amavisd-new / ClamAV / SpamAssassin
a Mac OS X HOWTO

1. - Introduction and Prerequisites

This article covers installing amavisd-new and ClamAV to provide spam and virus controls to Postfix. This method will provide server-wide spam and virus filters for all incoming and outgoing mail. It's a very robust system and can be easily configured to provide a number of different site-specific options for your users. However, combating unwanted e-mail is an ongoing war and not just a single engagement. This article will describe a good beginning to a full-featured mail system, but you should not stop here.

Examine the Macintosh.tar.gz archive included with amavisd-new and ClamAV for updated files and installation instruction, these files will be updated to correspond with it's released version as required.

The files included in the Macintosh.tar.gz archive and included instructions provide a way to roll the logs and rolled log support for amavis-stats is built in and startup items have been automatically generated for your specific OS version.

You must have the Developer Tools and the Developer SDK's installed.

If you created a separate CLAMAV and/or AMAVISD startup items, please delete them before you begin this installation procedure, the AMAVISCLAMAV or MAILTRANSPORT from any previous ClamAV or amavisd-new source distributions will automatically be modified to support the new installation.

Note this installation is absolutely independent from the one installed by Apple in Mac Os X server 10.4 and some of what you are going to install cannot be configured in any way by using Apple Server Admin Application.

This installation uses the Apple's config files and in this manual you will learn how to configure things.

However, this installation will not be affected by Apple's Software Update and does not overwrite any Apple installed software.

Apple has assigned 82 and 83 as UID/GID for clamav and amavisd users and groups and the default location of the clamav config files is now in "/private/etc/spam/clamav".

As well, some people have been experiencing building issues and this is a result of not selecting the correct default gcc compiler.

So in a terminal session and as root user, before you proceed with the include instructions, issue the following command to ensure that the correct default compiler is selected:

```
# gcc_select 3.3
```

2. - Getting the archives

amavisd-new, ClamAV, db (BerkeleyDB), gmp.

The official URLs for these libraries are:

amavisd-new
http://www.ijs.si/software/amavisd/
ClamAV
http://sourceforge.net/projects/clamav/
BerkeleyDB
http://www.sleepycat.com/download/db/
gmp
ftp://ftp.gnu.org/gnu/gmp/

You no longer need to download the archives (except for amavisd-new) as this process has been automated to make installation easier and is provided for informational purposes only.

Move the BuildSmart application to /usr/sbin, `chown root:wheel` and `chmod 0755` so it has the correct permissions and ownership.

Execute the following command as root user.

```
# BuildSmart
```

This will set up your build environment, fix any user or permission issues and generate proper Startup Items specific to your OS and generate the special Makefiles that will be used to build and install the software.

Once you have successfully completed this installation, to update ClamAV you would issue:

```
# BuildSmart -c
```

and follow the ClamAV building instructions.

(SEE BuildSmart -h for more options)

3. - Building BerkeleyDB (pre 10.4)

Don't install BerkeleyDB using an installer package, build it from source using the following instructions.

```
# cd /SourceCache/SpamAV/BerkeleyDB-1

# make

# make it-work
```

4. - Building gmp

```
# cd /SourceCache/SpamAV/GMP-1

# make

# make install

# make it-work
```

5. - Building ClamAV

```
# cd /SourceCache/SpamAV/ClamAV-1

# make

# make install

# make it-work
```

It doesn't get any easier than this, the software is built and installed, now you need to configure it.

You will notice a directory in your root "DIST_ROOT", please don't browse this with the finder.

"`DIST_ROOT/Release`" hold all of the software and dependancies so you can make an installer package for distribution or to install on other servers you own.

6. - Configuring ClamAV (10.3.x)

Open `/etc/spam/clamav/freshclam.conf` find and make the following changes.
("Example" is an actual line to be deleted or commented out)

```
# Example

UpdateLogFile /var/log/freshclam.log

LogVerbose

PidFile /var/clamav/freshclam.pid

DatabaseOwner clamav

DNSDatabaseInfo current.cvd.clamav.net

DatabaseMirror database.clamav.net

MaxAttempts 5

Checks 24
```

Open `/etc/spam/clamav/clamd.conf` find and make the following changes.
("Example" is an actual line to be deleted or commented out)

```
# Example

LogTime

LogFile /var/log/clamav.log

LogVerbose

PidFile /var/clamav/clamd.pid

LocalSocket /var/clamav/clamd.sock

MaxThreads 20

SelfCheck 1800

User clamav
```

7. - **Installing amavisd-new** (10.3.x) or updating (ANY) (with SpamAssassin)

```
# cd <path-to-amavisd-new>

# cp amavisd.conf-sample /etc/amavisd.conf

# chown root /etc/amavisd.conf

# chmod 0644 /etc/amavisd.conf
```

You need to make the following changes in the amavisd file.

around line 400 change:

```
$daemonize = 1;
```

to:

```
$daemonize = 0;
```

and change (`$DEBUG -> !1`): (your line may look different)

ex.

```
Amavis::Log::init("amavis", $DEBUG, $DO_SYSLOG, $SYSLOG_LEVEL, $LOGFILE);
```

to:

```
Amavis::Log::init("amavis", !1, $DO_SYSLOG, $SYSLOG_LEVEL, $LOGFILE);
```

Now we can install it.

```
# cp amavisd /usr/bin/

# chown root:wheel /usr/bin/amavisd

# chmod 0755 /usr/bin/amavisd
```

Now we need to get some perl modules installed. CPAN makes this easy, but we will have to force one or two of them to go. Also, when you are installing these perl modules you may run across dependencies that you don't have installed yet. Please respond in the affirmative when it asks you if you want them installed but if it asks about aliases for head and other programs, respond in the negative.

If you are already comfortable with CPAN, please skip this section.

If you require a more complex configuration this usually means you are an experienced CPAN user and know what you are doing so no further instructions are necessary for you.

CPAN comes installed by default in your Mac Os X but in the first run it has to be configured properly. From Terminal, prefereable logged in as root (`su`), type:

```
# perl -MCPAN -e shell
```

If you get the CPAN prompt:

```
cpan>
```

It means you have it already configured, read on to make sure you have it configured properly.

If you are unsure about your CPAN configuration you can reconfigure it from scratch your CPAN by typing the following at CPAN prompt:

```
cpan> o conf init
```

Typing '`perl -MCPAN -e shell`' for the very first time in your machine or by typing '`o conf init`' at the '`cpan>`' prompt, you will be carried into the configuration dialog.

Now we are going to make the minimal configuration we'll need and in either cases, you will see the following message, answer "no":

```
/System/Library/Perl/5.8.6/CPAN/Config.pm initialized.


CPAN is the world-wide archive of perl resources. It consists of about
100 sites that all replicate the same contents all around the globe.
Many countries have at least one CPAN site already. The resources
found on CPAN are easily accessible with the CPAN.pm module. If you
want to use CPAN.pm, you have to configure it properly.
```

```
If you do not want to enter a dialog now, you can answer 'no' to this
question and I'll try to autoconfigure. (Note: you can revisit this
dialog anytime later by typing 'o conf init' at the cpan prompt.)

Are you ready for manual configuration? [yes] no
```

and your Terminal window will be filled with a lot of lines showing the resultant configuration by default, you can ignore this.

Now is the time to replace some parameters we need to be sure they are right for present installation:

```
cpan> o conf make_arg /usr
    make_arg           /usr

cpan> o conf makepl_arg /usr
    makepl_arg         /usr

cpan> o conf prerequisites_policy ask
    prerequisites_policy ask
```

To finish with the configuration, we need to add some urls with mirrors of CPAN, use your browser to visit the mirrors list below and copy the urls of some of them near your country in order to achieve the best speed (whilst it is not always true the closest servers are the fastest, so select them as you please :)

http://mirrors.cpan.org/

We add some servers:

```
cpan> o conf urllist http://mirrors.gossamer-threads.com/CPAN

cpan> o conf urllist push ftp://ftp.ri.telefonica-data.net/CPAN

cpan> o conf urllist push ftp://ftp.mednor.net/pub/mirrors/CPAN
```

Last but not least, if you are behind a proxy you will need to tell cpan which is your proxy, to do that you should use the 'o conf ftp_proxy' command:

```
cpan> o conf ftp_proxy your_proxy_here
```

To verify all you did so far, use the 'o conf' command, without arguments to obtain a list of the whole configuration.

Now to save the settings

```
cpan> o conf commit
```

Remember the important settings in present case are:

```
make_arg           /usr
makepl_arg         /usr
prerequisites_policy ask
```

Now you are done with the CPAN configuration and can continue with the installation:

```
# perl -MCPAN -e shell
```

You will then type in the next four commands which will install the modules. Some of these modules may ask if you want to install some dependencies, answer "yes" to this.

```
cpan> install Archive::Tar Archive::Zip Compress::Zlib Convert::UUlib DBI
Unix::Syslog

cpan> install IO::String IO::Stringy Mail::Audit Mail::Internet
Mail::SpamAssassin BerkeleyDB

cpan> install Net::Server Time::HiRes Digest::HMAC Digest::MD5 Digest::SHA1
MIME::Base64 MIME::Parser

cpan> force install Convert::TNEF Net::SMTP
```

Finally exit out of CPAN.

```
cpan> quit
```

If you experience problems building any of the perl modules, you probably have CPAN configured to not follow dependancies and any module that requires another module to be installed will fail to install, reconfigure your perl to prompt for dependent modules rather than ignore them.

You now need to edit your amavisd config file. This file contains a huge number of options that will pretty much determine your spam and virus policies for your server. You should familiarize yourself with this file so that you get the desired results from this system. It's rather well commented so you shouldn't need to mess with it too much.

In Section I you'll need to change

$MYHOME to "/var/amavis"

$mydomain to your main e-mail domain.

$myhostname to your FQDN.

$daemon_user should be set to "clamav"

$daemon_group should be set to "clamav"

$pid_file to "$MYHOME/amavisd.pid"

$lock_file to "$MYHOME/amavisd.lock"

$unix_socketname to "$MYHOME/amavisd.sock"

@local_domains_maps = (1); # Covers all hosted domains

Section II and III you can leave alone.

Section IV will require you to make some decisions. This section determines what happens when an e-mail is determined to be a spam or virus e-mail. Here you can specify the notification templates for what your bounce messages say.
More importantly you can determine what you'll do with spam and virus e-mails.

The final destiny variables are what you are interested in here.

By default amavisd will bounce all spam back to the sender. You may find that this clogs up your mail system attempting to be nice to spammers. If that's the case you can set this to `D_DISCARD` which will effectively delete the mail in question.

You will also want to set your `$virus_admin` and `$spam_admin` settings where the respective notifications will be sent.

The quarantine settings allow you to specify where the spam and virus e-mails will be stored. If you are interested in keeping the e-mails you can direct them to an e-mail address or folder, otherwise you can set these to "undef" which will delete them.

Section V sets up white and black lists for amavis. Use these to add in any domains that you know are good or bad.

Section VI you can leave alone.

Section VII is where you specify when e-mail is tagged as spam, the sa_tag levels determine when to quarantine spam mails and when to kill them and where you would enter any virus scanners. For example, if your using clamd (part of ClamAV), you'll want to uncomment and ammend the clamd section, it should look something like this when done:

```
['Clam Antivirus-clamd',
\&ask_daemon, ["CONTSCAN {}n", "/var/clamav/clamd.sock"],
qr/bOK$/, qr/bFOUND$/,
qr/^.*?: (?!Infected Archive)(.*) FOUND$/ ],
```

Section VIII & IX you can leave alone.

8. - Editing postfix (10.3.x)

Now we edit the Postfix files, first you need to add the following lines to `/etc/postfix/main.cf` it will tell postfix to use amavisd-new as a content filter before delivery.

```
#
# ====================================================
# amavis-new/ClamAV
# ====================================================
#
content_filter=smtp-amavis:[127.0.0.1]:10024
```

Now add the following to `/etc/postfix/master.cf`:

```
#
# ====================================================
# amavis-new/ClamAV
# ====================================================
#
smtp-amavis unix - - y - 2 smtp
     -o smtp_data_done_timeout=1200
     -o smtp_send_xforward_command=yes
     -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 mynetworks=127.0.0.0/8
     -o strict_rfc821_envelopes=yes
     -o smtpd_error_sleep_time=0
     -o smtpd_soft_error_limit=1001
     -o smtpd_hard_error_limit=1000
     -o receive_override_options=no_header_body_checks
```

The grand finally is to restart postfix.

```
# postfix reload
```

9a. - Adding the log Roll item (10.3.x)

First we move the `logroll` directory to `/etc/` and set ownership and permissions.

```
# mv <path_to_logroll> /etc/

# chmod -R 0755 /etc/logroll

# chown -R root:wheel /etc/logroll
```

Now we make the connection so the scripts get execuited.

If /etc/daily.local doesn't exists we first need to create it but if it does exists we can skip to the entry.

```
# touch /etc/daily.local

# echo -en '#!/bin/sh -\n#\n\n' > /etc/daily.local
```

Now we can add our entry to /etc/daily.local.

```
# echo -e "
\tif [ -f /etc/logroll/700.server-spamav ]; then
\t\techo \"\"
\t\techo \"Running Additional log roll scripts:\"
\t\tsh /etc/logroll/700.server-spamav
\tfi\n
" >> /etc/daily.local
```

9b. - Fixing the log Roll item (10.4.x)

In `/etc/periodic/daily/700.daily.server.cyrus` make the following changes:

```
%logsAndProcesses = ("/var/log/mailaccess.log","syslog",
                        "/var/log/amavis.log","amavisd",
                        "/var/log/clamav.log","clamd",
                        "/var/log/freshclam.log","freshclam",
                        "/var/mailman/logs/*","python.*mailmanctl");
```

and: `(in "sub roll")`

```
     $return = rename("$filename", "$filename.0" );
     if ($return) {
          `/usr/bin/gzip "$filename.0"`;
          log_message("Notice: Renamed and compressed\"$filename\" -->
\"$filename.0.gz\".");
          if ($processPattern eq "amavisd") {
               `gzcat $filename.0.gz > $filename.0`;
               chmod($mode, $logFilePath.0);
               chown($uid, $gid, $logFilePath.0);
               log_message("Notice: Fixed rolled file for amavis-stats
\"$filename.0.gz\" --> \"$filename.0\"");
          }
```

(the log_message lines are wrapped in this document)

Notes:

See 'Section VII' of Installing amavisd-new to enable support for clamd in 10.3.x - 10.4.x.

Additional patch files may be included in the Macintosh archive 'Macintosh.tar.gz' for advanced/modified features, please see included 'README' for related information.

A new helper program was installed by BuildSmart that can be used to manually stop, start and reload amavisd-new, clamd and freshclam with a single command.