

# Postfix Mail Server

This howto explains how to setup postfix with features such as tls encryption, smtp auth, content filtering, spam protection, virus protection and grey listing. This howto uses postfix, amavisd-new, spam assassin, clamav and sqlgrey. Most of which are in Fedora Extras. Work is also being done to make sure all are in Fedora Extras soon.

## Applicable to Fedora Versions

- Fedora Core 5+

## Requirements

Mail servers rely on port 25 (tcp) to send and receive mail. It is also helpful to have a static IP address, however, it is not *needed* with todays offerings for dynamic DNS services. Some providers don't allow port 25 but this is mainly just restricted for "residential" lines. It is always good to keep Fedora updated and this howto assumes you are updated and running the latest versions for your release.

- Port 25 inbound
- Updated Fedora Core

## Assumptions Made in HowTo

This howto assumes the following:

- Domain name: **example.com**
- Host name: **host.example.com**
- Firewall is already configured to allow port 25:tcp
- IPv4
- Local user account: local-user

## Doing the Work

The first steps are to just get everything installed that will be needed to complete the full howto. At this time this includes a few things that are not in Fedora Extras yet. This howto will be updated as software makes it into Fedora Extras. We will be using mysql server for postgrey. If you don't want postgrey or you want to use a different database backend, either don't install a database server or install your database server of choice. Information about using Postgre SQL will be added at some point.

### 1. Install Needed Software

1. Install most of the needed software from Fedora Extras using yum:

```
yum install postfix mysql-server spamassassin clamav
amavisd-new cyrus-sasl clamav-update sqlgrey
```

## 2. Configure and Test Postfix

1. Do some basic configuration to setup postfix before first starting it. Find the configuration variables and update them. Edit the `/etc/postfix/main.cf` configuration file and make the following changes:

```
mydomain = example.com
myorigin = $mydomain
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain,
localhost, $mydomain
mynetworks_style = host
```

2. Start the server for the first time:

```
/sbin/service postfix start
```

3. Send a test mail to a local user using telnet:

commands are in **bold** reponses are in *italics*

```
telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
220 host.example.com ESMTP Postfix>
EHLO testdomain.com
250-host.example.com
250-PIPELINING
250-SIZE 10240000
250-VERFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
MAIL FROM: <user@testdomain.com>
250 2.1.0 Ok

RCPT TO: <local-user@example.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Subject: Hello local-user
Hey local-user,
I just wanted to send some test mail to you :-)
.
250 2.0.0 Ok: queued as B95C8110064
QUIT
```

4. Check the users mail with the 'mail' command when logged in as the *local-user*:

```
mail
```

If this worked delete the users mail and move on, 'man mail' for more information about the mail command.

### 3. Do Some More Configuration for Postfix

This section start to configure postfix to be more secure. There are some options that are personal preferences of the author and may be changed. They are as follows:

- Mail is stored in \$HOME
- TLS required for sending mail remotely
- Certificates in `/etc/postfix`
- The 'mandant' is `/etc/postfix`

5. Update the `/etc/postfix/main.cf` postfix configuration file and make the following changes:  
1. The following changes are updates:

```
home_mailbox = Maildir/
```

2. These changes are additions to the configuration file and may be added at the end of the file.

```
#TLS - SMTP AUTH
disable_vrfy_command = yes
smtpd_use_tls = yes
smtpd_tls_auth_only = yes
tls_random_source = dev:/dev/urandom
smtpd_tls_cert_file = /etc/postfix/cert.pem
smtpd_tls_key_file = /etc/postfix/key.pem
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
# Add some security
smtpd_recipient_restrictions =
  permit_sasl_authenticated, permit_mynetworks,
  reject_unauth_destination
```

6. Move your certificates to the proper location (`/etc/postfix/cert.pem` and `/etc/postfix/key.pem` respectively) and set proper permissions (600).

If you don't have a certificate already, you may generate a self signed cert with the following commands:

```
cd /etc/postfix
openssl req -new -x509 -nodes -out cert.pem -keyout key.pem
-days 3650
chmod 600 *.pem
```

7. Restart the server:

```
/sbin/service postfix restart
```

8. Try to send test mail to your local-user account both from localhost and a remote server.

This should work. It would also be a good test to make sure that your server will not relay mail so try to send mail to another host using your server. It is recommended to continue to send testing mail with telnet so the maximum amount of information is available to debug what is going wrong. You should notice a new response from the server after you `EHLO`:

```
EHLO testdomain.com
250-host.example.com
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
```

9. Test if TLS is working correctly:

commands are in **bold** responses are in *italics*

```
telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
220 host.example.com ESMTP Postfix
EHLO testdomain.com
250-host.example.com
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
STARTTLS
220 2.0.0 Ready to start TLS
```

If you do not see 'Ready to start TLS', something is wrong with your TLS setup.

## • Test SMTP Auth Using a Standard Mail Client

Use your favorite mail client to test if SMTP auth is working. If TLS is not working, SMTP auth will also not work because this howto forces postfix to use TLS when doing SMTP auth.

1. Start sasl

```
/sbin/service saslauthd start
```

2. Start your favorite email client and send a test message to another server/mail system. Connect to the server with the following settings: In addition to a remote account, you could also send a test message to root, another account or yourself.

- Host: host.example.com
- User: local-user
- Password: local-user's password
- Force TLS for SMTP
- Force SMTP Auth

## • Setup Amavisd-New, Spam Assassin, Clam-AV

Amavisd-new is the content filter that will run the spamassassin and clamav checks. It could also be configured to do other checks and has many other features. Those additional features are outside the scope of this howto and might be added later.

1. Configure amavisd-new. Make the following changes to the `/etc/amavisd/amavisd.conf` file:

*Myhostname* is only needed when the server has not been assigned a FQDN, however, it does not hurt to set the variable; check with the command 'hostname'

```
$mydomain = 'example.com';
$myhostname = 'host.example.com';
```

2. Configure SpamAssassin to do extended checks such as `rbf`, `pyzor`, `razor2`, etc. Make the following changes to the `/etc/mail/spamassassin/local.cf` config file:

```
report_safe 1
use_bayes 1
bayes_auto_learn 1
skip_rbl_checks 0
use_razor2 1
use_dcc 1
use_pyzor 1
whitelist_from *@example.com
```

3. Enable ClamAV to do automatic updates to virus definitions. Make the following changes to `/etc/cvsconf/freshclam`:

Note: The change is to comment out this line.

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

4. Update `/etc/freshclam.conf` to enable automatic updates:

Note: The change is to comment out 'Example'.

```
#Example
```

5. Start everything up:

```
/sbin/service amavisd start  
/sbin/service clamd.amavisd start  
/sbin/service spamassassin start
```

## • Configure Postfix to Use the New Content Filtering System

Postfix needs to be told to use the new content filtering system. A few things need to be changed to enable the new filtering system.

1. Add the following to `/etc/postfix/master.cf`:

```
2. smtp-amavis unix - n - 2 smtp  
-o smtp_data_done_timeout=1200  
-o smtp_send_xforward_command=yes  
-o disable_dns_lookups=yes  
-o max_use=20
```

```
3. 127.0.0.1:10025 inet n - - smtpd  
4. -o content_filter=  
5. -o local_recipient_maps=  
6. -o relay_recipient_maps=  
7. -o smtpd_restriction_classes=  
8. -o smtpd_delay_reject=no  
9. -o smtpd_client_restrictions=permit_mynetworks,reject  
10. -o smtpd_helo_restrictions=  
11. -o smtpd_sender_restrictions=  
12. -o smtpd_recipient_restrictions=permit_mynetworks,reject  
13. -o smtpd_data_restrictions=reject_unauth_pipelining  
14. -o smtpd_end_of_data_restrictions=  
15. -o mynetworks=127.0.0.0/8  
16. -o smtpd_error_sleep_time=0  
17. -o smtpd_soft_error_limit=1001  
18. -o smtpd_hard_error_limit=1000  
19. -o smtpd_client_connection_count_limit=0  
20. -o smtpd_client_connection_rate_limit=0  
-o  
receive_override_options=no_header_body_checks,no_unknown_recipient_checks
```

21. Add the following to the `/etc/postfix/main.cf` config file:

```
content_filter = smtp-amavis:[127.0.0.1]:10024
```

22. Restart postfix to apply the changes:

```
/sbin/service postfix restart
```

## • Setup Grey Listing

Grey listing is an anti-spam technique that is used to thwart spammers from doing drive by spamming. There are two steps to get it working with postfix. Setting up the mysql database and then enabling the checks. You may use any supported database you would like but additional database configurations are outside of the scope of this howto. *Replace sensitive information such as passwords with unique settings.*

1. Setup the mysql database:

This assumes mysql server has not been setup and we are dealing with a fresh configuration. If mysql is already setup, you will need to use the 'p' switch for the mysql commands and there is no reason to set a new mysql root password. Also note, you may use whatever user/database name you want but this will need to be updated in the conf file.

```
/sbin/service mysqld start  
mysql -u root
```

2. This will bring you to the mysql shell where you can add the needed user and database for sqlgrey:

Commands are in **bold** responses are in *italics*

```
mysql> create database sqlgrey;  
Query OK, 1 row affected (0.01 sec)  
mysql> grant all on sqlgrey.* to sqlgrey@localhost identified by  
'mysqlUserPassword';  
Query OK, 0 rows affected (0.01 sec)  
mysql> quit  
Bye
```

3. Set a root password for mysql:

```
mysqladmin -u root password "mysqlRootPassword"
```

4. Configure sqlgrey for the database. Make the following changes to the `/etc/sqlgrey/sqlgrey.conf` config:

```
db_type = mysql  
db_pass = mysqlUserPassword  
admin_mail = server-admin@example.com
```

5. Start the sqlgrey service:

```
/sbin/service sqlgrey start
```

## ● Setup Postfix to Do Grey Listing

Postfix needs to be configured to check the greylisting service for the status of a sender.

1. Configure postfix to do the greylist check. Make the following update to the `/etc/postfix/main.cf` config file:

```
smtpd_recipient_restrictions = permit_sasl_authenticated,  
permit_mynetworks, reject_unauth_destination,  
check_policy_service inet:127.0.0.1:2501
```

2. Restart postfix to apply the changes:

```
/sbin/service postfix restart
```

## ● Set Services to Run on Boot

The combination of services need to get set to run on boot. Do so with 'chkconfig':

```
/sbin/chkconfig postfix on  
/sbin/chkconfig amavisd on  
/sbin/chkconfig clamd.amavisd on  
/sbin/chkconfig spamassassin on  
/sbin/chkconfig mysqld on  
/sbin/chkconfig sqlgrey on
```

# Troubleshooting

## How to Test

Test by sending mail from a remote service/server.

## Common Problems and Fixes

The most common issue is networking issues. Please be sure your networking is setup correctly. For example the below is to allow port 25 tcp using iptables:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 25 -j  
ACCEPT
```