-2.11.3.tar.gz"
MD5SUM="c3f0f51d8865559b40e9350eb3816011"
...

# Fetch archive from URL in DOWNLOAD line #
wget http://postfix.cs.utah.edu/source/official/postfix-2.11.3.tar.gz

# Verify integrity of archive - compare to MD5SUM line #
md5sum postfix-2.11.3.tar.gz
c3f0f51d8865559b40e9350eb3816011

# Now build with database support #
chmod +x postfix.SlackBuild
DATABASE=mysql ./postfix.SlackBuild
```

The resulting package will be found in /tmp/postfix-2.11.3-x86_64-1_SBo.tgz (or simillar for 32 bit version).

Copy the package file to the target platform if necessary and install:

```
installpkg {path-to/}postfix-2.11.3-x86_64-1_SBo.tgz
```

Configuring The Postfix MTA

You must become familiar with the [postfix documentation](#) in order to properly configure and administer your mail server. Additionally, a complete set of online documentation for your version is installed from the package to /usr/doc/postfix-2.11.3/html/, learn what is there.

You *must* become familiar with the tools available to you in order to safely admin and maintain your mail server! Take the time necessary to *read the man pages* - man postfix, man postconf, man master at minimum. And become familiar with the resources available through **man postfix:SEE ALSO** section, *before* you start your mail server!

What follows is a minimum configuration to safely operate a virtual email server on the internet, but you will surely want to adapt it to your specific needs.

Postfix has two main configuration files:

- /etc/postfix/master.cf
- /etc/postfix/main.cf

The master.cf file governs operation of the master daemon which listens for mail events and manages the many postfix utility processes.

We want to listen for SMTP, SMTPS and SUBMISSION requests, so…

```
vi /etc/postfix/master.cf

# Uncomment the following lines #
smtp       inet  n       -       n       -       -       smtpd
submission inet  n       -       n       -       -       smtpd
smtps      inet  n       -       n       -       -       smtpd
```

Next, we want to configure the basic network environment for our postfix MTA in /etc/postfix/main.cf. However, the default main.cf is a very large file with many detailed comments and can be error prone to configure and administer. So it it recommended that you save a copy of the original and write the production version from scratch - it really isn't very long.

```
mv /etc/postfix/main.cf /etc/postfix/main.cf.original
vi /etc/postfix/main.cf

# Enter the following lines #

mynetworks_style = host
myorigin = $mydomain

# Change my-domain.com to the actual domain name of your server #
```

```
mydomain = my-domain.com

# mydestination must be localhost only to allow postfix to deliver non-
virtual system mail #
mydestination = localhost

#IMPORTANT - relay_domains should be empty to prevent your server from
becoming a spam relay!
#If you actually need to relay to other domains READ THE DOCUMENTATION
CAREFULLY!
relay_domains =

# We have no local users, so no local notifications #
biff = no
```

Now we need to configure the secure aspects of our mail server, again in main.cf:

```
vi /etc/postfix/main.cf

# Enter the following lines #

# SSL certificates will be created at these locations when we configure
dovecot #
smtpd_tls_cert_file = /etc/ssl/localcerts/dove.pem
smtpd_tls_key_file = /etc/ssl/private/dove.key

#Accept only secure smtp connections
smtpd_use_tls = yes
smtpd_tls_auth_only = yes

#Tell postfix to use dovecot for SASL
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes

#Allow ONLY authenticated users to send email
#Do not accept incoming email for other than local mail boxes (i.e. virtual
users)
smtpd_recipient_restrictions =
permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination
```

Finally, we must configure postfix to use the database to identify allowed domains and to authenticate smtp requests against virtual mail boxes. Again, in main.cf...

```
vi /etc/postfix/main.cf

# Enter the following lines #

#Tell postfix to use dovecot lmtp for virtual mail delivery
virtual_transport = lmtp:unix:private/dovecot-lmtp
```

```
#Set file paths to mysql handlers for domains, mail boxes and aliases
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

#Set localhost aliases path
alias_maps = hash:/etc/aliases
```

The last four files do not yet exist on the system and so, must be created. The files prefixed with
"virtual" tell postfix how to interact with the mysql database. The final alias_maps file tells the postfix
process where to send operational messages.

First we must tell postfix how to identify which virtual domains it handles mail for by providing a
proper query of the virtual database. This query must return true if the domain is in the database:

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

# Enter the following lines #
user= mailuser
password = {your mailuser password}
hosts = localhost
dbname = mailserver
query = SELECT 1 FROM virtual_domains WHERE name='%s'
```

You may test the database connection and query with the postmap utility…

```
postmap -q my-domain.com mysql:/etc/postfix/mysql-virtual-mailbox-domains.cf
```

This will return true (1) if the domain is found, NULL if not found, and an error message if the query or
connection is not correctly configured.

Also note that other database connection options are available, for example if you are connecting to a
remote database server, or multiple servers. See the mysql_table man page from the postfix package
for complete connect parameter information.

Next, we must tell postfix how to identify valid virtual mail box users. This query must also only return
true if the mail box exists in the database.

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

# Enter the following lines #
user= mailuser
password = {your mailuser password}
hosts = localhost
dbname = mailserver
query = SELECT 1 FROM virtual_users WHERE email='%s'
```

You may test the database connection and query with the postmap utility…

```
postmap -q me@my-domain.com mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf
```

Next, we must tell postfix how to identify valid aliases in the database. This query must return the destination address for the alias.

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

# Enter the following lines #
user= mailuser
password = {your mailuser password}
hosts = localhost
dbname = mailserver
query = SELECT destination FROM virtual_aliases WHERE source='%s'
```

You may test the database connection and query with the postmap utility…

```
postmap -q alias@my-domain.com mysql:/etc/postfix/mysql-virtual-alias-maps.cf
```

Finally, we must provide local aliases for postmaster and root mail on the local machine.

```
vi /etc/aliases

# Enter the following lines #
postmaster: root
root: root
```

After creating the aliases file, you must use the newaliases utility to build the local aliases database…

```
newaliases
```

This will create the file /etc/aliases.db in format useable by postfix.

It is important to provide the postmaster alias so that postfix can send operational messages to a real destination. The destination address may be a local user account or it may be a valid virtual mail box.

Return to main article page

Sources

- Based primarily on Postfix documentation
- Originally written by astrogeek

howtos, postfix, dovecot, mysql

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:network_services:postfix_dovecot_mysql:postfix

Last update: **2018/10/08 02:14 (UTC)**