HOWTO

Postfix + OpenLDAP + Courier-imap + SquirrelMail + SASL + TLS

Step-by-Step Guide to Configure a GNU/Linux System to use Postfix, OpenLDAP and Courier-imap

---

Go to top.

Intro

This is a step by step guide on how to configure GNU/Linux system with Postfix, OpenLDAP and Courier-imap. Even though this setup was configured on RHEL 3, with minor tweaks you should get it working on any GNU/Linux distribution.

About Author

Written by Daniel "drade" Radetic from boobah_dot_info.

Words of praise, critique, suggestion .. etc are welcome at address

---

1: Overview of the Software I used to create this setup

Distribution: RedHat Enterprise Linux ES release 3 (Taroon Update 3)

Certified Linux distribution, sold by subscription, delivers continuous value and is certified by top enterprise hardware and software vendors. From the desktop to the datacenter, Enterprise Linux couples the innovation of open source technology and the stability of a true enterprise-class platform.

MTA (Mail Transfer Agent): Postfix 2.0.16-14.RHEL3

Postfix attempts to be fast, easy to administer, and secure, while at the same time being sendmail compatible enough to not upset existing users. Thus, the outside has a sendmail-ish flavor, but the inside is completely different.

Backend Database: OpenLDAP 2.0.27-17

OpenLDAP Software is an open source implementation of the Lightweight Directory Access Protocol.

POP/IMAP Server: Courier-imap 4.0.1-1.3ES

Courier-IMAP is a fast, scalable enterprise IMAP server that uses Maildirs.

SASL (Simple Authentication and Security Layer): cyrus-sasl-2.1.15-8

This approach uses a totally different, IP independent method. Instead of checking the IP of the Mail client and comparing it to a range of permitted IP Addresses, the Mail client authenticates itself providing username and password to the mail server (or sharing a secret with it). These credentials are compared to a source that the mail server has access to and if valid data has been provided, the mail server will permit relaying.

TLS (Transport Layer Security): openssl-0.9.7a-33.12

The TLS protocol provides communications privacy over the Internet. The protocol allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery.

WebMail: squirrelmail-1.4.3-0.e3.1

SquirrelMail is a standards-based webmail package written in PHP4. It includes built-in pure PHP support for the IMAP and SMTP protocols, and all pages render in pure HTML 4.0 (with no JavaScript required) for maximum compatibility across browsers. It has very few requirements and is very easy to configure and install. SquirrelMail has all the functionality you would want from an email client, including strong MIME support, address books, and folder manipulation.

Content Inspection Software Interface: Amavisd-new-2.2.1-1.1.el3.rf

High-performance interface between mailer (MTA) and content checkers: virus scanners, and/or SpamAssassin. It is written in Perl for maintainability, without paying a significant price for speed. It talks to MTA via (E)SMTP or LMTP, or by using helper programs.

Anti UCE (Spam) Software: spamassassin-2.64-2.1.el3.dag

Mail filter to identify spam. It is an intelligent email filter which uses a diverse range of tests to identify unsolicited bulk email, more commonly known as Spam. These tests are applied to email headers and content to classify email using advanced statistical methods. In addition, SpamAssassin has a modular architecture that allows other technologies to be quickly wielded against spam and is designed for easy integration into virtually any email system.

Anti Virus Software: clamav-0.83-1.1.el3.rf

GPL anti-virus toolkit for UNIX. The main purpose of this software is the integration with mail servers (attachment scanning). The package provides a flexible and scalable multi-threaded daemon, a command line scanner, and a tool for automatic updating via Internet. The programs are based on a shared library distributed with the Clam AntiVirus package, which you can use with your own software. Most importantly, the virus database is kept up to date .

Go to top.

2: Building necessary software

I couldnt find courier-imap RPM package in rhel3's standard distribution so i built one myself, and here are steps on how to do it.

Building a new Courier RPM package.

Courier-Imap does not allow 'root' user (it fails miserably) to build the package so you'll have to add one regular user account and create RPM development hierarchy:

```
# useradd test
# passwd test
# /bin/su - test
# mkdir $HOME/rpm
# mkdir $HOME/rpm/SOURCES
# mkdir $HOME/rpm/SPECS
# mkdir $HOME/rpm/BUILD
# mkdir $HOME/rpm/SRPMS
# mkdir $HOME/rpm/RPMS
# mkdir $HOME/rpm/RPMS/i386
# echo "%_topdir    $HOME/rpm" >> $HOME/.rpmmacros
```

Obtain the newest courier-imap tarball from URL listed in Appendix: Package Repositories, save it in /home/test/rpm/SOURCES directory and unpack it.

```
# cp courier-imap-4.0.1.tar.bz2 /home/test/rpm/SOURCES
# cd /home/rpm/SOURCES
# tar xjf courier-imap-4.0.1.tar.bz2
```

Now 'cd' to newly created directory and edit courier-imap.spec using your favorite text editor:

```
# cd courier-imap-4.0.1
# nano courier-imap.spec
```

I like having things in more standard places (eg. /etc) so i changed following lines, and i suggest you to do the same:

```
%define _sysconfdir /etc/courier
%define _mandir /usr/share/man
%define _prefix /etc/courier
%define _localstatedir /var/run

%configure \
        --with-redhat \
        --enable-workaround-for-imap-client-bugs \
        %{?xflags: %{xflags}}

#%{__make} check
```

Once you have finished with modification of the 'courier-imap.spec' file, copy it to /home/test/rpm/SPECS directory and build RPM package (NOTE: You will need to install development packages of the services u wish to include support for, plus other requiremnts (openldap-devel, courier-authlib, courier-authlib-devel), before you start building RPM package for Courier-imap):

```
# rpm -Uvh courier-authlib-*
# rpmbuild -bb courier-imap.spec
```

When compilation process finnishes you should have a new courier-imap RPM package in RPMS/i386 directory, so 'cd' to the RPMS/i386 directory and install new Courier-imap RPM package. But you install courier-imap RPM package download and install 'courier-authlib' packages (courier-authlib-ldap) from URL listed in Appendix: Package Repositories:

```
# rpm -Uvh courier-authlib-*
# cd /home/test/rpm/RPMS/i386
# rpm -Uvh courier-imap-4.0.1-1.3ES.i386.rpm
```

Go to top.

Server Setup

Postfix Setup

Using your favorite text editor, edit 'main.cf' file in /etc/postfix directory and modify or add these lines to it (i recommend you to build main.cf file from scratch):

```
# Some basic Directives
#####################################################
myhostname = host.example.com
mydomain =  example.com
mydestination =  $myhostname, $mydomain, localhost.$mydomain

# Preventing multiple deliveries to the same account
#####################################################
default_destination_concurrency_limit =1
local_destination_concurrency_limit = 1

# Defining Banner for our Mailer
#####################################################
smtpd_banner = $myhostname ESMTP $mail_name (RHEL3)

# Anti UCE ( Spam ) restrictions
#####################################################
smtpd_helo_required = yes
disable_vrfy_command = yes

smtpd_recipient_restrictions =
   reject_invalid_hostname,
   reject_non_fqdn_hostname,
   reject_non_fqdn_sender,
   reject_non_fqdn_recipient,
   reject_unknown_sender_domain,
   reject_unknown_recipient_domain,
   permit_mynetworks,
   reject_unauth_destination,
   reject_rbl_client relays.ordb.org,
   reject_rbl_client opm.blitzed.org,
   reject_rbl_client list.dsbl.org,
   reject_rbl_client sbl.spamhaus.org,
   reject_rbl_client cbl.abuseat.org,
   reject_rbl_client dul.dnsbl.sorbs.net,
   permit
smtpd_data_restrictions =
   reject_unauth_pipelining,
   permit

# OpenLDAP stuff
#################################################
virtual_maps = ldap:ldapaltmail
ldapaltmail_timeout = 10
ldapaltmail_server_host = localhost
ldapaltmail_search_base = ou=Users,dc=example,dc=com
ldapaltmail_server_port = 389
ldapaltmail_domain = hash:/etc/postfix/searchdomains
ldapaltmail_query_filter = (&(mailAlternateAddress=%s)(accountstatus=active))
ldapaltmail_result_attribute = mail
ldapaltmail_special_result_attribute = uniquemember
ldapaltmail_bind = yes
ldapaltmail_bind_dn = cn=readmail,dc=example,dc=com
ldapaltmail_bind_pw = secret

# User mailbox destination
#################################################
home_mailbox = Maildir/
```

That is how 'main.cf' file should look like. Now to test your newly configured MTA execute:

```
# telnet YOUR_IP 25
Trying 192.168.1.105...
Connected to IP_ADDRESS (IP_ADDRESS).
Escape character is '^]'.
s: 220 host.example.com ESMTP Postfix (RHEL3)
c: EHLO example.com
s: 250-senza.samplecompany.org
s: 250-PIPELINING
s: 250-SIZE 10240000
s: 250-ETRN
s: 250-XVERP
s: 250 8BITMIME
c: QUIT
```

Letter 's' means server and it represents what server replies on your commands, letter 'c' are command you need to execute. Wow !!! we made it and our Postfix Server is Up and Ready :D.

OpenLDAP Setup

Using your favorite text editor, edit 'slapd.conf' file in /etc/openldap directory and modify or add these lines to it (i recommend you to build slapd.conf file from scratch):

```
# Defining schemas and schema file locations
#################################################################
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/qmail.schema
include         /etc/openldap/schema/courier.schema
include         /etc/openldap/schema/openldap.schema

schemacheck     on
pidfile         /var/run/slapd.pid
argsfile        /var/run/slapd.args

# Defining our database and admin user + password
#################################################################
database        ldbm
directory       /var/lib/ldap
suffix          "dc=example,dc=com"
rootdn          "cn=manager,dc=example,dc=com"
rootpw           secret

# Indexing for faster queries ( bad indexes can slow things up )
#################################################################
index objectClass,uidNumber,uid eq
index mailAlternateaddress,MailForwardingAddress,mail eq
index givenname,sn,cn pres,eq
defaultaccess read

# Defining Access Control Lists for access to various parts of our
# databse. We can live without ACL's aswell, but some security should
# be in place.
# preventng users from viewing passwords, employee number ... etc
#####################################################################
access to attr=userpassword,clearpassword,ldappassword
```

```
    by anonymous auth
    by self write
    by dn="cn=manager,dc=example,dc=com" write
    by * none

access to attr=accountstatus
    by dn="cn=manager,dc=example,dc=com" write
    by dn="cn=daemon,dc=example,dc=com" read

access to *
    by dn="cn=manager,dc=example,dc=com" write
    by dn="cn=daemon,dc=example,dc=com" read
    by users read
    by self write
    by * read
```

Once you have modified file to look like the one presented above, you will need to copy qmail and courier schema files in /etc/openldap/schema directory (You can obtain schema files from URL Appendix: Package Repositories

Next step in our OpenLDAP backend configuration we need to create hierarchy of our OpenLDAP domain. So using you favorite text editor create file called 'example.ldif; in /etc/openldap directory and add these lines to it:

```
dn: dc=example,dc=com
objectClass: top
objectClass: organization
objectClass: dcObject
dc: example
o: Sample Company Users Group
description: Our nifty example OpenLDAP company

dn: cn=manager,dc=example,dc=com
objectclass: organizationalRole
cn: manager
description: OpenLDAP Administrator ( Manager )

dn: ou=Users,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Users
description: Organizational Unit for our users

dn: cn=readmail,dc=example,dc=com
cn: readmail
sn: readmail
objectClass: top
objectClass: person
objectClass: simpleSecurityObject
description: This account is used to read info from openLDAP database.
userPassword: secret

dn: ou=Group,dc=example,dc=com
ou: Group
objectClass: top
objectClass: organizationalUnit

dn: cn=vmail,ou=Group,dc=example,dc=com
gidNumber: 1001
cn: vmail
objectClass: posixGroup
```

```
objectClass: top
userPassword: secret

dn: cn=wheel,ou=Group,dc=example,dc=com
gidNumber: 1002
cn: wheel
memberUid: someadmin
objectClass: posixGroup
objectClass: top
userPassword: secret
```

Now once you have made the above file, you need to add info from it to you LDAP database and you do it like this:

```
# ldapadd -x -v -w secret -D "cn=manager,dc=example,dc=com" -f example.ldif
```

You should see output scrolling across the screen informing you what entries it is adding to LDAP databse.

Now comes the part where we add a user to the LDAP databse. So once again prepare your typing skills and create a file named 'user.ldif' and add following entries in it:

```
dn: uid=foobar,ou=Users,dc=example,dc=com
cn: Foo Bar
givenName: foobi
sn: bar
uid: foobar
gecos: foobar,,,
mail: foobar@example.com
mailAlternateAddress: fbar@example.com
mailAlternateAddress: foob@example.com
mailAlternateAddress: barfoo@example.com
userPassword: secret
uidNumber: 1020
homeDirectory: /home/foobar
mailMessageStore: /home/foobar/Maildir
gidNumber: 1001
shadowMax: 99999
shadowWarning: 7
shadowLastChange: 12416
loginShell: /bin/bash
ou: Users
o: LDAP Example corp
accountStatus: active
mailQuota: 2480000S
physicalDeliveryOfficeName: Weee 23
employeeNumber: 1
telephoneNumber: 1-800-FOO-BAR
title: What a FooBar Person
homePostalAddress: Foobarella 10
homePhone: 1-800-SPANK-ME
userPassword: notsecret
objectClass: top
objectClass: posixAccount
objectClass: shadowAccount
objectClass: account
objectClass: qmailUser
objectClass: CourierMailAccount
objectClass: person
```

```
objectClass: organizationalPerson
objectClass: inetOrgPerson
```

Now once you have made the above file, you need to add info from it to you LDAP database and you do it like this:

```
# ldapadd -x -v -w secret -D "cn=manager,dc=example,dc=com" -f user.ldif
```

That concludes the part where we configured OpenLDAP to be used as out backend database.

All we need to do is set up system to use OpenLDAP for our users information. RedHat provides CLI tool called 'authconfig' which can be used to setup system authentication from various sources. Other distributions have their own tools, however if your using distribution which does not have tools like that, you will need to edit 'nsswitch.conf' file located in /etc directory and add these changes to it:

```
passwd:         files ldap
group:          files ldap
shadow:         files ldap
hosts:          files dns
networks:       files
protocols:      db files
services:       db files
ethers:         db files
rpc:            db files
netgroup:       nis
```

Addtitional thing that you should do is modify 'login' file in /etc/pam.d directory and modify it or add these entries to it:

```
auth       required     pam_securetty.so
auth       required     pam_stack.so service=system-auth
auth       required     pam_nologin.so
account    required     pam_stack.so service=system-auth
password   required     pam_stack.so service=system-auth
session    required     pam_stack.so service=system-auth
session    optional     pam_console.so
session    required     /lib/security/pam_mkhomedir.so skel=/etc/skel
umask=0022
```

The line with 'pam_mkhomedir.so' tells your system to create users home directory if it doesnt exist and pull settings from skel directory alongside with setting umask for user.

Courier-IMAP

Using your favorite text editor, edit 'imapd' file in /etc/courier directory and modify or add these lines to it (i recommend you to build imapd file from scratch):

```
ADDRESS=0
PORT=143
MAXDAEMONS=40
MAXPERIP=4
IMAPDSTART=YES
```

Using your favorite text editor, edit 'authdaemonrc' file in /etc/authlib directory and modify or add these lines to it:

```
authmodulelist="authldap"
daemons=5
DEBUG_LOGIN=2
```

Using your favorite text editor, edit 'authldaprc' file in /etc/authlib directory and modify or add these lines to it:

```
LDAP_SERVER             127.0.0.1
LDAP_PORT               389
LDAP_BASEDN             ou=Users,dc=example,dc=com
LDAP_BINDDN             cn=readmail,dc=example,dc=com
LDAP_BINDPW             secret
LDAP_TIMEOUT            5
LDAP_AUTHBIND           1
LDAP_MAIL               mail
LDAP_DOMAIN             example.com
LDAP_GLOB_GID           vmail
LDAP_HOMEDIR            homeDirectory
LDAP_MAILDIR            mailMessageStore
LDAP_MAILDIRQUOTA       mailQuota
LDAP_FULLNAME           cn
LDAP_UID                uidNumber
LDAP_GID                gidNumber
LDAP_DEREF              never
LDAP_TLS                0
```

That about covers the Server configuration part, now you need to restart daemons:

```
# /etc/init.d/courier-authlib restart
# /etc/init.d/courier-imap restart
# /etc/init.d/postfix restart
# /etc/init.d/ldap restart
```

First try logging in to system as our LDAP user (it should work as we made it a posix account aswell, home directory should get created by entry we made in 'login' file). Now try sending mail to the user foobar@example.com (i assume that you have set your system authentication to use LDAP):

```
# echo testing-ldap-user | mail -s TEST-MY-LDAP foobar@example.com
```

Configure you MUA (Mail User Agent - eg. Evolution) to recieve mail from imap server (if you configured DNS 'a priori' use hostname, otherwise u can use IP ADDRESS)

Go to top.

Webmail

SquirrelMail

Install SquirrelMail Package.

First we shall need to make SquirrelMail talk to the LDAP server, so 'cd' to the /usr/share/squirrelmail/config directory and run 'config.pl' file:

```
# cd /usr/share/squirrelmail/config
# ./config.pl
```

You should get a menu that looks like this:

```
SquirrelMail Configuration : Read: config.php (1.4.0)
---------------------------------------------------------
Main Menu --
1.  Organization Preferences
2.  Server Settings
3.  Folder Defaults
4.  General Options
5.  Themes
6.  Address Books (LDAP)
7.  Message of the Day (MOTD)
8.  Plugins
9.  Database

D.  Set pre-defined settings for specific IMAP servers

C.  Turn color off
S   Save data
Q   Quit

Command >>
```

Entering 6 as option should bring you to the following screen output:

```
SquirrelMail Configuration : Read: config.php (1.4.0)
---------------------------------------------------------
Address Books (LDAP)
1.  Change Servers
    >  localhost
2.  Use Javascript Address Book Search  : false

R   Return to Main Menu
C.  Turn color off
S   Save data
Q   Quit

Command >>
```

Enter 1 as option and set these:

```
hostname: localhost
base: cn=Users,dc=example,dc=com
binddn: cn=readmail,dc=example,dc=com
bindpw: secret
```

Once it is finnished choose 'S' to save data and 'Q' to Quit.

Now you need to start 'httpd' daemon (apache):

```
# /etc/init.d/httpd restart
```

Now point your favorite web browser to the URL:

```
http://127.0.0.1/webmail
```

Use Login name and password of the LDAP user you earlier added to LDAP

Go to top.

Security

SASL (Simple Authentication and Security Layer)

First install 'cyrus-sasl' packages.

To enable SASL support in postfix we need to edit 'main.cf' file in /etc/postfix directory and modify or add these lines:

```
# Some basic Directives
#####################################################
myhostname = host.example.com
mydomain =  example.com
mydestination =  $myhostname, $mydomain, localhost.$mydomain

# Preventing multiple deliveries to the same account
#####################################################
default_destination_concurrency_limit =1
local_destination_concurrency_limit = 1

# Defining Banner for our Mailer
#####################################################
smtpd_banner = $myhostname ESMTP $mail_name (RHEL3)

# Anti UCE ( Spam ) restrictions
#####################################################
smtpd_helo_required = yes
disable_vrfy_command = yes

# Adding 'permit_sasl_authenticated' here for SASL
#####################################################
smtpd_recipient_restrictions =
   permit_sasl_authenticated,
   reject_invalid_hostname,
   reject_non_fqdn_hostname,
   reject_non_fqdn_sender,
   reject_non_fqdn_recipient,
   reject_unknown_sender_domain,
   reject_unknown_recipient_domain,
   permit_mynetworks,
   reject_unauth_destination,
   reject_rbl_client relays.ordb.org,
   reject_rbl_client opm.blitzed.org,
   reject_rbl_client list.dsbl.org,
   reject_rbl_client sbl.spamhaus.org,
   reject_rbl_client cbl.abuseat.org,
   reject_rbl_client dul.dnsbl.sorbs.net,
   permit
smtpd_data_restrictions =
   reject_unauth_pipelining,
   permit
```

```
# OpenLDAP stuff
##################################################
virtual_maps = ldap:ldapaltmail
ldapaltmail_timeout = 10
ldapaltmail_server_host = localhost
ldapaltmail_search_base = ou=Users,dc=example,dc=com
ldapaltmail_server_port = 389
ldapaltmail_domain = hash:/etc/postfix/searchdomains
ldapaltmail_query_filter = (&(mailAlternateAddress=%s)(accountstatus=active))
ldapaltmail_result_attribute = mail
ldapaltmail_special_result_attribute = uniquemember
ldapaltmail_bind = yes
ldapaltmail_bind_dn = cn=readmail,dc=example,dc=com
ldapaltmail_bind_pw = secret

# User mailbox destination
##################################################
home_mailbox = Maildir/

# Adding SASL Support
##################################################
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
broken_sasl_auth_clients = yes
```

Now you need to restart services (postfix and saslauthd), and test SASL authentication:

```
# /etc/init.d/postfix restart
# /etc/init.d/saslauthd restart
# testsaslauth -u foobar -p secret
0: OK "Success."
```

First you will need to generate Base64 encoded string:

```
# perl -MMIME::Base64 -e 'print encode_base64("foobar\0foobar\0secret");'
Zm9vYmFyAGZvb2JhcgBzZWNyZXQ=
```

Now lets try authenticating to our SMTP server:

```
# telnet IP_ADDRESS 25
Trying IP_ADDRESS...
Connected to IP_ADDRESS (IP_ADDRESS).
Escape character is '^]'.
s: 220 host.example.com ESMTP Postfix (RHEL3)
c: EHLO example.com
s: 250-host.example.com
s: 250-PIPELINING
s: 250-SIZE 10240000
s: 250-ETRN
s: 250-AUTH PLAIN LOGIN GSSAPI
s: 250-AUTH=PLAIN LOGIN GSSAPI
s: 250-XVERP
s: 250 8BITMIME
c: AUTH PLAIN Zm9vYmFyAGZvb2JhcgBzZWNyZXQ=
s: 235 Authentication successful
```

```
c: QUIT
```

Again letter 's' means server and it represents what server replies on your commands, letter 'c' are command you need to execute. If everything went smoothly as described above that means that were using SASL to authenticate to our server.

Open your favorite MUA (Mail User Agent eg. Evolution) and set option where it says that server requires authentication.

TLS (Transport Layer Security)

When we implement this layer it encrypts the communication between two hosts.

Before we proceed you will need to install 'openssl' package.

Once you have installed 'openssl' package 'cd' to /usr/share/ssl directory, edit 'openssl.cnf' file and modify or add these lines (i will include an example below, change it to suit your preferences and needs):

```
countryName_default             = HR
stateOrProvinceName_default     = Zagreb
localityName_default            = Zagreb
0.organizationName_default      = example.com
organizationalUnitName_default  = POSTFIX_HOWTO
commonName_default              = host.example.com
emailAddress_default            = postmaster@example.com
```

Now 'cd' to /usr/share/ssl/misc directory and edit 'CA' file by adding '-nodes' to the section '# create a certificate':

```
-newcert)
    # create a certificate
    $REQ -new -nodes -x509 -keyout newreq.pem -out newreq.pem $DAYS
    RET=$?
    echo "Certificate (and private key) is in newreq.pem"
    ;;
-newreq)
    # create a certificate request
    $REQ -new -nodes -keyout newreq.pem -out newreq.pem $DAYS
    RET=$?
    echo "Request (and private key) is in newreq.pem"
    ;;
```

Now run the 'CA' file and generate new certificates:

```
# ./CA -newca
# ./CA -newreq
# ./CA -sign
```

The default values should be ok, since you have changed 'openssl.cnf' to reflect your needs.

Next step is to copy newly generated certificates to /etc/postfix directory:

```
# cp newcert.pem /etc/postfix/
# cp newreq.pem /etc/postfix/
# cp demoCA/cacert.pem /etc/postfix
```

Now add or modify following lines in your 'main.cf' file in order to add TLS support in it:

```
# Some basic Directives
#####################################################
myhostname = host.example.com
mydomain =  example.com
mydestination =  $myhostname, $mydomain, localhost.$mydomain

# Preventing multiple deliveries to the same account
#####################################################
default_destination_concurrency_limit =1
local_destination_concurrency_limit = 1

# Defining Banner for our Mailer
#####################################################
smtpd_banner = $myhostname ESMTP $mail_name (RHEL3)

# Anti UCE ( Spam ) restrictions
#####################################################
smtpd_helo_required = yes
disable_vrfy_command = yes

smtpd_recipient_restrictions =
   permit_sasl_authenticated,
   reject_invalid_hostname,
   reject_non_fqdn_hostname,
   reject_non_fqdn_sender,
   reject_non_fqdn_recipient,
   reject_unknown_sender_domain,
   reject_unknown_recipient_domain,
   permit_mynetworks,
   reject_unauth_destination,
   reject_rbl_client relays.ordb.org,
   reject_rbl_client opm.blitzed.org,
   reject_rbl_client list.dsbl.org,
   reject_rbl_client sbl.spamhaus.org,
   reject_rbl_client cbl.abuseat.org,
   reject_rbl_client dul.dnsbl.sorbs.net,
   permit
smtpd_data_restrictions =
   reject_unauth_pipelining,
   permit

# OpenLDAP stuff
################################################
virtual_maps = ldap:ldapaltmail
ldapaltmail_timeout = 10
ldapaltmail_server_host = localhost
ldapaltmail_search_base = ou=Users,dc=example,dc=com
ldapaltmail_server_port = 389
ldapaltmail_domain = hash:/etc/postfix/searchdomains
ldapaltmail_query_filter = (&(mailAlternateAddress=%s)(accountstatus=active))
ldapaltmail_result_attribute = mail
ldapaltmail_special_result_attribute = uniquemember
ldapaltmail_bind = yes
ldapaltmail_bind_dn = cn=readmail,dc=example,dc=com
ldapaltmail_bind_pw = secret

# User mailbox destination
################################################
home_mailbox = Maildir/
```

```
# Adding SASL Support
##############################################
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
broken_sasl_auth_clients = yes

# Adding TLS Support
#############################################
smtpd_use_tls = yes
smtpd_tls_auth_only = yes
smtpd_tls_key_file = /etc/postfix/newreq.pem
smtpd_tls_cert_file = /etc/postfix/newcert.pem
smtpd_tls_CAfile = /etc/postfix/cacert.pem
smtpd_tls_loglevel = 3
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
```

Restart postfix service and try starting TLS by connecting to your server with telnet:

```
# /etc/init.d/postfix restart
# telnet IP_ADDRESS 25
Connected to IP_ADDRESS (IP_ADDRESS).
Escape character is '^]'.
s: 220 host.example.com ESMTP Postfix (RHEL3)
c: EHLO example.com
s: 250-host.example.com
s: 250-PIPELINING
s: 250-SIZE 10240000
s: 250-ETRN
s: 250-STARTTLS
s: 250-AUTH PLAIN LOGIN GSSAPI
s: 250-AUTH=PLAIN LOGIN GSSAPI
s: 250-XVERP
s: 250 8BITMIME
c: STARTTLS
s: 220 Ready to start TLS
c: QUIT
```

Letter 's' means server and it represents what server replies on your commands, letter 'c' are command you need to execute.

Use Your favorite MUA (Mail User Agent eg. Evolution) and set the option where it says that the server requires secure connection (SSL).

Go to top.

Content Checking

Amavisd-new

amavisd-new is a high-performance and reliable interface between mailer (MTA) and one or more content checkers: virus scanners, and/or Mail::SpamAssassin Perl

module. It is written in Perl, ensuring high reliability, portability and maintainability. It talks to MTA via (E)SMTP or LMTP protocols, or by using helper programs. No timing gaps exist in the design, which could cause a mail loss.

To configure amavis you need edit 'amavisd.conf' file located in /etc directory and modify or add following lines:

```
$MYHOME = '/var/amavis';
$mydomain = 'example.com';
$daemon_user  = 'amavis';
$daemon_group = 'amavis';
$TEMPBASE = "$MYHOME/tmp";
$DO_SYSLOG = 1;
$LOGFILE = "$MYHOME/amavis.log";
$log_level = 2;
$QUARANTINEDIR = '/var/amavis/quarantine';
$forward_method = 'smtp:127.0.0.1:10025';
$notify_method = 'smtp:127.0.0.1:10025';
$inet_socket_port = 10024;
```

First edit 'main.cf' file and modify or add this line in it:

```
# Some basic Directives
#####################################################
myhostname = host.example.com
mydomain =  example.com
mydestination =  $myhostname, $mydomain, localhost.$mydomain

# Preventing multiple deliveries to the same account
#####################################################
default_destination_concurrency_limit =1
local_destination_concurrency_limit = 1

# Defining Banner for our Mailer
#####################################################
smtpd_banner = $myhostname ESMTP $mail_name (RHEL3)

# Anti UCE ( Spam ) restrictions
#####################################################
smtpd_helo_required = yes
disable_vrfy_command = yes

smtpd_recipient_restrictions =
   permit_sasl_authenticated,
   reject_invalid_hostname,
   reject_non_fqdn_hostname,
   reject_non_fqdn_sender,
   reject_non_fqdn_recipient,
   reject_unknown_sender_domain,
   reject_unknown_recipient_domain,
   permit_mynetworks,
   reject_unauth_destination,
   reject_rbl_client relays.ordb.org,
   reject_rbl_client opm.blitzed.org,
   reject_rbl_client list.dsbl.org,
   reject_rbl_client sbl.spamhaus.org,
   reject_rbl_client cbl.abuseat.org,
   reject_rbl_client dul.dnsbl.sorbs.net,
   permit
smtpd_data_restrictions =
   reject_unauth_pipelining,
```

```
   permit
# OpenLDAP stuff
##################################################
virtual_maps = ldap:ldapaltmail
ldapaltmail_timeout = 10
ldapaltmail_server_host = localhost
ldapaltmail_search_base = ou=Users,dc=example,dc=com
ldapaltmail_server_port = 389
ldapaltmail_domain = hash:/etc/postfix/searchdomains
ldapaltmail_query_filter = (&(mailAlternateAddress=%s)(accountstatus=active))
ldapaltmail_result_attribute = mail
ldapaltmail_special_result_attribute = uniquemember
ldapaltmail_bind = yes
ldapaltmail_bind_dn = cn=readmail,dc=example,dc=com
ldapaltmail_bind_pw = secret

# User mailbox destination
##################################################
home_mailbox = Maildir/


# Adding SASL Support
##################################################
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
broken_sasl_auth_clients = yes

# Adding TLS Support
##################################################
smtpd_use_tls = yes
smtpd_tls_auth_only = yes
smtpd_tls_key_file = /etc/postfix/newreq.pem
smtpd_tls_cert_file = /etc/postfix/newcert.pem
smtpd_tls_CAfile = /etc/postfix/cacert.pem
smtpd_tls_loglevel = 3
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom

# Amavis Section
##################################################
content_filter = smtp-amavis:[127.0.0.1]:10024
```

Now edit 'master.cf' file in /etc/postfix directory and add these lines at the end of the file:

```
smtp-amavis unix        -       -       n       -       2       smtp
        -o smtp_data_done_timeout=1200
        -o disable_dns_lookups=yes
127.0.0.1:10025 inet    n       -       n       -       -       smtpd
         -o content_filter=
        -o local_recipient_maps=
         -o relay_recipient_maps=
        -o smtpd_restriction_classes=
        -o smtpd_client_restrictions=
        -o smtpd_helo_restrictions=
        -o smtpd_sender_restrictions=
        -o smtpd_recipient_restrictions=permit_mynetworks,reject
        -o strict_rfc821_envelopes=yes
```

Now you need to restart amavisd and postfix daemons:

```
# /etc/init.d/amavisd start
# /etc/init.d/postfix restart
```

If you feel like testing use command 'touch' to create a file with '.exe' extension and send that file using your favorite MUA (Mail User Agent eg. Evolution) as attachement, observe the actions by 'tailing' maillog file

```
# tail -f /var/log/maillog
```

Go to top.

SpamAssasin

Spamassassin is perl based tool used to detect spam in mail messages, it is detected by Amavisd-new by default making their integration as seamless as possible.

All you need to do in order to enable SpamAssassin is to install package and run the daemon, of course by default it will detect some spam, however you can always tweak it to your needs and desires.

In order to get the best of spamassassin edit 'amavisd.conf' file in /etc directory and modify or add these lines:

```
$sa_tag_level_deflt  = 0.0;
$sa_tag2_level_deflt = 5.0;
$sa_kill_level_deflt = 5.0;
$mailfrom_notify_admin     = "foobar\@$mydomain";
$mailfrom_notify_recip     = "foobar\@$mydomain";
$mailfrom_notify_spamadmin = "foobar\@$mydomain";
$mailfrom_to_quarantine = 'spam-quarantine';
$warnspamsender = 1;
$spam_quarantine_to = 'foobar@example.com';
```

Now edit 'local.cf' file in /etc/mail/spamassassin directory and modify or add these lines:

```
required_hits  4.2
rewrite_subject  1
subject_tag  *****SPAM*****
report_safe  1
report_header  1
use_bayes  1
auto_learn  1
skip_rbl_checks  0
use_razor2  1
use_dcc  1
use_pyzor  1
ok_languages  all
ok_locales  all
```

Of course this is a sample (working) but it may not suit your need or preferences so browse the manual of spamassassin and amavisd-new for more information.

Restart spamassassin and amavisd-new daemon and enjoy.

```
# /etc/init.d/amavis restart
# /etc/init.d/spamassassin restart
```

Go to top.

ClamAV

Coming in few MINUTES/HOURS

Go to top.

Appendix: Package Repositories

Here you will find all packages required for successfully implementing solution in this tutorial, i will sort it by packages that ive installed.

OpenLDAP stuff.

Schema files - courier and qmail schema

courier.schema
qmail.schema

Courier-IMAP sources.

Courier-IMAP
courier-imap-authlib
courier-imap-authlib-devel
courier-imap-authlib-LDAP
courier-imap-authlib-mySQL

Amavisd-new package and dependecnies.

Amavisd-new RPM Package for RHEL3.

amavisd-new

The Amavisd-new depends on a LOT od perl modules, so instead of direct linking them here i will provide two URL's with RHEL3 packages

RPM-Find-RHEL3-By-Name
RPMS-dag-RHEL3

ClamAV packages.

clamav
clamav-DB

SpamAssasin packages

spamassassin
spamassassin-tools

Go to top.
