Authentication with PAM

Pluggable Authentication Modules (PAM) mediate access between users and applications in SUSE Linux Enterprise Server for NVIDIA. By centralizing authentication mechanisms through modular shared libraries, PAM allows administrators to adapt or introduce new authentication schemes without modifying individual applications. This framework significantly simplifies security management, reduces configuration errors and enhances system flexibility by enabling features like Universal 2nd Factor (U2F) hardware key authentication for local logins.

Prerequisites

- YubiKey or compatible Security Key hardware devices (for U2F configuration)

Publication Date: 05 Nov 2026

Contents

1 Introduction to PAM 3

2 Structure of PAM configuration 3

3 Configuration of PAM modules 8

4 Configuring PAM using **`pam-config`** 9

5 Manually configuring PAM 10

6 Configuring SLE for NVIDIA to require U2F keys for local login 11

7 Legal Notice 13

A GNU Free Documentation License 13

1 Introduction to PAM

System administrators and programmers often want to restrict access to certain parts of the system or to limit the use of certain functions of an application. Without PAM, applications must be adapted every time a new authentication mechanism, such as LDAP, Samba, or Kerberos, is introduced. However, this process is time-consuming and error-prone. One way to avoid these drawbacks is to separate applications from the authentication mechanism and delegate authentication to centrally managed modules. Whenever a newly required authentication scheme is needed, it is sufficient to adapt or write a suitable *PAM module* for use by the program in question.

The PAM concept consists of:

- *PAM modules*, which are a set of shared libraries for a specific authentication mechanism.
- A *module stack* with one or more PAM modules.
- A PAM-aware *service* which needs authentication by using a module stack or PAM modules. Usually a service is a familiar name of the corresponding application, like **`login`** or **`su`**. The service name `other` is a reserved word for default rules.
- *Module arguments*, with which the execution of a single PAM module can be influenced.
- A mechanism evaluating each *result* of a single PAM module execution. A positive value executes the next PAM module. The way a negative value is dealt with depends on the configuration: “no influence, proceed” up to “terminate immediately” and anything in between are valid options.

2 Structure of PAM configuration

PAM on SUSE Linux Enterprise Server for NVIDIA comes with a so-called directory-based configuration. The set of configuration files is stored in `/etc/pam.d`. Every service (or program) that relies on the PAM mechanism has its own configuration file in this directory. For example, the service for `sshd` can be found in the `/etc/pam.d/sshd` file.

The default configuration files provided by SUSE are stored in `/usr/lib/pam.d`. Do not modify these files, as they get overwritten during system updates. For custom changes to the configuration, use the directory `/etc/pam.d`. Either create configuration files there or use the directory to store **`pam-config`** generated configurations. If a particular configuration file exists in `/etc/pam.d/`, for example, `/etc/pam.d/sshd`, such a configuration file takes precedence over `/usr/lib/pam.d/sshd`.



Note: File-based configuration (`/etc/pam.conf`) not used on SUSE Linux Enterprise Server for NVIDIA

The configuration of each service can also be stored in `/etc/pam.conf`. However, for maintenance and usability reasons, this configuration scheme is not used in SUSE Linux Enterprise Server for NVIDIA.

The files under `/etc/pam.d/` define the PAM modules used for authentication. Each file consists of lines, which define a service, and each line consists of a maximum of four components:

```
TYPE  CONTROL
 MODULE_PATH  MODULE_ARGS
```

The components have the following meaning:

TYPE

Declares the type of the service. PAM modules are processed as stacks. Different types of modules have different purposes. For example, one module checks the password, another verifies the location from which the system is accessed, and yet another reads user-specific settings. PAM knows about four different types of modules:

`auth`

Check the user's authenticity, traditionally by querying a password. However, this can also be achieved with a chip card or through biometrics (for example, fingerprints or iris scan).

`account`

Modules of this type check if the user has general permission to use the requested service. As an example, such a check should be performed to ensure that no one can log in with the user name of an expired account.

`password`

The purpose of this type of module is to enable the change of an authentication token. Usually this is a password.

`session`

Modules of this type are responsible for managing and configuring user sessions. They are started before and after authentication to log login attempts and configure the user's specific environment.

CONTROL

Indicates the behavior of a PAM module. Each module can have the following control flags:

`required`

A module with this flag must be successfully processed before the authentication may proceed. After the failure of a module with the `required` flag, all other modules with the same flag are processed before the user receives a message about the failure of the authentication attempt.

`requisite`

Modules having this flag must also be processed successfully, in much the same way as a module with the `required` flag. However, in case of failure a module with this flag gives immediate feedback to the user and no further modules are processed. In case of success, other modules are subsequently processed, like any modules with the `required` flag. The `requisite` flag can be used as a basic filter checking for the existence of certain conditions that are essential for a correct authentication.

`sufficient`

After a module with this flag has been successfully processed, the requesting application receives an immediate message about the success and no further modules are processed, provided there was no preceding failure of a module with the `required` flag. The failure of a module with the `sufficient` flag has no direct consequences, in the sense that any subsequent modules are processed in their respective order.

`optional`

The failure or success of a module with this flag does not have any direct consequences. This can be useful for modules that are only intended to display a message (for example, to tell the user that mail has arrived) without taking any further action.

`include`

If this flag is given, the file specified as argument is inserted at this place.

MODULE_PATH

Contains a full file name of a PAM module. It does not need to be specified explicitly, if the module is located in the default directory `/lib/security` (for all 64-bit platforms supported by SUSE Linux Enterprise Server for NVIDIA, the directory is `/lib64/security`).

MODULE_ARGS

Contains a space-separated list of options to influence the behavior of a PAM module, such as `debug` (enables debugging) or `nullok` (allows the use of empty passwords).

In addition, there are global configuration files for PAM modules under `/etc/security`, which define the exact behavior of these modules (examples include `pam_env.conf` and `time.conf`). Every application that uses a PAM module calls a set of PAM functions, which then process the information in the configuration files and return the result to the requesting application.

To simplify the creation and maintenance of PAM modules, common default configuration files for the types `auth`, `account`, `password`, and `session` modules have been introduced. These are retrieved from every application's PAM configuration. Updates to the global PAM configuration modules in `common-*` are thus propagated across all PAM configuration files without requiring the administrator to update every single PAM configuration file.

The global PAM configuration files are maintained using the **`pam-config`** tool. This tool automatically adds new modules to the configuration, changes the configuration of existing ones or deletes modules (or options) from the configurations. Manual intervention in maintaining PAM configurations is minimized or no longer required.

2.1 An example of PAM configuration

To demonstrate a real use case example of PAM configuration, the configuration of sshd has been used in this section:

EXAMPLE 1: PAM CONFIGURATION FOR SSHD (`/etc/pam.d/sshd`)

```
#%PAM-1.0 ❶
auth     requisite      pam_nologin.so                          ❷
auth     include        common-auth                             ❸
account  requisite      pam_nologin.so                          ❷
account  include        common-account                          ❸
password include        common-password                         ❸
session  required       pam_loginuid.so                         ❹
session  include        common-session                          ❸
session  optional       pam_lastlog2.so   silent noupdate showfailed ❺
```

❶ Declares the version of this configuration file for PAM 1.0. This is merely a convention, but could be used in the future to check the version.

❷ Checks, if `/etc/nologin` exists. If it does, no user other than `root` may log in.

❸ Refers to the configuration files of four module types: `common-auth`, `common-account`, `common-password`, and `common-session`. These four files hold the default configuration for each module type.

❹ Sets the login UID process attribute for the process that was authenticated.

5 Displays and manages user last login information using a Y2038-safe database.

By including the configuration files instead of adding each module separately to the respective PAM configuration, you automatically get an updated PAM configuration when an administrator changes the defaults.

The first include file (`common-auth`) calls modules of the `auth` type: `pam_env.so`, `pam_gnome_keyring.so` and `pam_unix.so`. See *Example 2, "Default configuration for the `auth` section (`common-auth`)"*. Keep in mind that the modules may differ according to your installation.

EXAMPLE 2: DEFAULT CONFIGURATION FOR THE `auth` SECTION (`common-auth`)

```
auth  required  pam_env.so                    1
auth  optional  pam_gnome_keyring.so          2
auth  required  pam_unix.so  try_first_pass  3
```

1 `pam_env.so` loads `/etc/security/pam_env.conf` to set the environment variables as specified in this file. It can be used to set the `DISPLAY` variable to the correct value, because the `pam_env` module knows about the location from which the login is taking place.

2 `pam_gnome_keyring.so` checks the user's login and password against the GNOME key ring

3 `pam_unix` checks the user's login and password against `/etc/passwd` and `/etc/shadow`.

The whole stack of `auth` modules is processed before `sshd` gets any feedback about whether the login has succeeded. All modules of the stack having the `required` control flag must be processed successfully before `sshd` receives a message about the positive result. If one of the modules is not successful, the entire module stack is still processed and only then is `sshd` notified about the negative result.

When all modules of the `auth` type have been successfully processed, another include statement is processed, in this case, that in *Example 3, "Default configuration for the `account` section (`common-account`)"*. `common-account` contains only one module, `pam_unix`. If `pam_unix` returns the result that the user exists, sshd receives a message announcing this success and the next stack of modules (`password`) is processed, shown in .

EXAMPLE 3: DEFAULT CONFIGURATION FOR THE `account` SECTION (`common-account`)

```
account  required  pam_unix.so  try_first_pass
```

Again, the PAM configuration of `sshd` involves only an include statement referring to the default configuration for `password` modules located in `common-password`. These modules must successfully be completed (control flags `requisite` and `required`) whenever the application requests the change of an authentication token.

Changing a password or another authentication token requires a security check. This is achieved with the `pam_cracklib` module. The `pam_unix` module used afterward carries over any old and new passwords from `pam_cracklib`, so the user does not need to authenticate again after changing the password. This procedure makes it impossible to circumvent the checks carried out by `pam_cracklib`. Whenever the `account` or the `auth` type are configured to complain about expired passwords, the `password` modules should also be used.

As the final step, the modules of the `session` type (bundled in the `common-session` file) are called to configure the session according to the settings for the user in question. The `pam_limits` module loads the file `/etc/security/limits.conf`, which may define limits on the use of certain system resources. The `pam_unix` module is processed again. The `pam_umask` module can be used to set the file mode creation mask. Since this module carries the `optional` flag, a failure of this module would not affect the successful completion of the entire session module stack. The `session` modules are called a second time when the user logs out.

3 Configuration of PAM modules

Some PAM modules are configurable. The configuration files are located in `etc/security`. This section briefly describes the configuration files relevant to the sshd example—`pam_env.conf` and `limits.conf`.

3.1 pam_env.conf

`pam_env.conf` can be used to define a standardized environment for users that is set whenever the `pam_env` module is called. With it, preset environment variables using the following syntax:

```
VARIABLE  [DEFAULT=VALUE]  [OVERRIDE=VALUE]
```

VARIABLE
: Name of the environment variable to set.

[DEFAULT=<value>]
: Default *VALUE* the administrator wants to set.

[OVERRIDE=<value>]
: Values that may be queried and set by `pam_env`, overriding the default value.

A typical example of how `pam_env` can be used is the adaptation of the `DISPLAY` variable, which is changed whenever a remote login takes place. This is shown in *Example 4, "pam_env.conf"*.

EXAMPLE 4: PAM_ENV.CONF

```
REMOTEHOST  DEFAULT=localhost          OVERRIDE=@{PAM_RHOST}
DISPLAY     DEFAULT=${REMOTEHOST}:0.0  OVERRIDE=${DISPLAY}
```

3.2 limits.conf

System limits can be set on a user or group basis in `limits.conf`, which is read by the `pam_limits` module. The file allows you to set hard limits, which may not be exceeded, and soft limits, which may be exceeded temporarily. For more information about the syntax and options, see the comments in `/etc/security/limits.conf`. The `/usr/etc/security` is only for distributions; administrators are not allowed to change this. The `pam_env.conf` follows the UAPI configuration file specification; administrators should follow that for their changes.

4 Configuring PAM using **`pam-config`**

The **`pam-config`** tool helps you configure the global PAM configuration files (`/etc/pam.d/common-*`) and several selected application configurations. For a list of supported modules, use the **`pam-config --list-modules`** command. Use the **`pam-config`** command to maintain your PAM configuration files. Add new modules to your PAM configurations, delete other modules or modify options to these modules. When changing global PAM configuration files, no manual tweaking of the PAM setup for individual applications is required.

A simple use case for **`pam-config`** involves the following:

1. **Auto-generate a fresh Unix-style PAM configuration.** Let **`pam-config`** create the simplest possible setup which you can extend later on. The **`pam-config --create`** command creates a simple Unix authentication configuration. Pre-existing configuration files not maintained by **`pam-config`** are overwritten, but backup copies are kept as `*.pam-config-backup`.

2. **Add a new authentication method.** Adding a new authentication method (for example, LDAP) to your stack of PAM modules comes down to a simple **`pam-config --add --ldap`** command. LDAP is added wherever appropriate across all `common-*-pc` PAM configuration files.

3. **Add debugging for test purposes.** To make sure the new authentication procedure works as planned, turn on debugging for all PAM-related operations. The **`pam-config --add --ldap-debug`** turns on debugging for LDAP-related PAM operations.

4. **Query your setup.** Before you finally apply your new PAM setup, check if it contains all the options you wanted to add. The **`pam-config --query --`***`MODULE`* command lists both the type and the options for the queried PAM module.

5. **Remove the debug options.** Finally, remove the debug option from your setup when you are entirely satisfied with the performance of it. The **`pam-config --delete --ldap-debug`** command turns off debugging for LDAP authentication. In case you had debugging options added for other modules, use similar commands to turn these off.

For more information on the **`pam-config`** command and the options available, refer to the manual page of **`pam-config(8)`**.

5 Manually configuring PAM

If you prefer to manually create or maintain your PAM configuration files, make sure to disable **`pam-config`** for these files.

When you create your PAM configuration files from scratch using the **`pam-config --create`** command, it creates symbolic links from the `common-*` to the `common-*-pc` files. **`pam-config`** only modifies the `common-*-pc` configuration files. Removing these symbolic links effectively disables **`pam-config`**, because **`pam-config`** only operates on the `common-*-pc` files and these files are not put into effect without the symbolic links.



Warning: Include `pam_systemd.so` in configuration

If you are creating your own PAM configuration, make sure to include `pam_systemd.so` configured as `session optional`. Not including the `pam_systemd.so` can cause problems with `systemd` task limits. For details, refer to the man page of `pam_systemd.so`.

6 Configuring SLE for NVIDIA to require U2F keys for local login

To provide more security during the local login to SLE for NVIDIA, you can configure two-factor authentication using the `pam-u2f` framework and the U2F feature on YubiKeys and Security Keys.

For this, you must have a YubiKey device and install the required `pam-u2f` package on your system.

```
> sudo  zypper install pam_u2f
```

To set up U2F on your SLE for NVIDIA system, you need to associate your key with your account on SLE for NVIDIA. After that, configure your system to use the key. The procedure is described in the following sections.

6.1 Associating the U2F key with your account

To associate your U2F key with your account, proceed as follows:

1. Log in to your machine.
2. Insert your U2F key.
3. Create a directory for the U2F key configuration:

   ```
   > sudo mkdir -p ~/.config/Yubico
   ```

4. Run the **`pamu2fcfg`** command that outputs configuration lines:

   ```
   > sudo pamu2fcfg > ~/.config/Yubico/u2f_keys
   ```

5. When your device begins flashing, touch the metal contact to confirm the association.

We recommend using a backup U2F device, which you can set up by running the following commands:

1. Run:

   ```
   > sudo pamu2fcfg -n >> ~/.config/Yubico/u2f_keys
   ```

2. When your device begins flashing, touch the metal contact to confirm the association.

You can move the output file from the default location to a directory that requires the `sudo` permission to modify the file to increase security. For example, move it to the `/etc` directory. To do so, follow the steps:

1. Create a directory in `/etc`:

```
> sudo mkdir /etc/Yubico
```

2. Move the created file:

```
> sudo mv ~/.config/Yubico/u2f_keys /etc/Yubico/u2f_keys
```



Note: Placing the `u2f_keys` to a non-default location

If you move the output file to a different directory than is the default (`$HOME/.config/Yubico/u2f_keys`), you need to add the path to the `/etc/pam.d/login` file as described in *Section 6.2, "Updating the PAM configuration"*.

6.2 Updating the PAM configuration

After you have created the U2F keys configuration, you need to adjust the PAM configuration on your system.

1. Open the file `/etc/pam.d/login`.

2. Add the line `auth required pam_u2f.so` to the file as follows:

```
#%PAM-1.0
auth      include    common-auth
auth      required   pam_u2f.so
account   include    common-account
password  include    common-password
session   optional   pam_keyinit.so revoke
session   include    common-session
#session  optional   pam_xauth.so
```

3. If you placed the `u2f_keys` file to a different location than `$HOME/.config/Yubico/u2f_keys`, you need to use the `authfile` option in the `/etc/pam.d/login` PAM file as follows:

```
#%PAM-1.0
auth     requisite pam_nologin.so
auth     include   common-auth
```

```
auth  required pam_u2f.so authfile=<PATH_TO_u2f_keys>
...
```

where *<PATH_TO_u2f_keys>* is the absolute path to the `u2f_keys` file.

7 Legal Notice

Copyright© 2006– 2026 SUSE LLC and contributors. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or (at your option) version 1.3; with the Invariant Section being this copyright notice and license. A copy of the license version 1.2 is included in the section entitled “GNU Free Documentation License”.

For SUSE trademarks, see https://www.suse.com/company/legal/. All other third-party trademarks are the property of their respective owners. Trademark symbols (®, ™ etc.) denote trademarks of SUSE and its affiliates. Asterisks (*) denote third-party trademarks.

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither SUSE LLC, its affiliates, the authors, nor the translators shall be held liable for possible errors or the consequences thereof.

A GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. 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. Secondarily, 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 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.

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 non-commercially, 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 for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

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 https://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.

ADDENDUM: How to use this License for your documents

```
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 1.2
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.