通过 PAM 进行身份验证

解释

本文介绍了 PAM 概念、PAM 配置结构以及用于配置 PAM 的工具的用法。

原因

您需要知道如何设置 PAM 模块并将系统配置为使用 U2F 密钥。

工作量

读完本文大约需要 20 分钟。

要求

- 要使用 U2F 密钥设置身份验证，您需要具有 YubiKey 或安全密钥。

出版日期：2025 年 12 月 11 日

目录

1 PAM 简介 3

2 PAM 配置的结构 3

3 PAM 模块的配置 8

4 使用 **pam-config** 配置 PAM 9

5 手动配置 PAM 10

6 将 SLE Micro 配置为需要 U2F 密钥才能进行本地登录 10

7 法律声明 13

A GNU 自由文档许可证 13

1 PAM 简介

系统管理员和编程人员经常要将访问限制在系统的某些部分或限制对应用程序某些功能的使用。没有 PAM，每次引入新的身份验证机制（例如 LDAP、Samba 或 Kerberos）时都必须对应用程序进行调整，而此过程非常耗时且容易出错。避免这些缺点的一种方法是将应用程序从身份验证机制中分开并将身份验证委托给集中管理的模块。每当需要使用新的必要身份验证模式时，只要调整或编写合适的 **PAM 模块**供相关程序使用即可。

PAM 的概念包括：

- **PAM 模块**，用于特定身份验证机制的一组共享库。
- 一个**模块堆栈**，其中包含一个或多个 PAM 模块。
- PAM 感知**服务**，需要使用模块堆栈或 PAM 模块进行身份验证。通常，服务是用户所熟悉的相应应用程序名称，例如 `login` 或 `su`。服务名称 `other` 是默认规则的保留字。
- **模块参数**，可用于影响单个 PAM 模块的执行。
- 用于评估执行单个 PAM 模块所产生的每种**结果**的机制。如果为正值，则执行下一个 PAM 模块。对负值的处理方式取决于配置：“无影响，继续”到“立即终止”之间的所有选项都有效。

2 PAM 配置的结构

SLE Micro 上的 PAM 附带所谓的基于目录的配置。配置文件集存储在 `/etc/pam.d` 中。依赖于 PAM 机制的每个服务（或程序）在此目录中都有各自的配置文件。例如，可以在 `/etc/pam.d/sshd` 文件中找到 `sshd` 的服务。

注意：SLE Micro 上不使用基于文件的配置 (`/etc/pam.conf`)。

每项服务的配置也可以存储在 `/etc/pam.conf` 中。但是，出于维护和可用性考虑，SLE Micro 中不使用此配置方案。

`/etc/pam.d/` 下的文件定义用于身份验证的 PAM 模块。每个文件都包含用于定义某项服务的行，而每行最多包含四个组成部分：

```
TYPE
CONTROL
MODULE_PATH
MODULE_ARGS
```

组成部分的含义如下：

`TYPE`

声明服务的类型。PAM 模块是成批处理的。不同类型的模块具有不同的用途。例如，一个模块检查口令，一个模块校验访问系统的位置，还有一个模块读取用户特定的设置。PAM 可以识别四种不同类型的模块：

`auth`

检查用户的真实性，传统方法是通过查询口令进行检查，但也可以通过芯片卡或生物特征（例如指纹或虹膜扫描）来实现此目的。

`account`

这种类型的模块会检查用户是否具有使用所请求服务的一般权限。例如，应执行这种检查以确保任何人都不能使用失效帐户的用户名登录。

`password`

这种类型的模块的用途是启用身份验证令牌的更改。这通常是一个口令。

`session`

这种类型的模块负责管理和配置用户会话。这些模块在身份验证前后启动，以记录登录尝试并配置用户的特定环境。

`CONTROL`

指示 PAM 模块的行为。每个模块都可以具有以下控制标志：

`required`

在进行身份验证之前，必须先成功处理带有此标志的模块。在处理带有 `required` 标志的模块失败后，将继续处理带有相同标志的所有其他模块，之后用户才会收到有关身份验证尝试失败的消息。

`requisite`

也必须成功处理带有此标志的模块，处理方式在很大程度上与带有 `required` 标志的模块类似。但是，如果某个带有此标志的模块失败，将立即向用户提供反馈并且不再继续处理其他模块。如果成功，则接着处理其他模块，就像带有 `required` 标志的任何模块一样。`requisite` 标志可用作基本过滤器，检查进行正确身份验证所必需的某些条件是否存在。

`sufficient`

在成功处理带有此标志的模块后，请求方应用程序会立即收到处理成功的消息并且不再处理其他模块，但前提是之前所有带有 `required` 标志的模块均未失败。带有 `sufficient` 标志的模块失败没有任何直接后果，所有随后的模块都将按其各自的顺序进行处理。

`optional`

带有此标志的模块成功或失败不会产生任何直接后果。此标志可用于只用来显示消息（例如，通知用户收到了邮件）而不采取任何进一步操作的模块。

`include`

如果给出此标志，则在此处插入指定为参数的文件。

`MODULE_PATH`

包含 PAM 模块的完整文件名。如果模块位于默认目录 `/lib/security`（对于 SLE Micro 支持的所有 64 位平台，默认目录均为 `/lib64/security`）中，则无需明确指定此文件名。

`MODULE_ARGS`

包含用于影响 PAM 模块行为的选项的空格分隔列表，例如 `debug`（启用调试）或 `nullok`（允许使用空口令）。

另外，`/etc/security` 下提供了用于 PAM 模块的全局配置文件，它们定义这些模块的确切行为（其中包括 `pam_env.conf` 和 `time.conf`）。使用 PAM 模块的每个应用程序都会调用一组 PAM 函数，这些函数随后将处理配置文件中的信息，并将结果返回给请求方应用程序。

为了简化 PAM 模块的创建和维护，现已引入 `auth`、`account`、`password` 和 `session` 模块类型的通用默认配置文件。这些配置取自每个应用程序的 PAM 配置。因此，对 `common-*` 中全局 PAM 配置模块进行的更新将在所有 PAM 配置文件中传播，而无需管理员更新每个 PAM 配置文件。

您可以使用 **`pam-config`** 工具维护全局 PAM 配置文件。此工具可自动将新模块添加到配置、更改现有模块的配置，或者从配置中删除模块（或选项）。它最大限度地减少甚至完全消除了维护 PAM 配置时所需的人工干预。

2.1 PAM 配置示例

为了演示 PAM 配置的真实应用场景示例，本节中使用了 sshd 的配置：

例 1： SSHD 的 PAM 配置 (`/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_lastlog.so   silent noupdate showfailed ❺
```

❶ 为 PAM 1.0 声明此配置文件的版本。这只是一项惯例，但将来可以使用它来检查版本。

❷ 检查 `/etc/nologin` 是否存在。如果不存在，则除 `root` 以外的任何用户都无法登录。

❸ 引用四种模块类型的配置文件：`common-auth`、`common-account`、`common-password` 和 `common-session`。这 4 个文件包含每种模块类型的默认配置。

❹ 设置已经过身份验证的进程的登录 UID 进程属性。

❺ 显示有关用户上次登录的信息。

通过包含配置文件而不是将每个模块单独添加到相应的 PAM 配置，您可以在管理员更改默认设置后自动获取更新的 PAM 配置。

第一个 include 文件 (common-auth) 会调用 auth 类型的模块：pam_env.so、pam_gnome_keyring.so 和 pam_unix.so。请参见例 2 “auth 部分的默认配置 (common-auth)”。请记住，模块可能因您的安装而异。

例 2：auth 部分的默认配置 (common-auth)

```
auth   required   pam_env.so                    ❶
auth   optional   pam_gnome_keyring.so         ❷
auth   required   pam_unix.so   try_first_pass ❸
```

❶ pam_env.so 加载 /etc/security/pam_env.conf 以设置此文件中指定的环境变量。它可用于将 DISPLAY 变量设置为正确的值，因为 pam_env 模块知道登录发生的位置。

❷ pam_gnome_keyring.so 根据 GNOME 密钥环检查用户的登录名和口令。

❸ pam_unix 根据 /etc/passwd 和 /etc/shadow 检查用户的登录名和口令。

整个 auth 模块堆栈处理完后，sshd 才会获得有关登录是否成功的反馈。堆栈中带有 required 控制标志的所有模块都必须成功处理，sshd 才能收到有关正面结果的消息。如果其中的某个模块不成功，则仍将继续处理整批模块，在此之后 sshd 才会收到处理失败的通知。

成功处理所有 auth 类型的模块后，将处理另一条 include 语句，在本例中为例 3 “account 部分的默认配置 (common-account)”中的语句。common-account 仅包含一个模块：pam_unix。如果 pam_unix 返回的结果证明用户存在，则 sshd 会收到一条处理成功的消息，然后处理下一批模块 (password)。

例 3：account 部分的默认配置 (common-account)

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

同样，sshd 的 PAM 配置仅涉及一条 include 语句，该语句引用了 password 模块的默认配置（位于 common-password 中）。每当应用程序请求获取身份验证令牌的更改信息时，都必须成功完成这些模块（控制标志为 requisite 和 required）。

更改口令或另一个身份验证令牌需要进行安全检查。这项检查是通过 pam_cracklib 模块实现的。随后使用的 pam_unix 模块带有来自 pam_cracklib 的任何旧口令和新口令，因此用户在更改口令后无需再次进行身份验证。此过程可确保不能绕过 pam_cracklib 所执行的检查。每当配置了 account 或 auth 类型来指出口令失效时，还应使用 password 模块。

最后，调用 session 类型的模块（捆绑在 common-session 文件中）以根据相关用户的设置来配置会话。pam_limits 模块加载文件 /etc/security/limits.conf，该文件可以定义某些系统资源的使用限制。再次处理 pam_unix 模块。pam_umask 模块可用于设置文件模式创建掩码。由于此模块带有 optional 标志，因此此模块的失败将不会影响整个会话模块堆栈的成功完成。当用户注销时，将再次调用 session 模块。

3 PAM 模块的配置

某些 PAM 模块是可配置的。配置文件位于 /etc/security 中。本节简要介绍与 sshd 示例相关的配置文件 - pam_env.conf 和 limits.conf。

3.1 pam_env.conf

pam_env.conf 可用于定义每次调用 pam_env 模块时为用户设置的标准化环境。它允许您使用以下语法预设环境变量：

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

VARIABLE
: 要设置的环境变量的名称。

[DEFAULT=<value>]
: 管理员要设置的默认 VALUE。

[OVERRIDE=<value>]
: 可能由 pam_env 查询并设置的值，会覆盖默认值。

有关如何使用 pam_env 的典型示例是 DISPLAY 变量的调整，每当发生远程登录时，该变量就会改变。例 4 “pam_env.conf” 中显示了这一点。

例 4： PAM_ENV.CONF

```
REMOTEHOST  DEFAULT=localhost          OVERRIDE=@{PAM_RHOST}
```

```
DISPLAY     DEFAULT=${REMOTEHOST}:0.0  OVERRIDE=${DISPLAY}
```

第一行将 REMOTEHOST 变量的值设置为 localhost，每当 pam_env 不能确定任何其他值时就会使用该值。DISPLAY 变量又包含 REMOTEHOST 的值。在 /etc/security/pam_env.conf 的注释中可以找到详细信息。

3.2 limits.conf

可以在 pam_limits 模块将会读取的 limits.conf 中基于用户或组设置系统限制。该文件可用于设置硬限制（即不能超出的限制）和软限制（即可以暂时超出的限制）。有关语法和选项的详细信息，请参见 /etc/security/limits.conf 中的注释。

4 使用 pam-config 配置 PAM

pam-config 工具可帮助您配置全局 PAM 配置文件 (/etc/pam.d/common-*) 和多个选定的应用程序配置。如需受支持模块的列表，请使用 **pam-config --list-modules** 命令。使用 **pam-config** 命令可以维护 PAM 配置文件。可将新模块添加到 PAM 配置，删除其他模块，或修改这些模块的选项。更改全局 PAM 配置文件时，无需手动调整单个应用程序的 PAM 设置。**pam-config** 的简单用例包括：

1. **自动生成全新的 Unix 样式 PAM 配置。** 让 **pam-config** 创建最简单的可行设置，您可以稍后再扩展。**pam-config --create** 命令创建简单的 Unix 身份验证配置。**pam-config** 不负责维护的现有配置文件将被重写，但会以 *.pam-config-backup 的形式保留备份副本。

2. **添加新的身份验证方法。** 可通过简单的 **pam-config --add --ldap** 命令将新的身份验证方法（例如 LDAP）添加到 PAM 模块堆栈。在适用的情况下，LDAP 将添加到所有 common-*-pc PAM 配置文件中。

3. **添加调试以进行测试。** 为确保新的身份验证过程按预期工作，请对所有 PAM 相关操作开启调试。**pam-config --add --ldap-debug** 为 LDAP 相关的 PAM 操作启用调试。

4. **查询您的设置。** 在最终应用您的新 PAM 设置之前，请检查该设置是否包含您要添加的所有选项。`pam-config --query --MODULE` 命令会列出所要查询的 PAM 模块的类型和选项。

5. **去除调试选项。** 最后，当您对设置性能完全满意时，请从设置中去除调试选项。`pam-config --delete --ldap-debug` 命令为 LDAP 身份验证禁用调试。如果您为其他模块添加了调试选项，请使用类似的命令关闭这些选项。

有关 `pam-config` 命令和可用选项的详细信息，请参见 `pam-config(8)` 的手册页。

5 手动配置 PAM

如果您偏向于手动创建或维护 PAM 配置文件，请确保对这些文件禁用 `pam-config`。

当您使用 `pam-config --create` 命令从头开始创建 PAM 配置文件时，此命令会创建从 `common-*` 到 `common-*-pc` 文件的符号链接。`pam-config` 仅修改 `common-*-pc` 配置文件。去除这些符号链接会有效禁用 `pam-config`，因为 `pam-config` 仅对 `common-*-pc` 文件运行，而在没有符号链接的情况下，这些文件不起作用。

警告：在配置中包含 pam_systemd.so

如果您要创建自己的 PAM 配置，请务必包含配置为 `session optional` 的 `pam_systemd.so`。不包含 `pam_systemd.so` 可能会导致 `systemd` 任务限制出现问题。有关细节，请参见 `pam_systemd.so` 的手册页。

6 将 SLE Micro 配置为需要 U2F 密钥才能进行本地登录

为了在本地登录 SLE Micro 期间提供更高的安全性，您可以使用 `pam-u2f` 框架以及 YubiKey 和安全密钥上的 U2F 功能配置双重身份验证。

要在 SLE Micro 系统上设置 U2F，您需要将密钥与您在 SLE Micro 上的帐户相关联。之后，将系统配置为使用该密钥。以下部分说明了此过程。

6.1 将 U2F 密钥与您的帐户关联

要将 U2F 密钥与您的帐户关联，请按以下步骤操作：

1. 登录计算机。
2. 插入 U2F 密钥。
3. 创建用于存储 U2F 密钥配置的目录：

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

4. 运行可输出配置行的 **pamu2fcfg** 命令：

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

5. 当设备开始闪烁时，触摸金属触点以确认关联。

建议使用备份 U2F 设备，可以通过运行以下命令来设置：

1. 运行：

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

2. 当设备开始闪烁时，触摸金属触点以确认关联。

为了提高安全性，可以将输出文件从默认位置移到需要 `sudo` 权限才能修改文件的目录。例如，将其移到 `/etc` 目录中。为此，请执行以下步骤：

1. 在 `/etc` 中创建一个目录：

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

2. 移动创建的文件：

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

注意：将 u2f_keys 放到非默认位置

如果将输出文件移到默认目录 (`$HOME/.config/Yubico/u2f_keys`) 以外的目录，则需要在 `/etc/pam.d/login` 文件中添加相应路径（如第 6.2 节 “更新 PAM 配置”所述）。

6.2 更新 PAM 配置

创建 U2F 密钥配置后，需要调整系统上的 PAM 配置。

1. 打开 `/etc/pam.d/login` 文件。
2. 将 `auth required pam_u2f.so` 一行添加到文件中，如下所示：

```
#%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. 如果将 `u2f_keys` 文件放到 `$HOME/.config/Yubico/u2f_keys` 以外的其他位置，则需要在 `/etc/pam.d/login` PAM 文件中使用 `authfile` 选项，如下所示：

```
#%PAM-1.0
auth     requisite pam_nologin.so
auth     include   common-auth
auth  required pam_u2f.so authfile=<PATH_TO_u2f_keys>
...
```

其中 `<PATH_TO_u2f_keys>` 是 `u2f_keys` 文件的绝对路径。

7 法律声明

A GNU 自由文档许可证

0. 导言

此许可证的目的是赋予手册、教科书或其他功能性的和有用的文档以“自由”：即保证每个人都有复制和再分发此类文档的有效自由，无论是否进行修改，也无论将其用于商业或非商业用途。其次，此许可证为作者和出版者保留了因工作获得声誉但不视为对他人所做修改负责的方式。

本许可证是一种“非盈利版权”，这意味着从该文档衍生的作品也必须是以同一方式自由的。它补充了 GNU 通用公共许可证（为自由软件设计的非盈利版权许可证）。

我们设计此许可证旨在将其用于免费软件的手册，因为免费软件需要自由文档：免费程序所附手册应具有与软件本身同样的自由。但是此许可证不限于软件手册；它可以用于任何文本作品，无论主题如何或它是否作为印刷书籍出版。建议本许可证主要用于目的是指导或参考的作品。

1. 适用性和定义

此许可证适用的对象：由版权所有者在其中明确声明可按照此许可证条款以任何媒体分发的任何手册和其他作品。此类声明授予在此处所述的条款和条件下使用该作品的全球无限期无版权许可证。下述“文档”指任何此类手册或作品。任何公众成员都是一个被许可人，以下称为“您”。如果您以需要版权法许可的任何方式复制、修改或分发该作品，则表示您接受该许可证。

该文档的“修改版本”表示包含该文档或其一部分（或者逐字复制或者有修改和/或翻译为另一语言）的任何作品。

“次要章节”是该文档的命名附录或扉页章节，专门讲述该文档的出版者或作者与该文档整个主题（或相关问题）的关系，不包含与整个主题相关的内容。（因此，如果该文档是数学课本的一部分，则辅助部分可能不说明任何数学问题。）这种关系可以是与主题或相关问题的历史联系，或与它们相关的法律、商业、哲学、伦理或政治地位。

在该文档基于此许可证项发布的声明中，“固定章节”是将其标题指定为固定章节标题的一些辅助章节。如果一个章节不适用上述辅助章节的定义，则不允许将其指定为固定章节。该文档可能不包含固定章节。如果该文档不标识任何固定章节，则表示没有固定章节。

在该文档基于此许可证项发布的声明中，“封页文本”是作为封面文本或封底文本列出的简短文本段落。封面文本最多 5 个单词，封底文本最多 25 个单词。

文档的“透明”副本是一个机器可读的副本，使用公众可以得到其规范的格式表达，这样的副本适合于使用通用文本编辑器、（对于像素构成的图像）通用绘图程序、（对于绘制的图形）广泛使用的绘画编辑器直接修改文档，也适用于输入到文本格式处理程序或自动翻译成各种适于适用于输入到文本格式处理程序的格式。一个用其他透明文件格式表示的副本，如果该格式的标记（或缺少标记）已经构成了对读者的后续修改的障碍，那么就是不透明的。表示实质性数量的文本的图像格式都是不透明的。不“透明”的副本称为“不透明”。

适于作为透明副本的格式的示例有：没有标记的纯 ASCII 文本、Texinfo 输入格式、LaTeX 输入格式、使用公共可用 DTD 的 SGML 或 XML，符合标准的简单 HTML、可以人为修改的 PostScript 或 PDF。透明图像格式的示例有 PNG、XCF 和 JPG。不透明的格式包括：仅可以被私有版权的字处理软件使用的私有版权格式、所用的 DTD 和/或处理工具不是广泛可用的的 SGML 或 XML、机器生成的 HTML、一些字处理器生成的只用于输出目的的 PostScript 或 PDF。

对于印刷书籍，“扉页”就是扉页本身以及随后的一些用于补充的页，显然本许可资料需要出现在扉页上。对于那些没有扉页的作品形式，“扉页”代表接近作品最突出标题的、在文本正文之前的文本。

“命名为 XYZ”的章节表示文档的一个特定的子单元，其标题就是 XYZ 或在括号中包含 XYZ 且后跟 XYZ 的其他语言翻译文本。（这里 XYZ 代表下面提及的特定章节名称，比如“致谢”、“题献”、“签名”或“历史”。）要在修改文档时对这类章节“保留标题”就是依据此定义保持这样一个“命名为 XYZ”的章节。

文档可能在文档遵照此许可证的声明后面包含免责声明。这些免责声明应作为参考信息包含在此许可证中，但是只能将其视作免责声明：这些免责声明暗指的任何其他含义均无效，且对此许可证的含义不产生任何影响。

2. 逐字复制

您可以用任何媒体复制并分发文档，无论是出于商业还是非商业目的，只要保证此许可证、版权声明和声称此许可证应用于文档的声明都完整地、无任何附加条件地存在于所有副本中。不能使用任何技术手段阻碍或控制您制作或发布的副本的阅读或再次复制。不过您可以在副本交易中得到报酬。如果发布足够多的副本，则您必须遵循下面第三节中的条件。

您也可以在如上的条件下出租副本和向公众放映副本。

3. 大量复制

如果您出版的文档印刷版副本（或是有印制封页的其他媒体副本）多于 100 份，而文档的许可证声明中要求有封页文本，则您必须将它清晰地置于封页之上，封面文本在封面上，封底文本在封底上。封面和封底上还必须标明您是这些副本的出版者。封面必须同等显著地完整展现标题的所有文字。您可以在封页上加入其他资料。改动仅限于封页的复制，只要保持文档的标题不变并满足这些条件，可以在其他方面被视为逐字复制。

如果需要加上的文本对于封面或封底过多，无法明显地表示，您应该在封页上列出前面的（在合理的前提下尽量多），把其他的放在邻近的页面上。

如果您出版或分发了超过 100 份文档的不透明副本，则必须在每个不透明副本中包含一份计算机可读的透明副本，或是在每个不透明副本中给出一个计算机网络地址，通过这个地址，网络公共用户可以使用标准网络协议下载文档的无任何附加资料的完整透明副本。如果您选择后

者，则必须在开始大量分发非透明副本的时候采用相当谨慎的步骤，保证透明副本在其所给出的位置在（直接或通过代理和零售商）分发最后一次该版本的非透明副本的时间之后一年之内始终是有效的。

在重新大量发布副本之前，请您（但不是必须）与文档的作者联系，以便他们可以有机会向您提供文档的更新版本。

4. 修改

在上述第 2、3 节的条件下，您可以复制和分发文档的修改版本，前提是严格按照此许可证发布修改后的文档，将修改版本用作文档，从而允许任何拥有此修改版副本的人执行分发或修改。另外，在修改版中，您需要做到如下几点：

A. 用于与文档以及以前各个版本（如果有，应该列在文档的“历史”章节中）显著不同的扉页（和封页，如果有）。如果那个版本的原始发行者允许的话，您可以使用和以前版本相同的标题。

B. 与作者一样，在扉页上列出承担修改版本中的修改的作者责任的一个或多个人或实体和至少五个文档的原作者（如果原作者不足五个就全部列出），除非他们免除了您的这个责任。

C. 与原来的发行者一样，在扉页上列出修改版的发行者的姓名。

D. 保持该文档的全部版权声明不变。

E. 在与其他版权声明邻近的位置加入恰当的针对您的修改的版权声明。

F. 在紧接着版权声明的位置加入许可声明，按照下面附录中给出的形式，以本许可证给公众授于是用修订版本的权利。

G. 保持原文档的许可声明中的全部不可变章节、封面文字和封底文字的声明不变。

H. 包含一份未作任何修改的本协议的副本。

I. 保持命名为“历史”的章节不变，保持它的标题不变，并在其中加入一项，至少声明扉页上的已修改版本的标题、年份、新作者和出版者。如果文档中没有命名为“历史”章节，则请新建它，并加入一项以声明原文档扉页上所列的标题、年份、作者与出版者，再在其后加入如上所说的描述修改版本的项。

J. 如果文档中有用于公共用户访问的文档透明副本的网址，则保持网址不变，并同样提供它所基于的以前文档版本的网址。这些网址可以放在“历史”章节。您可以不给出那些在原文档发行之前已经发行至少四年的版本给出的网址，或者该版本的发行者授权不列出网址。

K. 对于任何命名为“致谢”或“题献”的章节，保持其标题不变，并保持其全部内容以及对每位贡献者致谢和/或题献的语气不变。

L. 保持文档的所有固定章节不变，不改变它们的标题和内容。章节的编号或相当的内容不被认为是章节标题的一部分。

M. 删除命名为“签名”的章节。这样的章节不可以被包含在修改后的版本中。

N. 不要把任何现有章节重命名为“签名”或与任何不可变章节相冲突的标题。

O. 保持任何免责声明不变。

如果修改版本加入了新的符合次要章节定义的引言或附录章节，并且不含有从原文档中复制的内容，您可以选择将其标记为固定。如果需要这样做，则将它们的标题加入修改版本许可声明的不可变章节列表之中。这些标题必须和其他章节的标题相区分。

您可以加入一个命名为“签名”的章节，只要它只包含对您的修改版本由不同的各方给出的签名，例如书评或是声明文本已经被一个组织认定为一个标准的权威定义。

您可以加入一个最多 5 个字的段落作为封面文本和一个最多 25 个字的段落作为封底文本，将它们加入修改版本的封页文本列表末端。一个实体只可以添加（或编排）一段封面和一段封底文本。如果原文档已经为该封页（封面或封底）包含了封页文本，由您或您所代表的实体先前加入或排列的文本，不能再新加入一个，但您可以在原来的发行者的明确许可下替换掉原来的那个。

作者和发行者不能通过本许可证授权公众使用他们的名字推荐或暗示认可任何一个修改版本。

5. 组合文档

遵照第 4 节所说的修改版本的规定，您以将文档和其他文档合并并以本许可证发布，只要您在合并结果中包含原文档的所有不可变章节，对它们不加以任何改动，并在合并结果的许可声明中将它们全部列为不可变章节，而且维持原作者的免责声明不变。

合并作品仅需要包含一份此许可证，多个相同的固定章节可以由一个副本取代。如果有多个名称相同但内容不同的固定章节，通过在章节名称后面的括号中加上原作者或出版者的姓名（如果已知）来加以区别，或者使用唯一编号加以区别。并对合并作品许可声明中的固定章节列表中的章节标题做相同的调整。

在合并过程中，必须合并不同原始文档中任何命名为“历史”的章节，从而形成新的命名为“历史”的章节；类似地，还要合并命名为“致谢”和“题献”的章节。必须删除所有命名为“签名”的章节。

6. 文档的合集

您可以制作一个文档和其他文档的合集，在本许可证下发布，并在合集中将不同文档中的多个本许可证的副本以一个单独的副本来代替，只要您在文档的其他方面遵循本许可证的逐字复制的条款即可。

您可以从一个这样的合集中提取一个单独的文档，并将它在本许可证下单独发布，只要您想这个提取出的文档中加入一份本许可证的副本，并在文档的其他方面遵循本许可证的逐字复制的原则。

7. 独立作品的聚合体

将文档或其派生品以及其他独立和无关文档或作品编撰在一个储存卷中或分发媒体上，这称为文档的“聚合体”，前提是编撰成品的著作权对其使用者的法律权限的限制未超出各个独立作品的许可范围。当基于此许可证发布的文档包含在一个聚合体中时，此许可证不适用于聚合体中的本非该文档派生作品的其他作品。

如果第 3 节中的封页文本要求适用于这些文档的副本，则若文档在聚合体中所占的比重小于全作品的一半，文档的封页文本可以放置在聚合体内包含文档部分的封页上，或是电子文档中的等效部分。否则，它必须位于整个聚合体的印刷的封页上。

8. 翻译

翻译被视为一种修改，因此您可以根据第 4 节的条款分发文档的翻译。将固定章节替换为翻译内容需要经得其版权所有者的特别许可，但除了这些固定章节的原始版本之外，您还可以包含一部分或所有固定章节的翻译。您可以包含一个此许可证以及所有许可证声明和免责声明的翻译版本，前提是同时包含它们的原始英文版本。当翻译版本和英文版发生冲突的时候，原始版本有效。

如果在文档中有命名为“致谢”、“题献”或“历史”的章节，保持标题（第 1 节）的要求（第 4 节）恰恰需要更换实际的标题。

9. 终止

除非此许可证中有明确规定，否则您不能对该文档进行复制、修改、分授许可或分发。在此许可证规定外对该文档所进行的任何复制、修改、分授许可或分发都是无效的，并且将自动终止您在此许可证下所拥有的权利。但是，对于在此许可证的规定下从您这里获得副本或权利的各方，只要其完全遵守此许可证的规定，其许可证将不会被终止。

10. 本许可的未来修订版本

自由软件基金会有时会发布 GNU 自由文档许可证的新的修订版版本。这些新版本的主旨和精神与当前版本是一致的，但在解决新问题的具体细节方面可能有所不同。请参见 https://www.gnu.org/copyleft/。

许可证的每个版本都有一个不同的版本号。如果文档指定了适用于它的此许可证“或任何后续版本”的特定带编号版本，则您可以选择遵从指定版本或自由软件基金会发布的任何随后版本（非草稿）的条款和条件。如果文档没有指定此许可证的版本号，您可以选择自由软件基金会发布的任何许可证版本（非草稿）。

附录：如何针对您的文档使用此许可证

```
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”.
```

如果您有固定章节、封面文本和封底文本，请将“with...Texts”部分替换为：

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

如果有不可变章节而没有封页文本，或这三种内容（不可变章节、封面文本、封底文本）的任何其他组合，请合并这两个备选项以适应您的情况。

如果您的文档包含不一般的程序代码示例，建议同时选择自由软件许可证（如 GNU 通用公共许可证）发布这些示例，以允许它们可以用于自由软件。