## Sympa Administration Manual

Version 6.2.54

### Table of contents

- **Audiences and scope**
- **Requirements**
- **Installing Sympa**
  - Install Sympa distribution
  - Install dependent modules
  - Generate initial configuration
  - Setup database
  - Configure system log
  - Configure mail server
  - Configure HTTP server
  - Start mailing list service
  - Automate startup
- **Upgrading Sympa**
  - Upgrading Sympa in place
  - Moving to another server
  - Upgrading notes
- **Administering Sympa**
  - Managing services
  - Command line interface
  - Managing lists
    - List creation, editing and removal
    - Managing list members, owners and moderators
  - Web interface for listmaster
- **Customizing Sympa**
  - Customization basics
    - Message workflow
    - Roles and privileges
    - Configuration hierarchy
    - Configuration for each list
    - Templates
    - Authorization scenarios
    - Tasks
    - Internationalization
    - List families
  - Customizing Sympa services
  - Customizing web interface
  - Sympa services: Optional features
  - Web interface: Optional features
  - Extending Sympa
  - Sympa and other systems
  - Other topics

---

Sympa mailing list server - https://www.sympa.org/
References
- Reference manual
- Daemons
- Web interface
- Shell Interface
- Auxiliary Programs
- Configuration files
- Internals

Directory layout

Documentation on older versions

Manual convention

- The `monospace font` represents the text readable by machine: Name of any hosts, directories, files, commands or parameters, or value of any parameters. Example: > The `pw` command may add a user to `/etc/passwd`.
- Fenced block represents input and/or output on the console, or excerpt from any file. Example: `bash > $ echo 'Sympa!' > Sympa! > `Example: `>` domain mail.example.org `wwsympa_url https://web.example.org/sympa`
- `$` or `#` prepended to the line represents shell prompt. `$` indicates that commands may be executed by unprivileged user. `#` indicates that commands should be executed under super-user (root) privilege. In latter case `sudo(1)` may also be used. Example: `bash

make install

` is the same as `bash $ sudo make install`
- To help readers understand, several symbols are used to stand for particular path names. See "Directory layout".

2019/03/26 11:23