Mail server for a VPS: Postfix, Dovecot, Spamassassin, policyd-weight

Full featured mail server with memory footprint small enough even for a VPS, with dovecot, postfix, spamassassin, clamav, policyd-weight with all the configs you need.

UPDATE, 2014-12-18

I've recently wrote an updated version of a similar, better setup[^1].

UPDATE, 2012-02-25

I've updated my system to Dovecot 2, and removed ClamAV from the whole line. I haven't received any virus mails in the last 4 years, also they usually end up as spam, and ClamAV was eating up ~300 MB memory total (50 RAM, 250 swap).

For nearly 5 years, I always used Virtualmin GPL[^2] everywhere I could, because I did not had to configure many features myself, it came with pre-configs and really good backend scripts. But as always, it had a price: memory and CPU usage, what is luxury in the world of VPS[^3]'.

I tried to look for the best solution to handle emails, filtering them for spam and virus, and the only system I came across with was always Amavisd[^4]. amavisd is basically a wrapper for spam and virus filtering: it can simultaneously use more than one for both purpose, and most people say it's a nice program. Unfortunately, I tried to configure it, not just use it, and for me it was hell. I've known that Perl is somewhat evil[^5], but configuring amavisd is a mess at all, so I searched for a way to bypass it.

~~It wasn't easy, but in the depth of the postfix forums, I've found out, that postfix is able to pass the mail to a program than catch the output and pass to another program and so on. The trickiest part was to include ClamAV filtering in the way, because ClamAV does not passes the mail back, so the filtering had to be included into Spamassassin.~~

I've also found, that Virtualmin uses it's configured version of Procmail[^6], which was the most hard to replace. The reason for the replacement was that I haven't find any plugin for Roundcube to manage the server-side filtering of procmail. Since Dovecot has already added a version of Sieve[^7] to it's core combined with Dovecot's Local Delivery Agent[^8], it could replace, or could even be a better solution than Procmail. Also Sieve can be accessed from outside, from, for example from Thunderbird with a plugin.

Probably for the best security Virtualmin used local users for everything. While this is an easy and truly secure solution, for a bit more flexibility I used MySQL as source of data. Per user SpamAssassin could also be stored in MySQL and RoundCube has a plugin for this purpose too.

install & configure mysql server

This is not the topic for a mysql server install, there are plenty of tutorials on this topic.

SQL scheme for our mail server

```sql
--
-- Table structure for table `domains`
--

CREATE TABLE IF NOT EXISTS `domains` (
  `domain` varchar(50) NOT NULL,
  PRIMARY KEY (`domain`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--
-- Table structure for table `forwardings`
--

CREATE TABLE IF NOT EXISTS `forwardings` (
  `source` varchar(80) NOT NULL,
  `destination` text NOT NULL,
  PRIMARY KEY (`source`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--
-- Table structure for table `transport`
--

CREATE TABLE IF NOT EXISTS `transport` (
  `domain` varchar(128) NOT NULL DEFAULT '',
  `transport` varchar(128) NOT NULL DEFAULT '',
  UNIQUE KEY `domain` (`domain`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--
-- Table structure for table `users`
--

CREATE TABLE IF NOT EXISTS `users` (
  `email` varchar(80) NOT NULL,
  `password` varchar(255) NOT NULL,
  PRIMARY KEY (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

the vmail user

We're about to use a system user, named `vmail` for all our purposes. All the mails will be stored under the name of this user, with private rights.

```bash
groupadd --gid 5000 vmail
adduser -uid 5000 -gid 5000 --home /vmail --create-home vmail
```

<h2dovecot I start with Dovecot, because even Postfix will rely on it in the authentication process. Also, Dovecot is going to be the local delivery agent (LDA) in order to use Sieve. POP3 is already out of sight, don't search for it ;)

install dovecot

```bash
apt-get install dovecot-common dovecot-imapd
```

update: config for dovecot 2.X/etc/dovecot/dovecot.conf

```perl
#
# Main
#
disable_plaintext_auth = no

log_timestamp = "%Y-%m-%d %H:%M:%S "
login_greeting = webportfolio.hu

mail_location = maildir:~/Maildir:INDEX=/var/lib/dovecot/index/
%u:CONTROL=/var/lib/dovecot/control/%u
mail_privileged_group = mail

protocols = imap sieve
```

```
ssl_cert =
<summary><ins datetime="2012-02-25T06:27:20+00:00">update:
config for dovecot 1.3</ins>/etc/dovecot/dovecot.conf</summary>
```perl

## Dovecot configuration file
protocols = imap imaps managesieve
disable_plaintext_auth = no

##
## Logging
##

log_timestamp = "%Y-%m-%d %H:%M:%S "
syslog_facility = mail

##
## SSL settings
##
ssl = yes
ssl_cert_file = /etc/ssl/your_domain.crt
ssl_key_file  = /etc/ssl/your_domain.key

##
## Login processes
##
login_process_size = 64
login_processes_count = 4
login_max_processes_count = 32
login_max_connections = 32
login_greeting = hi

##
## Mailbox locations and namespaces
##

mail_location = maildir:~/Maildir:INDEX=/var/lib/dovecot/index/
%u:CONTROL=/var/lib/dovecot/control/%u
mail_privileged_group = mail

##
```
```

```
## IMAP specific settings
##

protocol imap {
    imap_client_workarounds = outlook-idle
}

##
## MANAGESIEVE specific settings
##

protocol managesieve {
    login_executable = /usr/lib/dovecot/managesieve-login
    mail_executable = /usr/lib/dovecot/managesieve
}

##
## LDA specific settings
##

auth_executable = /usr/lib/dovecot/dovecot-auth

protocol lda {
    postmaster_address = root@localhost

    # you going to need to create this file by hand with 0777
rights, I could not make it writeable any other way
    log_path = /var/log/dovecot.log
    info_log_path = /var/log/dovecot.log

    mail_plugins = sieve
}


##
## Authentication processes
##

auth default {
    user = root
```

```
    passdb sql {
        args = /etc/dovecot/dovecot-sql.conf
    }

    userdb static {
        args = uid=5000 gid=5000 home=/vmail/%d/%n
allow_all_users=yes
    }

    socket listen {
        master {
            path = /var/run/dovecot/auth-master
            mode = 0600
            user = vmail
        }

        client {
            path = /var/spool/postfix/private/auth
            mode = 0660
            user = postfix
            group = postfix
        }
    }
}

##
## Plugin settings
##

plugin {
    # the /etc/dovecot/sieve/sieve.default will run _before_
any user defined sieve scripts
    sieve_before = /etc/dovecot/sieve/sieve.default
}
```

/etc/dovecot/dovecot-sql.conf

```perl

driver = mysql
connect = host=127.0.0.1 dbname=mail_mysql_db
```

```
user=mail_mysql_user password=mail_mysql_password
default_pass_scheme = CRYPT
password_query = SELECT email as user, password FROM users
WHERE email='%u';
```

/etc/dovecot/sieve/sieve.default

```perl
require "fileinto";

    if header :contains "X-Spam-Virus" "Yes" {
        fileinto "spam";
        stop;
    }

    if header :contains "X-Spam-Status" "Yes" {
        fileinto "spam";
        stop;
    }
```

postfix

Postfix is the SMTP server; this is the one that communicates with the other mail servers, so it also receives and sends all the messages.

install postfix

```bash
apt-get install postfix postfix-mysql
```

/etc/postfix/main.cf

```apache
smtpd_banner = your_mailserver_domain
biff = no
append_dot_mydomain = no
delay_warning_time = 4h
readme_directory = no

alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases

virtual_alias_domains =
virtual_alias_maps = proxy:mysql:/etc/postfix/mysql-
virtual_forwardings.cf, mysql:/etc/postfix/mysql-
virtual_email2email.cf
virtual_mailbox_domains = proxy:mysql:/etc/postfix/mysql-
virtual_domains.cf
virtual_mailbox_maps = proxy:mysql:/etc/postfix/mysql-
virtual_mailboxes.cf
virtual_mailbox_base = /vmail
virtual_uid_maps = static:5000
virtual_gid_maps = static:5000
virtual_create_maildirsize = yes
virtual_maildir_extended = yes
proxy_read_maps = $local_recipient_maps $mydestination
$virtual_alias_maps $virtual_alias_domains
```

```
$virtual_mailbox_maps $virtual_mailbox_domains
$relay_recipient_maps $relay_domains $canonical_maps
$sender_canonical_maps $recipient_canonical_maps
$relocated_maps $transport_maps $mynetworks
$virtual_mailbox_limit_maps
virtual_transport=dovecot
dovecot_destination_recipient_limit=1

local_recipient_maps = proxy:unix:passwd.byname $alias_maps

smtpd_tls_cert_file = /etc/ssl/your_domain.crt
smtpd_tls_key_file  = /etc/ssl/your_domain.key
smtpd_tls_note_starttls = yes
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/
smtpd_scache

myhostname = your_mailserver_domain
myorigin = your_mailserver_domain
mydestination = your_mailserver_domain your_mailserver_name
localhost.localdomain locahost
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128

mailbox_size_limit = 0
message_size_limit = 52428800
recipient_delimiter = +
inet_interfaces = all

smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_sasl_authenticated_header = yes
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_tls_security_level = may

maximal_queue_lifetime = 1d
queue_run_delay = 500s
minimal_backoff_time = 500s
bounce_queue_lifetime = 1d
```

```
smtpd_helo_required = yes
smtpd_helo_restrictions = permit_mynetworks,
    reject_invalid_hostname,
    permit

smtpd_recipient_restrictions =     permit_mynetworks,
    permit_sasl_authenticated,
    reject_invalid_hostname,
    reject_non_fqdn_hostname,
    reject_non_fqdn_recipient,
    reject_unknown_recipient_domain,
    reject_unauth_pipelining,
    reject_unauth_destination,
    check_policy_service inet:127.0.0.1:12525,
    permit
```

###/etc/postfix/master.cf

```
apache
smtp      inet  n       -       -       -       -       smtpd
smtps     inet  n       -       -       -       -       smtpd
pickup    fifo  n       -       -       60      1       pickup
cleanup   unix  n       -       -       -       0       cleanup
qmgr      fifo  n       -       n       300     1       qmgr
tlsmgr    unix  -       -       -       1000?   1       tlsmgr
rewrite   unix  -       -       -       -       -
trivial-rewrite
bounce    unix  -       -       -       -       0       bounce
defer     unix  -       -       -       -       0       bounce
trace     unix  -       -       -       -       0       bounce
verify    unix  -       -       -       -       1       verify
flush     unix  n       -       -       1000?   0       flush
proxymap  unix  -       -       n       -       -
proxymap
proxywrite unix -       -       n       -       1
proxymap
smtp      unix  -       -       -       -       -       smtp
relay     unix  -       -       -       -       -       smtp
  -o smtp_fallback_relay=
showq     unix  n       -       -       -       -       showq
```

```
error      unix  -       -       -       -       -       error
retry      unix  -       -       -       -       -       error
discard    unix  -       -       -       -       -       discard
local      unix  -       n       n       -       -       local
virtual    unix  -       n       n       -       -       virtual
lmtp       unix  -       -       -       -       -       lmtp
anvil      unix  -       -       -       -       1       anvil
scache     unix  -       -       -       -       1       scache

dovecot    unix  -       n       n       -       -       pipe
  flags=DRhu user=vmail:vmail argv=/usr/bin/spamc -e /usr/lib/
dovecot/deliver -d ${recipient} -f {sender}

maildrop   unix  -       n       n       -       -       pipe
  flags=DRhu user=vmail argv=/usr/bin/maildrop -d ${recipient}

uucp       unix  -       n       n       -       -       pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nexthop!
rmail ($recipient)

ifmail     unix  -       n       n       -       -       pipe
  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop
($recipient)

bsmtp      unix  -       n       n       -       -       pipe

  flags=Fq. user=bsmtp argv=/usr/lib/bsmtp/bsmtp -t$nexthop -
f$sender $recipient

scalemail-backend unix - n       n       -       2       pipe
  flags=R user=scalemail argv=/usr/lib/scalemail/bin/scalemail-
store ${nexthop} ${user} ${extension}

mailman    unix  -       n       n       -       -       pipe
  flags=FR user=list argv=/usr/lib/mailman/bin/postfix-to-
mailman.py
  ${nexthop} ${user}
```

/etc/postfix/mysql-virtual_domains.cf

```apache
user = mail_mysql_user
password = mail_mysql_password
dbname = mail_mysql_db
hosts = 127.0.0.1
query = SELECT domain AS virtual FROM domains WHERE
domain='%s'
```

/etc/postfix/mysql-virtual_email2email.cf

```apache
user = mail_mysql_user
password = mail_mysql_password
dbname = mail_mysql_db
hosts = 127.0.0.1
query = SELECT email FROM users WHERE email='%s
```

/etc/postfix/mysql-virtual_forwardings.cf

```apache
user = mail_mysql_user
password = mail_mysql_password
dbname = mail_mysql_db
hosts = 127.0.0.1
query = SELECT destination FROM forwardings WHERE source='%s'
```

/etc/postfix/mysql-virtual_mailboxes.cf

```apache
user = mail_mysql_user
password = mail_mysql_password
dbname = mail_mysql_db
hosts = 127.0.0.1
query = SELECT
CONCAT(SUBSTRING_INDEX(email,'@',-1),'/',SUBSTRING_INDEX(email,
'@',1),'/') FROM users WHERE email='%s'
```

/etc/postfix/dynamicmaps.cf

```apache
tcp    /usr/lib/postfix/dict_tcp.so        dict_tcp_open
mysql    /usr/lib/postfix/dict_mysql.so        dict_mysql_open
```

/etc/postfix/sasl/smtpd.conf

```apache
pwcheck_method: saslauthd
mech_list: plain login
allow_plaintext: true
auxprop_plugin: mysql
sql_hostnames: 127.0.0.1
sql_user: mail_mysql_user
sql_passwd: mail_mysql_password
sql_database: mail_mysql_db
sql_select: select password from users where email = '%u'
```

policyd-weight

I've already wrote a post about policyd-weight and why it is better than blocklists in postfix config[^9] directly, so this is only my current config file and the install. This config is tweaked at some points but I honestly forgot, what I configured exactly, so here is the full version.

```bash
apt-get install policyd-weight
```

/etc/policyd-weight.conf

```perl
#
------------------------------------------------------------------
-
#  policyd-weight configuration (defaults) Version 0.1.14
beta-17
#
------------------------------------------------------------------
-

# 1 or 0 - don't comment
$DEBUG = 0;

$REJECTMSG = "550 Mail appeared to be SPAM or forged. Ask your
Mail/DNS-Administrator to correct HELO and DNS MX settings or
to get removed from DNSBLs";

$REJECTLEVEL = 4; # Mails with scores which exceed this
REJECTLEVEL will be rejected

# A space separated case-sensitive list of strings on which if
found in the $RET logging-string policyd-weight changes
# its action to $DEFER_ACTION in case.
# of rejects.
# USE WITH CAUTION!
# DEFAULT: "IN_SPAMCOP= BOGUS_MX="
$DEFER_STRING = 'IN_SPAMCOP= BOGUS_MX=';
```

```
# Possible values: DEFER_IF_PERMIT, DEFER_IF_REJECT,
# 4xx response codes. See also access(5)
# DEFAULT: 450
$DEFER_ACTION = '450';

# DEFER mail only up to this level
# scores greater than DEFER_LEVEL will be
# rejected
# DEFAULT: 5
$DEFER_LEVEL  = 5;

$DNSERRMSG = '450 No DNS entries for your MTA, HELO and
Domain. Contact YOUR administrator';

# 1: ON, 0: OFF (default)
# If ON request that ALL clients are only checked against RBLs
$dnsbl_checks_only = 0;

# specify a comma-separated list of regexps
# for client hostnames which shall only be RBL checked.
# This does not work for postfix' "unknown" clients.
# The usage of this should not be the norm and is a tool for
people which like to shoot in their own foot.
# DEFAULT: empty
@dnsbl_checks_only_regexps = (
# qr/[^.]*(exch|smtp|mx|mail).*..*../,
# qr/yahoo.com$/
);


# 1: ON (default), 0: OFF
# When set to ON it logs only RBLs which affect scoring
(positive or negative)
$LOG_BAD_RBL_ONLY  = 1;


## DNSBL settings
@dnsbl_score = (
    # host,hit score, miss score, log name
```

```
    'bl.spamcop.net',3,0,'bl.spamcop.net',
    'cbl.abuseat.org',    3,    0,    'cbl.abuseat.org',
    'dnsbl.njabl.org',    3,    0,    'dnsbl.njabl.org',
    'dnsbl.sorbs.net',    3,    0,    'dnsbl.sorbs.net',
    'zen.spamhaus.org',    3,    0,    'zen.spamhaus.org',
    'pbl.spamhaus.org',    3,    0,    'pbl.spamhaus.org',
    'list.dsbl.org',3,0,'list.dsbl.org',
);

# If Client IP is listed in MORE DNSBLS than this var, it gets
REJECTed immediately
$MAXDNSBLHITS  = 3;

# alternatively, if the score of DNSBLs is ABOVE this level,
reject immediately
$MAXDNSBLSCORE = 9;

$MAXDNSBLMSG = '550 Az levelezoszerveruk IP cime tul sok
spamlistan talahato, kerjuk ellenorizze! / Your MTA is listed
in too many DNSBLs; please check.';

## RHSBL settings
@rhsbl_score = (
    'multi.surbl.org',4,0,'multi.surbl.org',
    'rhsbl.ahbl.org',4,0,'rhsbl.ahbl.org',
    'dsn.rfc-ignorant.org',3.5,0,'dsn.rfc-ignorant.org',
    'postmaster.rfc-ignorant.org', 0.1,0,'postmaster.rfc-
ignorant.org',
    'abuse.rfc-ignorant.org',0.1,0,'abuse.rfc-ignorant.org'
);

# skip a RBL if this RBL had this many continuous errors
$BL_ERROR_SKIP = 2;

# skip a RBL for that many times
$BL_SKIP_RELEASE = 10;

## cache stuff
# must be a directory (add trailing slash)
$LOCKPATH = '/var/run/policyd-weight/';
```

```
# socket path for the cache daemon.
$SPATH = $LOCKPATH.'/polw.sock';

# how many seconds the cache may be idle before starting
maintenance routines
#NOTE: standard maintenance jobs happen regardless of this
setting.
$MAXIDLECACHE = 60;

# after this number of requests do following maintenance jobs:
checking for config changes
$MAINTENANCE_LEVEL = 5;

# negative (i.e. SPAM) result cache settings
####################################

# set to 0 to disable caching for spam results. To this level
the cache will be cleaned.
$CACHESIZE = 2000;

# at this number of entries cleanup takes place
$CACHEMAXSIZE = 4000;

$CACHEREJECTMSG  = '550 temporarily blocked because of
previous errors';

# after NTTL retries the cache entry is deleted
$NTTL = 1;

# client MUST NOT retry within this seconds in order to
decrease TTL counter
$NTIME = 30;

# positve (i.,e. HAM) result cache settings
####################################

# set to 0 to disable caching of HAM. To this number of
entries the cache will be cleaned
$POSCACHESIZE = 1000;

# at this number of entries cleanup takes place
```

```
$POSCACHEMAXSIZE = 2000;

$POSCACHEMSG = 'using cached result';

#after PTTL requests the HAM entry must succeed one time the
RBL checks again
$PTTL = 60;

# after $PTIME in HAM Cache the client must pass one time the
RBL checks again.
#Values must be nonfractal. Accepted time-units: s, m, h, d
$PTIME = '3h';

# The client must pass this time the RBL checks in order to be
listed as hard-HAM
# After this time the client will pass immediately for PTTL
within PTIME
$TEMP_PTIME = '1d';


## DNS settings

# Retries for ONE DNS-Lookup
$DNS_RETRIES = 1;

# Retry-interval for ONE DNS-Lookup
$DNS_RETRY_IVAL  = 5;

# max error count for unresponded queries in a complete policy
query
$MAXDNSERR = 3;

$MAXDNSERRMSG = 'passed - too many local DNS-errors';

# persistent udp connection for DNS queries.
#broken in Net::DNS version 0.51. Works with Net::DNS 0.53;
DEFAULT: off
$PUDP= 0;

# Force the usage of Net::DNS for RBL lookups.
# Normally policyd-weight tries to use a faster RBL lookup
```

```
routine instead of Net::DNS
$USE_NET_DNS  = 0;

# A list of space separated NS IPs
# This overrides resolv.conf settings
# Example: $NS = '1.2.3.4 1.2.3.5';
# DEFAULT: empty
$NS  = '';

# timeout for receiving from cache instance
$IPC_TIMEOUT  = 2;

# If set to 1 policyd-weight closes connections to smtpd
clients in order to avoid too many
#established connections to one policyd-weight child
$TRY_BALANCE  = 0;

# scores for checks, WARNING: they may manipulate eachother
# or be factors for other scores.
#  HIT score, MISS Score
@client_ip_eq_helo_score = (1.5, -1.25 );
@helo_score  = (1.5, -2 );
@helo_score  = (0, -2 );
@helo_from_mx_eq_ip_score= (1.5, -3.1  );
@helo_numeric_score= (2.5,  0 );
@from_match_regex_verified_helo= (1,-2 );
@from_match_regex_unverified_helo = (1.6, -1.5  );
@from_match_regex_failed_helo  = (2.5,  0 );
@helo_seems_dialup = (1.5,  0 );
@failed_helo_seems_dialup= (2, 0 );
@helo_ip_in_client_subnet= (0,-1.2  );
@helo_ip_in_cl16_subnet  = (0,-0.41 );
#@client_seems_dialup_score  = (3.75, 0 );
@client_seems_dialup_score  = (0, 0 );
@from_multiparted  = (1.09, 0 );
@from_anon= (1.17, 0 );
@bogus_mx_score = (2.1,  0 );
@random_sender_score  = (0.25, 0 );
@rhsbl_penalty_score  = (3.1,  0 );
@enforce_dyndns_score = (3, 0 );
```

```
$VERBOSE = 0;

# Switch on or off an additional
# X-policyd-weight: header
# DEFAULT: on
$ADD_X_HEADER  = 1;

# Fallback response in case the weighted check didn't return
any response (should never appear).
$DEFAULT_RESPONSE = 'DUNNO default';

#
# Syslogging options for verbose mode and for fatal errors.
# NOTE: comment out the $syslog_socktype line if syslogging
does not
# work on your system.
#

# inet, unix, stream, console
$syslog_socktype = 'unix';

$syslog_facility = "mail";
$syslog_options  = "pid";
$syslog_priority = "info";
$syslog_ident = "postfix/policyd-weight";



#
# Process Options
#

# User must be a username, no UID
$USER = "polw";

# specify GROUP if necessary
# DEFAULT: empty, will be initialized as $USER
$GROUP = "polw";

# Upper limit if child processes
$MAX_PROC = 5;
```

```
# keep that minimum processes alive
$MIN_PROC = 1;

# The TCP port on which policyd-weight listens for policy
requests from postfix
$TCP_PORT = 12525;

# IP-Address on which policyd-weight will listen for requests.
# You may only list ONE IP here, if you want# to listen on all
IPs you need to say 'all' here.
# Default is '127.0.0.1'.
# You need to restart policyd-weight if you change this.
$BIND_ADDRESS = '127.0.0.1';

# Maximum of client connections policyd-weight accepts
# Default: 1024
$SOMAXCONN = 128;

# how many seconds a child may be idle before it dies.
$CHILDIDLE = 30;

$PIDFILE= "/var/run/policyd-weight.pid";
```

Spamassassin

SpamAssassin in this case means both SpamAssassin itself and ClamAV as well, since we're going to use a plugin to get ClamAV to check for viruses in the spam filtering process as well.

install SpamAssassin ~~and ClamAV~~

```bash
apt-get install spamassassin spamc
```

~~To fire up the ClamAV plugin, Spamassassin~~[^10] ~~already have a Wiki page for.~~

Afterwords

This is the base of my current mail system, and it's doing it's job quite well, but I don't this it would stand as a very large traffic mailserver without some additional tweaks, for example, compiling the spamassassin rules[^11], and so on. That's for another day.

Links

1. https://petermolnar.net/linux-tech-coding/debian-lightweight-mailserver-postfix-dovecot-dspam/
2. http://virtualmin.com/
3. http://cheapvps.co.uk/plans-xen.php
4. http://www.amavis.org/
5. http://www.freesoftwaremagazine.com/files/nodes/3288/strip.jpg
6. http://www.procmail.org/
7. http://wiki2.dovecot.org/Pigeonhole/Sieve
8. http://wiki2.dovecot.org/LDA
9. https://petermolnar.net/changing-to-policyd-weight-from-postfixs-built-in-reject_rbl_client/
10. http://wiki.apache.org/spamassassin/ClamAVPlugin
11. http://spamassassin.apache.org/full/3.2.x/doc/sa-compile.html

Created by Peter Molnar <mail@petermolnar.net>, published at 2012-01-05 10:39 UTC, last modified at 2021-05-11 11:49 UTC , to canonical URL https://petermolnar.net/article/mail-server-for-a-vps-postfixdovecotspamassassinclamavpolicyd-weight/ , licensed under CC-BY-4.0 .