

Sympa Mailinglist Manager

Stefan Hornburg (Racke)

Contents

Setup	4
Database Connection	4
Listmaster	4
Fast CGI	4
Logging	4
Configuration	5
Messages	5
Aliases	6
Generic aliases	6
Listmaster	6
Virtual domains (Robots)	7
Exim integration	8
Routers	8
Multiple domains	8
Postfix integration	10
Aliases	10
FCGI with systemd socket and multiwatch	11
Socket unit (Debian)	11
Service unit (Debian)	11
Nginx integration	12
Archives	14
Templates	15
Customizing	15
Email	15
Password request	15
Subscriptions	15
Subscription expiration	15
Messages	15
Reminders	15
Digests	16
Shared documents	16
Web	16
Custom list templates	16
Bounces	17

Topics	18
Commandline	19
Create a (test) list	19
Add user to a list	19
Trigger list synchronisation	20
Close a list	20
Purge a list	20
Open a list	20
Move list to another robot	20
Change configuration values	20
Mail aliases	21
Scripts	22
General Data Protection Regulation (GDPR)	23
Troubleshooting	24
No outgoing emails	24
Login problems	24
Maintenance mode	24
Sending emails	24
Important changes	24
Release 6.2.68	24
Known bugs	24
Internal processes	26
Distribute Message	26
Moderation	26
Outgoing messages	26
Bounces	26
Resources	27

Setup

Database Connection

In order to use a local socket connection to PostgreSQL, use the socket directory as database host in the Sympa configuration:

```
db_host /var/run/postgresql
```

This allows the Sympa user to safely connect to the database without password.

Listmaster

In your initial setup, you most likely need to adjust the global *listmaster* setting in `sympa.conf`:

```
listmaster mail@post.perl.dance
```

Fast CGI

Make sure that `use_fastcgi` is enabled in `sympa.conf`:

```
## use_fast_cgi
## Is FastCGI module for HTTP server installed (0 | 1)
## This module provide much faster web interface
use_fast_cgi 1
```

Fast CGI will be automatically enabled from Sympa 6.2.24.

Logging

Follow logs on a *systemd* based setup:

```
journalctl -u nginx-sympa -u sympa-archived -u sympa-bounced -u sympa.bulk -u sympa -u sympa-task-ma
```

Configuration

Messages

The maximum size of an email is 5 MB.

This setting can be changed by the *max_size* variable which uses bytes as unit.

So in order to set the maximum size to 50 MB (50 * 1024 * 1024 bytes):

```
max_size 52428800
```

Aliases

Aliases are used to push incoming emails to Sympa. We distinguish between generic aliases and list aliases.

Generic aliases

Listmaster

Email sent to the listmaster alias (e.g. *listmaster@sympa.pm*) will be relayed to all listmasters.

From: the original sender

Envelope from: sympa-request (e.g. *sympa_request@sympa.pm*).

Virtual domains (Robots)

A robot for a virtual domain (e.g. `mydomain.sympa.pm`) is specified by a configuration file `robot.conf` in a directory with the same name as the virtual domain.

This directory is located in the configuration directory (usually `/etc/sympa`).

So for our example `mydomain.sympa.pm` the path to the configuration file would be `/etc/sympa/mydomain.sympa.pm/`

A minimal example of that configuration file looks like:

```
domain mydomain.sympa.pm
wvsympa_url https://mydomain.sympa.pm/sympa
```

In addition to that, a directory with the same name as the virtual domain is **required** in the list home directory, e.g `/usr/local/sympa/list_data/mydomain.sympa.pm`.

Exim integration

Routers

We need a routers for the generic aliases and for the list aliases.

A simple solution for a single domain would be the following router:

```
sympa_list_aliases:
  debug_print = "R: sympa list aliases for $local_part@$domain"
  driver = redirect
  domains = sympa.example.com
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/mail/sympa/aliases}}
  user = sympa
  group = sympa
  pipe_transport = address_pipe
```

This applies to a setup with Debian packages, split configuration enabled.

Add a file with your routers, e.g. `etc/exim4/conf.d/router/450_sympa-routers:`

```
sympa_generic_router:
  driver = accept
  local_part_suffix_optional
  local_part_suffix = -bounces : -bounces+* : -editor : \
                    -confirm+* : -join : -leave : \
                    -request : -admin : -subscribe : -unsubscribe
  transport = sympa_transport
```

Add a file with your transports, e.g. `etc/exim4/conf.d/transport/30_sympa-transports:`

```
sympa_transport:
  driver = pipe
  command = /usr/lib/sympa/bin/queue ${local_part}${local_part_suffix}\@$domain
  user = sympa
  group = sympa
```

Update your Exim configuration file with `update-exim4.conf`.

Multiple domains

We are using one alias file per robot in `etc/mail/sympa`.

Add this router to the Exim configuration:


```
sympa_alias_router:
  debug_print = "R: list aliases $local_part@$domain"
  driver = redirect
  allow_fail
  allow_defer
  domains = dsearch;/etc/mail/sympa/
  data = ${expand:${lookup{$local_part}lsearch*@{/etc/mail/sympa/$domain}}}
  retry_use_local_part
  user = sympa
  group = sympa
  pipe_transport = address_pipe
  file_transport = address_file
  no_more
```

The location of the alias file is configured in the corresponding robot configuration file, e.g. for the domain `post.perl.dance`:

```
/etc/sympa# cat post.perl.dance/robot.conf
wvsympa_url https://post.perl.dance/sympa
title Perl Dancer mailing lists
sendmail_aliases /etc/exim4/sympa-robots/post.perl.dance
```

Postfix integration

This applies to a setup with Debian packages.

Aliases

We put the generic Sympa aliases like `sympa`, `sympa-request` and `sympa-owner` into `/etc/aliases`.

List aliases are going into `/etc/mail/sympa/aliases`. These are automatically updated by Sympa. Please make sure that this file is owned by the `sympa` user.

Add the file with Sympa aliases to `main.cf`:

```
alias_maps = hash:/etc/aliases,hash:/etc/mail/sympa/aliases
alias_database = hash:/etc/aliases,hash:/etc/mail/sympa/aliases
```

Remember that each time you update this file manually, you have to run the following command:

```
postalias /etc/mail/sympa/aliases
```

FCGI with systemd socket and multiwatch

This is the recommended method and will be used in future releases of Debian and Fedora packages.

Socket unit (Debian)

```
# /lib/systemd/system/wwsympa.socket
[Unit]
Description = Sympa Web Interface Socket

[Socket]
SocketUser = www-data
SocketGroup = www-data
SocketMode = 0660

RuntimeDirectory = sympa
ListenStream = /run/sympa/wwsympa.socket

[Install]
WantedBy = sockets.target
```

Service unit (Debian)

```
# /lib/systemd/system/wwsympa.service
[Unit]
Description=Sympa Web Interface FastCGI backend

[Service]
User=sympa
Group=sympa
ExecStart=/usr/bin/multiwatch -f $WWS_FCGI_CHILDREN \
    -- /usr/lib/cgi-bin/sympa/wwsympa.fcgi
StandardOutput=null
StandardInput=socket
StandardError=null

Environment="WWS_FCGI_CHILDREN=5"
EnvironmentFile=-/etc/default/sympa

Restart=always
RestartSec=5

[Install]
WantedBy=multi-user.target
```

Nginx integration

The systemd unit file for Debian looks like that:

```
[Unit]
Description=WWSympa - Web interface for Sympa mailing list manager
After=syslog.target
BindTo=sympa.service

[Service]
Type=forking
PIDFile=/run/sympa/wwsympa.pid
ExecStart=/usr/bin/spawn-fcgi -F $FCGI_CHILDREN \
  -P /run/sympa/wwsympa.pid \
  -u $FCGI_USER -g $FCGI_GROUP $FCGI_OPTS -- \
  /usr/lib/sympa/bin/wwsympa.fcgi
Environment="FCGI_CHILDREN=5"
Environment="FCGI_USER=sympa"
Environment="FCGI_GROUP=sympa"
Environment="FCGI_OPTS=-s /run/sympa/wwsympa.socket -M 0600 -U nginx"
EnvironmentFile=-/etc/sysconfig/sympa
Restart=always

[Install]
WantedBy=multi-user.target
```

This requires `spawn-fcgi` to be installed.

We are using the following snippet for inclusion into virtual hosts:

```
location /wvs {
    gzip off;
    fastcgi_pass    unix:/var/run/sympa/wwsympa.sock;
    fastcgi_split_path_info ^(/wvs)(.+)$.
    fastcgi_param  QUERY_STRING        $query_string;
    fastcgi_param  REQUEST_METHOD      $request_method;
    fastcgi_param  CONTENT_TYPE        $content_type;
    fastcgi_param  CONTENT_LENGTH      $content_length;
    fastcgi_param  PATH_INFO           $fastcgi_path_info;
    fastcgi_param  SCRIPT_NAME         $fastcgi_script_name;
    fastcgi_param  REQUEST_URI         $request_uri;
    fastcgi_param  DOCUMENT_URI        $document_uri;
    fastcgi_param  DOCUMENT_ROOT       $document_root;
    fastcgi_param  SERVER_PROTOCOL     $server_protocol;
    fastcgi_param  GATEWAY_INTERFACE   CGI/1.1;
    fastcgi_param  SERVER_SOFTWARE     nginx;
```

```
fastcgi_param  REMOTE_ADDR      $remote_addr;
fastcgi_param  REMOTE_PORT      $remote_port;
fastcgi_param  SERVER_ADDR      $server_addr;
fastcgi_param  SERVER_PORT      $server_port;
fastcgi_param  SERVER_NAME      $server_name;
fastcgi_param  REMOTE_USER      $remote_user;
fastcgi_param  SCRIPT_FILENAME  /usr/lib/cgi-bin/sympa/wwsympa-wrapper.fcgi;
fastcgi_param  HTTP_HOST        $server_name;
fastcgi_param  HTTPS            $https;
fastcgi_intercept_errors on;
```

```
}
```

Archives

Whether archives are used for a list is controlled by the *process_archive* variable.

Although the default for this variable is **off**, virtually every list is created with archiving enabled.

This can only be prevented if you modify a list template or create a new list template with *process_archive* set to off.

Templates

Documentation

Customizing

In order to customize templates you need to **copy them first** from their installation directory into the Sympa configuration directory.

For example, to edit the template for the welcome letter:

```
mkdir -p /etc/sympa/mail_tt2
cp /usr/share/sympa/default/mail_tt2/welcome.tt2 /etc/sympa/mail_tt2
```

The directories differ depending on your distribution and installation method.

Email

Password request

`sendpasswd.tt2` Password request

Subscriptions

`bye.tt2` Confirmation of unsubscription

`removed.tt2` Notification of removal from mailing list

`welcome.tt2` Welcome letter

Subscription expiration

`expire_deletion.tt2`

`expire_warning1.tt2`

`expire_warning2.tt2`

Messages

`reject.tt2` Rejection (only for moderated lists)

Reminders

`global_remind.tt2` Reminder for all subscriptions

`remind.tt2` Reminder for mailing list

Digests

`digestplain.tt2`

`digest.tt2`

Shared documents

`d_install_shared.tt2` Successful document upload

`d_reject_shared.tt2` Failed document upload

Web

`compose_mail.tt2` Form to compose emails

`footer.tt2` Footer (Logo and Sympa version)

`header.tt2` Top menu bar

`list_menu.tt2` Left hand side menu

`list_panel.tt2` Panel with subscribers, owners and editors

`editsubscriber.tt2` Edit subscriber information

`review.tt2` Administer subscribers

`user_menu.tt2` User dropdown menu in top bar

`user_notification.tt2` Various notifications, e.g. info email for new moderators

Custom list templates

Directory `/etc/sympa/create_list_templates`

Each template consists of a config (`config.tt2`) and comment (`comment.tt2`) template.

The contents of the comment template are shown on the "Request a List" page.

Don't put comments other than on the top into the config template as they will get mangled in the resulting config file.

Available templates can be configured in `/etc/sympa/create_list.conf`.

Bounces

Sympa uses the *ErrorsTo* header for the return address, e.g. test-owner@mylist.de.

Topics

Configuration file: `/etc/sympa/topics.conf`
Topics are loaded by `load_topics`

Commandline

The new syntax refers to Sympa release 6.2.68 and later.

Create a (test) list

Command:

```
sympa --create_list --input_file=/var/lib/sympa/testlist.xml
```

Sample contents of the file:

```
<?xml version="1.0" ?>
<list>
  <listname>testliste</listname>
  <type>html-news-letter</type>
  <subject>Test list</subject>
  <description/>
  <topics>computer</topics>
  <status>open</status>
  <owner multiple="1">
    <email>racke@linuxia.de</email>
  </owner>
  <owner multiple="1">
    <email>test@linuxia.de</email>
  </owner>
</list>
```

Add user to a list

```
~ echo test@linuxia.de | sympa add test@sympa.icdev.de
add [notice] 1 subscribers added
add [notice] User test@linuxia.de has been notified
```

Old syntax:

```
~ echo racke@linuxia.de | sympa --import test@sympa.icdev.de
add [notice] User racke@linuxia.de is now subscriber of list test.
Total imported subscribers: 1
```

Add `--quiet` parameter to prevent a welcome email send to the subscriber.

Trigger list synchronisation

For subscribers:

```
$ sympy include testlist@sympa.icdev.de
```

For owners:

```
$ sympy include --role=owner testlist@sympa.icdev.de
```

Old syntax:

```
$ sympy --sync_include=test@linuxia.de
```

Close a list

```
~# sympy --close_list test
close_list [notice] List has been closed
```

Lists that are included in another list through `include_sympa_list` can't be closed.

Purge a list

This **completely removes** a list. It can't be restored or reopened.

```
~# sympy --purge_list test
close_list [notice] List has been purged
```

New syntax:

```
$ sympy close --mode=purge test
close_list [notice] List has been purged
```

Open a list

```
~# sympy --open_list test
open_list [notice] Aliases have been installed.
open_list [notice] List has been restored
```

Move list to another robot

```
~# sympy --rename_list=test@linuxia.de --new_listname=test
--new_listrobot=icdev.de
```

Change configuration values

```
$ sympy config lang=en
/etc/sympa/sympa.conf have been updated.
Previous versions have been saved as /etc/sympa/sympa.conf.20221129114546.
```

Mail aliases

Create a temporary file with all aliases:

```
sympa --make_alias_file
```

Scripts

General Data Protection Regulation (GDPR)

In order to comply with the GDPR, Sympa allows users to delete their account and cancel subscriptions if the configuration parameter *allow_account_deletion* is set to *On*.

This feature was introduced with version 6.2.42 and considered as experimental.

Troubleshooting

No outgoing emails

Posts appear in the archives, but they are not sent to the subscribers. Check if *sympa-outgoing* service is running.

Login problems

Grep the Sympa log for the term `check_auth`:

```
$ grep check_auth /var/log/sympa.log
Jan 31 10:26:21 mysympa wwsympa[844003]: notice Sympa::WWW::Auth::ldap_authentication() Authentication
Jan 31 10:26:21 mysympa wwsympa[844003]: err main::#1556 > main::do_login#3175 > Sympa::WWW::Auth::c
```

This is especially useful to find out why authentication with LDAP / Active Domain Controller fails.

Maintenance mode

This happens when WWSympa discovers that your installation has defects, e.g. if the value in `/etc/sympa/data_structure.version` is different than your actual Sympa version.

Sending emails

Sympa will drop emails that don't match the following requirements:

- unique message ID
- Sender: header

Important changes

Release 6.2.68

Introduced new CLI interface (`sympa add` instead of `sympa.pl -add`)

Known bugs

This list is by all means not exhaustive. These bugs are already fixed unless otherwise noted.
Fixed in next release (6.2.64)

- Sympa should not lock out users using password authentication with LDAP (#1132)

Release 6.2.58 (and older)

- Upgrade fails with stricter SQL mode in MySQL 8 (#1028)

Release 6.2.54 (and older)

- Crash due to syntax error in regular expression (#860)

Release 6.2.52 (only)

- Evaluation of scenari fails (#841)

Release 6.2.52 (and older)

- Error on messages signed with PGP/Mime (#867)
- Gecos missing from file import (#873), introduced through version 6.2.48

Release 6.2.40 (and older)

- Shared feature can not be turned off (#1035)

Internal processes

Distribute Message

The following modules are called:

- Sympa::Spindle::TransformIncoming
- Sympa::Spindle::ToArchive
- Sympa::Spindle::TransformOutgoing
- Sympa::Spool::Outgoing
- Sympa::Spindle::ToDigest
- Sympa::Spindle::ToList

Sympa::Spool::Outgoing stores the messages in the bulk spool (bulk/msg and bulk/pct).
The bulk daemon picks up these messages.

Moderation

Spindle Sympa::Spindle::ModerateMessage

The message is either distributed (we pass to Sympa::Spindle::DistributeMessage) or rejected.
The following steps are taken before we distributing the message:

- Add `X-Validation-by` header
- Inform author (unless *quiet* is enabled)

Outgoing messages

This applies to other messages than list posts.

Spindle Sympa::Spindle::ToOutgoing

Bounces

Spindle Sympa::Spindle::ProcessBounce

Resources

GitHub repository <https://github.com/sympa-community/sympa>

Linuxia Wiki

Stefan Hornburg (Racke)
Sympa Mailinglist Manager

wiki.linuxia.de