

AEGEE-milter 1.0 Manual, updated 4 September 2019

Dilyan Palauzov

This manual is for aegee-milter 1.0

Copyright © 2007-2019 Free Software Foundation, Inc.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with the Invariant Sections being “Free Software” and “Free Software Needs Free Documentation”, with the Front-Cover Texts being “A GNU Manual,” and with the Back-Cover Texts as in (a) below.

(a) The FSF’s Back-Cover Text is: “You are free to copy and modify this GNU Manual. Buying copies from GNU Press supports the FSF in developing GNU and promoting software freedom.”

Table of Contents

1	Introduction	1
2	Objectives	1
3	Circumventing the SMTP Limitations	2
3.1	One recipient per transaction. defer for other recipients	2
3.2	One recipient per transaction. faked defer for other recipients ...	3
3.3	Rejection message for partial delivery and partial non-delivery ...	4
3.4	Send Non Delivery Reports	4
3.5	A combination of all approaches	4
3.6	Use PRDR	5
3.7	Do nothing	5
3.8	Limitations of the different nodes	5
4	Concepts – Core – Modules – Lists	5
5	Installation	6
6	Configuration	7
6.1	Connecting aegee-milter with the MTA	8
6.1.1	Order of Milters	8
6.2	Configuration: Section General	8
6.3	Configuration: Section Milter	8
6.4	List Log	8
6.5	List Memcached	8
6.6	List Timsieved Scripts	8
6.7	Module ARF	9
6.8	Module block_sender	9
6.9	Module delay	9
6.10	Module equal	9
6.11	Module relayed	10
6.12	Module sieve	10
6.13	Module sieve_global	10
6.14	Module via	10
7	Invoking aegee-milter	10
8	Bugs and Community	10
9	Roadmap	11

10	Development	12
10.1	Developing the Core.....	12
10.2	The build System	12
10.3	Developing Modules	13
10.4	Developing Lists.....	13
11	History	14
Appendix A GNU GENERAL		
	PUBLIC LICENSE.....	14
Appendix B GNU Free Documentation License ..		
		25
	Index.....	33

1 Introduction

Emails are exchanged between servers using the Simple Mail Transport Protocol. With SMTP the client first communicates which are the recipients of a message. The server tells for each recipient, whether it accepts emails for that destination. Then the client sends the content of the message, and the server answers, whether the message is accepted for all previously accepted recipients or for none of them.

With this workflow it is not possible for a multi-recipient SMTP transaction to accept an email for some recipients and reject it for others. Some users might want to reject whatever suspicious emails they receive, rather than storing them in their Junk folder. Other users might want to have the suspicious mails in their Junk folders. And a third group of users might want to deliver the message in their Junk folder, but draw the attention of the sender and, by rejecting the message at SMTP level, communicate to the sender that the message was delivered in the Junk folder. When a mail is sent to more than one of the above groups, what can the MTA server do to satisfy all wishes?

SMTP makes such use cases hard to handle. aegee-milter tries to help.

To the critics, stating that there is a prize for solving the shortcomings of SMTP, the answer is that one has to start somewhere, e.g. with email segmentation.

For understanding this manual the reader is expected to have general knowledge about the Simple Mail Transfer Protocol. The manual assumes that the reader knows the concepts of SMTP transaction, distinction between SMTP client and SMTP server, reply code, enhanced status code. The terms junk and spam are used interchangeably.

2 Objectives

The objectives of aegee-milter are to:

- save time of humans, spent processing data, at the cost of either delayed mail delivery, double delivery or rejection messages for partially delivered emails and partially undelivered emails.
- solve the problem with legitimate emails being put in the Junk folder and are not read in time.
- have a mail server with good IP reputation, which is achieved by reducing the amount of sent bounces.
- use Sieve for deciding whether to accept or reject a message at SMTP level.

Sometimes mailing list subscribers post with email addresses, they are not subscribed to the list. The mailing list manager can defer the decision on whether to accept the email to the mailing list owner, or bounce the message back to the sender, telling her that she cannot write to that address. Bouncing messages has the risk of blacklisting the mail server in some backscatters DNSBLs. Delegating the decision to the mailing list owner steals her time, forcing her to handle also spam messages, and does not scale. aegee-milter offers to reject at SMTP level such messages and communicate in this way to the sender what is going on. With this approach the time of the mailing list owner is saved, as the work is shifted to the sender.

Anti-spam software classifies sometimes legitimate emails as spam. If this happens, the messages are dispositioned in the Junk folder and neither the sender nor the recipients are aware, that the mail is missing. `aegee-milter` offers to reject the messages, telling the sender why the message was evaluated as spam, and possibly inserting in the rejection message how the recipient can be contacted (snail mail, phone, fax etc). This saves the recipients the time usually spent digging in the Spam folder, and the consequences of reading a mail much later than expected.¹ A further option is to deliver the message as Junk to the recipient, but use rejection at SMTP level, telling the sender that the mail was delivered as spam, why it was evaluated as spam, and what are the alternatives to contact the recipient.

Imperfect anti-spam software makes errors and senders can ask why their email was evaluated as spam. When using anti-spam software from the public domain, the outcome of the anti-spam evaluation is not a secret. Putting the output of the anti-spam software in the SMTP rejection reason, when the email is rejected due high spam probability, saves to the site administrator questions, why a specific email was evaluated as spam.

Sieve is a standard language for filtering emails RFC 5228 (<https://tools.ietf.org/html/rfc5228>). There is software that can upload and download Sieve scripts to a server using the ManageSieve protocol RFC 5804 (<https://tools.ietf.org/html/rfc5804>). With Sieve, the user can inspect the headers a message has, in particular the amount of scores assigned by the anti-spam software, and based on them to decide to accept or reject the message. When a message is rejected, the user can specify the text to be returned to the sender. This text can include the telephone or fax number of the recipient and its snail mail address. In this way, when the anti-spam software classifies a legitimate email as spam, the communication flow is not interrupted by silently delivering messages as Junk.

3 Circumventing the SMTP Limitations

There are some approaches possible to circumvent the limitation of SMTP, that a multi-recipient message has to be either accepted or rejected for all recipients. All approaches have drawbacks, but using none of them also has drawbacks.

3.1 One Recipient per Transaction. Defer for other Recipients

With this approach on each SMTP transaction the server accepts the email for the first recipient and defers it for all other recipients, that have different opinion on whether to accept the message from the first recipient. Deferring is done with a "451 4.5.3 Use PRDR or retry immediately, possibly in the same TCP connection, after DATA End of data; retry=0" message.

The 451 reply code means, that there is a temporary error and the client shall retry later.

The 4.5.3 Enhanced Status Code means more recipients were specified for the message than could have been delivered by the protocol. This error should normally result in the

¹ With `aegee-milter` mails are sometimes delivered delayed. The assumption here is, that the disadvantages imposed by reading post-factum a message wrongly delivered in the spam folder are higher than the delay imposed by segmenting a message, or the risk of double delivery.

segmentation of the message into two, the remainder of the recipients to be delivered on a subsequent delivery attempt. [RFC3463, Section 3.6]. RFC 5248 adds, that 4.5.3 can only be used with ESC 451.

Note, that RFC 821 defines 452 reply code, as "Insufficient storage", which RFC 5821 repeats, but adds, that 452 can also mean "Too many recipients" and that a conforming server should be able to accept at least 100 recipients per iteration before returning 452.

The ideal answer would be "452 4.5.3", stating twice the same thing, but this combination is not allowed per RFC 5248. If 452 is interpreted as "Insufficient storage now, after some time the server might have more place", then the client will impose some delay. On the other side, with "451 4.5.3" there are no indications for the client to impose a particular delay on the retry.

When the SMTP client retries, the message has one recipient less than the initial message. With this approach, a multi recipient message is split (segmented) into one message per recipient and on each iteration the message for the currently handled recipient can be accepted or rejected.

The advantage of this approach is, that the recipients preferences are matched exactly. As a side effect, if a deferred message arrives the second time, the anti-spam software can evaluate it the second time as likely spam, while the earlier evaluation of the same message, could have classified the message as unlikely spam.

The disadvantage of this approach is, that it delays the delivery of the message for all recipients but the first one. A message with many recipients can be delayed so much, that the SMTP client gives up retrying for some recipients.

The delayed delivery is not mitigated by inserting more than two IP addresses accepting SMTP connections. In its default configuration, in order to preserve resources, Postfix does try to deliver emails to up to two different IP addresses, and after the second try falls asleep. Courier MTA does try to deliver emails only up to one different IP address, before falling asleep.

The `retry=0` hint means, that the client shall retry immediately. This is specified in SMTP Service Extension for Greylisting Operations (<https://tools.ietf.org/html/draft-santos-smtpgrey>). However, the specification does not say what to do with zero behind `retry=` and it is not known if there is some significant software implementing that internet draft. Moreover, the hint is defined to be only useful with the GREYLIST service extension and aegee-milter does not take care to announce support for it. Nevertheless adding `retry=0` does not harm.

3.2 One Recipient per Transaction. Faked defer for other Recipients

Like `<undefined>` [One Recipient per Transaction. Defer for other Recipients], page `<undefined>`, but delivers the messages promptly to the recipients, that want to accept the message, and remembers, the result of each action. When the SMTP client retries, the next recipient is accepted in the RCPT TO: phase, together with all recipients that share the same opinion on accepting a message with that recipient. At the end of DATA, the SMTP server communicates to the client the outcome of the iteration, but does not perform any delivery. Remember, the action was performed on the first iteration.

On the first iteration, the SMTP server remembers the Message-Id, the recipients and the actions per recipient.

The advantage of this approach is, that compared to no faked deferring, it does not cause delivery delay to the recipients.

The disadvantages are, that the sender might get a notification, that the delivery of her message was delayed, while in fact the message was delivered. Moreover, on retrying some SMTP clients change the Message-Id, so suppressing the action on a retry will not work and the receivers will get the same message more than once.

A disadvantage compared to the not-faked deferral is that during a later retry, the anti spam software can consider a mail as spam, while it was not considered as spam at the first try. To mitigate this disadvantage, the final mail store can perform anti-spam evaluation periodically after the delivery and mark messages as spam, even if they were not considered as spam at the time of delivery. However, such delayed reevaluation can be performed only until the user checks its inbox or spam mailbox, as subsequent moving of emails between the two mailboxes will confuse the users.

3.3 Rejection message for Partial Delivery and Partial non-Delivery

An email for a recipient is either ordinary delivered, delivered as Junk or rejected. A recipient can decide whether to communicate to the sender, that the message was just delivered, or delivered as Junk. When a mail is directed to many recipients and the recipients have different opinion on whether they accept the message, the server can at the end of DATA return a message, to which recipients the message was ordinary delivered, to which recipients the message was rejected and to which recipients the mail was delivered as junk. With the latter part, of course, taking the preference of the user to communicate to the sender, that the message was delivered as spam.

The advantage of this approach is, that the preferences of the users are enacted exactly and there are no delays.

A possible disadvantage is, that the rejection message contains, for the senders unusual, different statements about the disposition for each recipient. The senders must be able to understand the language, in which the rejection message was written.

When an email from a mailing list manager is sent to many recipients and the rejection message spells that some recipients have and other have not received the message, the mailing list manager will interpret only the number codes and conclude, that the message was not delivered to any recipient. On repeated non-delivery retries, the mailing list manager may remove all recipients from the mailing list.

3.4 Send Non Delivery Reports

It is possible, that the MTA accepts initially the email for all recipients and delivers later non-delivery reports from the recipients, not willing to accept the email.

The disadvantage is, that non-delivery reports (bounces) can blacklist the IP address of the server. However, when aegee-milter is used with the MSA, the NDR are send only to local users and there are no bad side effects.

3.5 A combination of all Approaches

In theory, it is possible to combine different approaches for the same message: do some segmentation with deferred delays, some segmentation with faked delay, and some partial return replies.

aegee-milter does not offer such mixed mode of operation, as the author has no use-case for it.

3.6 Use PRDR

The Per Recipient Delivery Response SMTP Service Extension borrows a concept from LMTP (Local Mail Transfer Protocol). LMTP is used to deliver mails on the receiving server to the mailboxes. It assumes, that the server for the mailboxes cannot send bouces and, e.g. due quota limitations, does accept the message for some recipients of the mail transaction and reject it for others. The communication, which recipients receive and reject the message is done after the message body is transmitted. The specitication is in the doc/directory.

The disadvantage is, that nearly no software implements PRDR. When PRDR is available, aegee-milter does make use of it.

3.7 Do nothing

Without doing nothing, the email workflow stays as you know it. The disadvantage is, that the objectives aegee-milter has are not met.

3.8 Limitations of the different Modes

An MTA listens to port 25 and accepts connections from other MTAs. The assumption is, that these MTAs do retry.

A program accepting emails from the user, MSA, listens to a different port. The common submitting mail user agent does not automatically retry to resend a message on temporary failure, but requires interaction from the user in such cases. When aegee-milter is applied in the MSA, then the Rejection message for partial delivery and partial non-delivery or "Send Non Delivery Reports" mode are the only one that can work, as they do not rely on retrying. In the first mode, the errors are communitated during the SMTP dialog to the mail user agent, in the second mode, the errors are communicated as Non Delivery Report.

If a user installs a Sieve script, that is supposed to be executed during SMTP delivery from outside, it will be surprizing for the Sieve script owner to figure out, that the Sieve script was not called for mails delivered over the MSA. To match the expectations of the users, aegee-milter shall be run also in the MSA.

4 Concepts – Core – Modules – Lists

aegee-milter consists of "core", "modules" and "lists".

The modules say per recipient how to act. They are queried at the RCPT TO:, end of headers and end of data whetherto accept the message for the recipient, or to reject it, and what message to use when rejecting the message. Example for modules is mod_delay, which can throttle SMTP connections, or mod_sieve to execute Sieve scripts per recipient at SMTP Level.

The core glues aegee-milter to the MTA and asks each module, how to act. Its role is to map the per recipient decisions of the modules, to the SMTP protocol, depending on the preferences of the site administrator.

The lists are data sources or sinks. The core and the modules can read or write data to a list, by knowing its name. E.g. to log data, core send data to a list called log. Modules rely on the presence of lists with specific name. That is, if a module is programmed to put or get data in a list with particular name and this list is not configured to be loaded, then aegee-milter crashes at the moment, when data for that list is inserted or queried.

More on the core, modules and lists is explained in the Developer's documentation. As user, the understanding of these concepts is necessary for doing the right configuration.

The order in which modules are executed is significant. The per module documentation explains the rationale of the built-in order. aegee-milter's is built by default with an module execution order, that makes sense.

5 Installation

aegee-milter is distributed by its author in source code. To install the software the manual describes the specifics on compiling aegee-milter. The reader is supposed to have general knowledge on compiling software with autoconf.

The dependencies are both build-time and run-time dependencies: at build time, the corresponding headers must be present.

Dependencies

- (required) libmilter to connect aegee-milter with the MTA (sendmail or postfix). libmilter is part of sendmail.
- (required) Glib (<https://developer.gnome.org/glib/stable/>) for reading the configuration file
- (optional) liblistserv (<https://mail.aegee.org/software/liblistserv/>), memcached (<http://memcached.org/>) and libmemcached (<https://libmemcached.org/libMemcached.html>) to remove subscribers from listserv mailing lists on received ARF report, needed by the ARF module
- (optional) Cyrus Imap (<https://www.cyrusimap.org/>) to execute Sieve scripts and for the ARF module. Only version 3.0 is tested: 3.1 was never tested and 2.5 has worked good at some moment in the past without known problems.

Patches in contrib/

- Sendmail does some validations on the milter protocol, which validations do not permit to execute PRDR. A patch lifts these limitations.

- Cyrus Imap does not install some headers files in $\$(prefix)/include$, necessary to compile aegee-milter: `lib/libconfig.h`, `lib/prot.h`, `lib/ptrarray.h`, `lib/util.h`, `lib/byteorder64.h`, `imap/conversations.h`, `imap/mailbox.h`, `imap/message.h`, `imap/message_guid.h`, `imap/quota.h`, `imap/sequence.h`, `sieve/grammar.h`, `sieve/varlist.h`.
- Move `struct conversations_open *open_conversations` from `conversation.h` to `conversations.c`. This patch is included in Cyrus Imap 3.0.12.
- RFC 5429 Sieve Email Filtering: Reject and Extended Reject Extensions (<https://tools.ietf.org/html/rfc5429>) does not recommend to permit using `ereject` or `reject` with the actions "keep", "fileinto" and "redirect".

6 Configuration

On start aegee-milter reads its configuration file. The configuration file is called `aegee-milter.ini`. The file must be located either under `$sysconffdir` during installation, or passed to aegee-milter with the `-c` parameter.

The format of the file is INI-style. It is parsed according to glib's rules (<https://developer.gnome.org/glib/stable/glib-Key-value-file-parser.html>). The section names and the keys within a section are case-sensitive.

A module or a list is loaded at execution time, if there is a section in the configuration file with the name of the module or the list. Sections may be empty. For configuring each module or list there is a separate section. Sections referring non-existent modules or lists are ignored.

The core needs two sections, called General and Milter.

An example for a INI file

```
[General]
bounce-mode=delayed
pidfile=/var/run/aegee-milter.pid
expire=1
sendmail=/usr/local/sbin/sendmail -C /etc/mail/sendmail/sendmail.cf

[Milter]
socket=local:/var/run/aegee-milter.sock
timeout =3600

[mod_arf]
[mod_sieve_global]
[mod_sieve]
[mod_via]
[mod_relayed]
[mod_rewrite_from]
[mod_equal]
[mod_delay]
ehlo=1
data=1
mail=1
```

```

rcpt=1
[list_timsieved_scripts]
[list_memcached]
[list_log]
file=/var/log/aegee-milter.log
[list_listserv]
#email=xxx
#password=yyyy
#host=localhost

```

The details of the `list_` and `mod_` sections are explained in the documentation of the respective list/module sections below.

6.1 Connecting aegee-milter with the MTA

Milters can be configured either with sendmail or with postfix.

aegee-milter needs the macros `j` at the connect stage and the `i` macro on `envfrom`.

The configuration of a milter with the MTA is for aegee-milter not different, than for any other milter. See bullet [Configure Your MTA](#) for an example. Consult the manual of the MTA.

6.1.1 Order of Milters

It makes sense to first check the incoming emails for viruses. If emails containing viruses are reject, processing of a message can stop here.

The spam evaluation shall happen before aegee-milter is run. The idea is that the anti-spam software inserts in headers the result of the spam-evaluation, the Sieve module of aegee-milter evaluates the inserted headers and based on them takes decisions.

It is recommended to run aegee-milter after the DMARC evaluation was performed. aegee-milter can redirect messages and applying the DMARC policy (reject) after a message failing DMARC was redirected makes things more complicated.

6.2 Configuration: Section General

This section specifies global options. The section can be skipped.

- `pidfile` If present, aegee-milter writes in this file its Process ID.
- `sendmail` How to invoke sendmail. The default is `/usr/bin/sendmail`.
- `expire` (Currently unused). How often the lists shall expunge outdated information.

6.3 Configuration: Section Milter

This section specifies the milter interface.

- `socket` The socket where aegee-milter shall connect with the MTA.
- `timeout` (optional) Sets the number of seconds aegee-milter will wait for an MTA communication (read or write) before timing out. The default timeout is 7210 seconds.

6.4 List Log

The list is loaded, when the configuration file contains section [list_log]. The section may contain the key "file". If "file" is present, on loading list_log, the file is opened and a marker is inserted. On proper shut down, another marker is inserted. The modules send data to list_log and the latter fills the data to the file.

On the roadmap is to add ability to list_log to add data to syslog and journald.

6.5 List Memcached

The list is loaded, when the configuration file contains section [list_memcached].

With `params` the parameters passed to initialize libmemcached. The default is

```
--SERVER=localhost --HASH-WITH-NAMESPACE --NAMESPACE=aegee-milter
--TCP-NODELAY --TCP-KEEPALIVE --BINARY-PROTOCOL.
```

6.6 List Timsieved Scripts

The list is loaded, when the configuration file contains section [list_timsieved_scripts].

For a domain, user, and script name, this list determines the location on the file system, where the Sieve script is located.

The only option in the section is `imapd.conf`. If present, it specifies from which file to load the default Cyrus Imap configuration. When absent, the usual Cyrus Imap mechanisms are used to find the configuration file.

`list_timsieved_scripts` reads from `imapd.conf` the variables `sievedir`, `virtdomains`, `defaultdomain`, and `fulldirhash`. How the paths are calculated is best seen by looking at the source code. As a rule of thumb, if timsieved uploads a sieve script for a user and timsieved and aegee-milter share the same `imapd.conf` file, then aegee-milter will find the file uploaded by aegee-milter.

`list_timsieved_scripts` looks for scripts per recipient based on the RCPT TO: parameter.

For addresses without domain, the `defaultdomain` applies.

In a mail system there are different forms of equivalence between two addresses. The MTA has an alias system, and virtual domains, that are used to determine if two addresses deliver emails to the same destination. The authentication system can consider also two distinct user names as equivalent.

aegee-milter knows nothing about the aliasas in the MTA or the authentication system. If two domains are supposed to be handled as equivalent by aegee-milter, then either there must be hard or symbolic links on the file system, so that aegee-milter can find the right file on the file system. Likewise, if a person has two addresses and the same Sieve filtering rules shall be applied to emails to any of the addresses, the file system must have on the right, distinct place the Sieve scripts for both addresses. Think of two uploads over timsieved for the one and for the other address.

Besides personal scripts, there are also global scripts per domain. If a global script is searched for a particular domain and is missing, the global scripts of the default domain are not retried.

A final hack: in terms of ManageSieve there is a default script per user. `timsieved` can be used change the default (active) script. These scripts can be executed by two different systems: by `aegee-milter` at SMTP time, which has the capability to reject messages at SMTP level, but cannot call `fileinto`, and by Cyrus Imap as MDA which cannot perform reject at SMTP level, but can do `fileinto`. To permit of both systems to coexists, when `list_timsieved_scripts` loads the default script for a recipient, it first check if a file `default.siv.bc` exists. That is, if for the user over ManageSieve a file called `default.siv` was uploaded. If this file exists, then `aegee-milter` uses it in place of the default script. Otherwise `list_timsieved_milter` delivers the name of the default (active) Sieve script.

6.7 Module ARF

The module is loaded, when the configuration file contains section `[mod_arf]`.

This modules parses emails received at XXX and sent from the Yahoo, Microsoft or Seznam.cz feedback loop service. It extracts from them the email addresses, which complained about received spam, the email addresse, which was used to send the message. If the latter is a mailing list, `mod_arf` checks if the complaining address is subscribed to the mailing list. If it is, it removes the address from the mailing list, and sends notifications to both the removed address, and the list-owners. If the mailing list is a sub-list of another list, the email address is removed also from the super list and the notification is sent to the list owners of the super list, too.

If the complaining address is not subscribed to the mailing list or the sending address is not a mailing list, then an email notification is sent to the local administrators.

Once a combination sender-address/complaining address is handled, `mod_arf` inserts a record via `list_memcached` and whenever the same combination appears within a month, no more messages are sent. This is useful, if for the same mailing list a recipient marks more than one messages in a row as spam. Then the FBL service sends several notifications and only the first of it is relevant.

Note, `mod_arf` works only with local listserv mailing lists and the sent messages are customized for AEGEE. Before using this module, the messages in the source code need to be customized.

6.8 Module block_sender

The module is loaded, when the configuration file contains section `[mod_block_sender]`.

6.9 Module delay

The module is loaded, when the configuration file contains section `[mod_delay]`.

The more time spammers sent in waiting, the less amount of spam they send. `mod_delay` ensures that the SMPT server spends a minimum amount of time in each phase during the delivery.

- `ehlo` the minimum amount of seconds that the client will wait for an answer of EHLO
- `mail` the minimum amount of seconds that the client will wait for an answer of MAIL FROM:

- `rcpt` the minimum amount of seconds that the client will wait for an answer of RCPT TO:
- `data` the minimum amount of seconds that the client will wait for an answer of end-of-data-fullstop.

If another module needs more time for a phase, than mandated by `mod_delay`, then `mod_delay` does not impose additional delays.

6.10 Module equal

The module is loaded, when the configuration file contains section `[mod_equal]`.

This module makes sure, that the envelope sender and the first recipient of the mail envelope are distinct addresses. There is some amount of spam, where the envelope sender and recipient coincide and for some sites, there are no use cases, where the sender can coincide with the first recipient.

6.11 Module relayed

The module is loaded, when the configuration file contains section `[mod_relayed]`.

For mails that are not rejected, a line with the sender and recipient is inserted in `list_log`. With `sendmail` this information is hard to extract from the log files.

6.12 Module sieve

The module is loaded, when the configuration file contains section `[mod_sieve]`.

It executes a Sieve script on every incoming mail per recipient.

The only actions executed are `redirect`, `reject` (rejects at SMTP level), `explicit keep` and `discard`. Actions `fileinto` and `vacation` are ignored. Cyrus Imap 3.1 has action `ereject`, but `aegee-milter` works with 3.0 and 3.0 has no `ereject`.

Calling the Sieve extensions from RFC 5490 is undefined behaviour.

Long rejection lines are split (folded) by `aegee-milter` when sent over SMTP, so the Sieve script can contain very long text after the `reject` action.

The executed script is the default script for the recipient. If there is no such script, then the default global script for the domain is executed. If there is no such script, `mod_sieve` does nothing.

6.13 Module sieve_global

The module is loaded, when the configuration file contains section `[mod_sieve_global]`.

It executes the same Sieve script for every incoming email.

The remarks for See Section 6.12 [Module sieve], page 10, apply also here.

The executed script is called "global.siv" and must be present in the global scripts for the domain of the recipient. If the script is missing, `sieve_global` does nothing.

6.14 Module via

The module is loaded, when the configuration file contains section [mod_via].

Its purpose is to log the second last mailhost from the Received: header via list_log.

7 Invoking aegee-milter

On start aegee-milter looks for its configuration file `aegee-milter.ini` in the `$sysconfdir`, set at configure time. To change the directory, where the configuration file is looked for, or the configuration file, use the `-c` command line parameter.

By default, aegee-milter forks in the background. To run it in the foreground, use `-f`.

To print the version of aegee-milter or a short usage information, use `-v` and `-h` respectively.

The following invocation looks for `aegee-milter.ini` in `/etc/mails/am` and starts aegee-milter as daemon process.

```
aegee-milter -c /etc/mails/am/
```

The following invocation loads the configuration file `/etc/mails/am.ini` and keeps the process in the foreground:

```
aegee-milter -f -c /etc/mails/am.ini
```

8 Bugs and Community

Interested parties, willing to report bugs, send patches or propose feature requests, are invited to join the `milter-1@lists.aegee.org` mailing list by visiting <https://lists.aegee.org/join/milter-1>. Leaving the mailing list can be triggered from the same URL. Writing to the mailing list is permitted only with addresses that are subscribed: One of Sender:, From:, Resent-Sender:, or Resent-From: headers must match, otherwise the posting is rejected.

Please use common sense when writing to the mailing list on non-technical matters and avoid making statements, which you would not make talking directly to a person.

Interested parties, for which subscribing to a mailing list is inconvenient, can write to `milter@aegee.org`. The author of aegee-milter prefers, that interested parties use the mailing list.

Contributions, meaning patches sent in any form, will be handled in reasonable time. Patches, for which there are no compelling reasons to reject them, will be accepted. This means, in the event that the author has no time to review a patch and there are no objections over the mailing list three months after proposing a patch, that the patch will be accepted.

Changes that have no other purpose, except to silence warnings from the compiler, linker or code analyzer will be accepted, unless they slow down the code. The rationale is, that if such changes are not accepted, in the future somebody will propose them again and handling similar proposals several times is avoidable work.

The language is C++14. Lifting to C++17 will reduce the amount of systems, where the software can be compiled.

Shorter code is better, but rewriting the whole program to be a single line of code, leads nowhere.

There is no strict code-style.

Text is in British English.

There is no plan for the event that the author is not capable to respond, but ideas for such a plan are welcomed.

These guidelines might change in the future.

9 Roadmap

Priorities are, in this order:

- Fixing problem in features that are claimed to be implemented.
- Updating the documentation
- Implementing Reject for all recipients, spelling in the reject reason the applied action per recipient.
- Implementing PRDR rejection mode. This is updating sendmail or postfix to be able to offer PRDR, when acting as server, and aegee-milter to be able to provide Per recipient delivery response over the milter interface.
- Implementing faked delay rejection mode. That is, from the perspective of the sender performing segmenting,

10 Development

10.1 Developing the Core

When a message is sent to many recipient and is handled by many modules, aegee-milter creates internally something like a table. Each column represents a recipient and each row is a module. When a module provides the same results for two recipients, then that cell in the table is extended and the two modules share the cell.

For example for a message with three recipients, that is handled by the modules delay and sieve, the table looks like:

	rec1	rec2	rec3
mod_sieve	A	B	A
mod_delay	D	D	D

The same letter in two cells means that the cell is shared.

Here, mod_sieve think that the result of the evaluation for rec1 and rec3 will be the same, therefore the cell in row mod_sieve in columns rec1 and rec3 is shared. mod_delay has for all recipients the same handling, therefore all cells in the last row a shared.

aegee-milter executes the code for each cell only once. In the above example mod_sieve will be executed for rec1 and rec3. For rec2 it will not be executed, because the result of executing it will be the same, as for rec1. mod_delay will be executed only once for this

message and all recipients, since the result for executing the module for each recipient is the same.

The core exports a variable `GKeyFile *inifile`, which lists and modules can import with `extern "C" GKeyFile *inifile`. It points to the open INI configuration file. Lists and modules shall check the value only in the sections with their own name. Currently, if there is an unrecognized option in a section, the corresponding list/module throws an exception and aegee-milter terminates. However, if there is an unrecognized section name, the section is ignored.

The core stores per SMTP transaction data in a variable of type `Privdata`. This variable is passed to any call of the modules and is not passed to call to methods from the lists. It turned out in practice, that lists do not need the per SMTP transaction data, in order to serve their purpose.

10.2 The build System

aegee-milter is build with `autoconf`, `automake` and `libtool`. There is a single Makefile, except for `libtool`.

Lists and modules are `libtool` Section “`dlpreopened`” in `libtool` `dlpreopened` objects. This means, that each list and module is first linked for itself and then the linked list or module is linked in the executable of aegee-milter. At startup aegee-milter iterates over its ELF symbols, stripping them to the first dot.

If a section in the INI file exists with that name, and the preloaded name starts with `'m'`, aegee-milter tries to load it as module.

If a section in the INI file exists with that name, and the preloaded name starts with `'l'`, aegee-milter tries to load it as list.

At `make install` the `libtool` `dlpreloaded` objects are installed. This is a known deficiency, since the installed objects are not needed at runtime, with an unknown solution.

10.3 Developing Modules

If aegee-milter considers a symbol to be loaded as module, it extracts and executes a function called `'init'` from the `dlpreopened` object. If this function exists, and returns a `SoModule` object, then the module is loaded and added to `AegeeMilter::so_modules`.

Modules are executed for each recipient in the order they are loaded by aegee-milter and the order of loading depends on the order modules are appended to aegee-milter during linking.

Each module derives from the `SoModule` class.

aegee-milter offers a module to store data per SMTP session, per transaction and per recipient.

When a method from a module is called, the first passed argument is of type `Privdata`.

To compare, whether for two recipients a module will have the same results, on the SMTP transaction (accept/reject/reject stating the message was quarantiened), the core calls for the second recipient on the `Equal` method with two recipients as parameter. If it returns true, then the core executes that module only once, for the first recipient, and the result of the execution is applied to both recipients.

Note, that if `Equals()` returns false, the implications are not only increased run-time needs. Depending on the mode of `aegee-milter`, this can segment the message, in one way or another imposing delivery delays.

`aegee-milter` calls the `Run` method of the modules at these stages:

- after RCPT:
- after end of headers
- after end of data

To see at which stage a module is called, it can execute

10.4 Developing Lists

If `aegee-milter` considers a symbol to be loaded as list, it extracts and executes a function called `'init'` from the `dlpreopened` object. If this function exists, and returns a `SoList` object, then the `Tables()` method of that object is called. A single `SoList` object can provide several tables. If it returns a non-empty `std::vector<std::string>`, then the `std::strings` are added to `AegeeMilter::so_tables` and linked to the `libtool dlpreopened` object. Otherwise the list is unloaded.

The order in which lists are loaded is insignificant.

Each list derives from the `SoList` class.

The method `Query` gets as parameters: the table name, from which to obtain information, the user (recipient), for which to obtain information and what information is asked for that user.

The method `Insert` inserts for a table, user and key a value.

The method `Remove` is used to remove from a table and user a specific key. Currently no list implements this method and no module calls it.

The method `Expire` is meant to expunge outdated data from a table and to be triggered regularly by `aegee-milter`. Currently no list implements it and the default implementation does nothing. The idea is to insert in a table information when vacation/out-of-office messages are sent, and to delete the information.

How the parameters are exactly interpreted is left to the modules and lists implementations and the agreements between them. For instance, `list_log::Insert` uses the user and key to compose an appropriate log message and `list_log::Query` is not called. `list_timsieved_scripts::Query` encodes in the user either which user and domain are meant, or if the global scripts for particular domain are requested. The key is used to request either the default script, or a particular script name. `list_timsieved_scripts::Insert` is not used by any module and is not implemented.

11 History

`aegee-milter` was developed for a mail server that served a lot of aliases redirecting to different email providers, and mailing lists with mailboxes on different email providers. The challenge was to keep the IP reputation good. This means not to forward received spam, recognized as such by the anti-spam software and not to send bounces. Bounces were sent

from the mailing list manager e.g. when a sender uses an address for posting which cannot post to the mailing list.

The further, some mailboxes received more spam than the others, so their spam filter had to be set more aggressively. Thus one mail sent simultaneously to an address with more aggressive spam filter and with less aggressive spam filter had to be rejected for the first recipient and accepted for the second.

When the author of aegee-milter took over a mail server and installed anti-spam software, he modified the mailing lists to bounce emails evaluated as spam.

The mail server

Appendix A GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that

modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list

of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit

them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a. The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined

with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage.

For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d. Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the

first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey

the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM

TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

Appendix B GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
<http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time

you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

- K. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled “Endorsements”. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C) year your name.  
Permission is granted to copy, distribute and/or modify this document  
under the terms of the GNU Free Documentation License, Version 1.3  
or any later version published by the Free Software Foundation;  
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover  
Texts. A copy of the license is included in the section entitled ‘‘GNU  
Free Documentation License’’.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being list their titles, with  
the Front-Cover Texts being list, and with the Back-Cover Texts  
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index

(Index is nonexistent)