

# 使用 `transactional-update` 命令管理 SLE Micro

## 解释

使用 `transactional-update` 命令可以修改只读文件系统。修改是在单独的快照中执行的，并且在您引导进入快照之前不会影响正在运行的系统。

## 原因

您想要管理 SLE Micro 及其更新，并最大程度地减少因更新失败而导致系统停机的风险，同时确保可轻松回滚。

## 工作量

在 30 分钟内即可了解 `transactional-update` 命令。

## 目标

您将了解的 `transactional-update` 工作原理，以及如何使用它来管理系统。

## 要求

- 一个正在运行的 SLE Micro 实例
- `root` 特权

出版日期：2025 年 12 月 11 日

## 目录

- 1 事务更新 3
- 2 **transactional-update** 命令的用法 4
- 3 在不重引导的情况下应用多项更改 6
- 4 执行快照清理 8
- 5 注册产品 9
- 6 管理软件包 10
- 7 执行系统回滚 11
- 8 管理自动事务更新 13
- 9 法律声明 15
- A GNU 自由文档许可证 15

# 1 事务更新

## 1.1 什么是事务更新？

为了保持基础操作系统的稳定性和一致性，SLE Micro 使用只读的根文件系统。因此，您无法直接对根文件系统执行更改操作，例如使用 **zypper** 命令。但是，SLE Micro 引入了**事务更新**功能，可让您对根文件系统应用一项或多项更改。

**transactional-update** 的默认行为是在每次更改发生后基于当前根文件系统创建新快照。要应用更改，就需要重引导主机。在不重引导的情况下，您无法多次运行 **transactional-update** 命令将其他更改添加到快照。此操作将创建独立的快照，其中不包含先前快照中的更改。

## 1.2 事务更新的工作原理

每次调用 **transactional-update** 命令来更改系统时（无论是安装软件包、执行更新还是应用补丁），系统都会执行以下操作：

### 过程 1：修改根文件系统

1. 基于当前的根文件系统或您指定的快照创建新的读写快照。
2. 应用所有更改（执行更新、修补操作或安装软件包）。
3. 将快照切换回只读模式。
4. 如果所有更改都成功应用，则将新的根文件系统快照设为默认快照。
5. 重引导之后，系统引导进入新的快照。

## 1.3 事务更新的优点

- 事务更新属于原子操作，只有在成功完成后才会应用更新。
- 更改将在单独的快照中应用，因此不影响正在运行的系统。
- 更改可以轻松回滚。

## 1.4 **transactional-update** 命令内的环境

每次运行 **transactional-update** 命令时，更改都是在新快照中进行。快照中的环境可能不同于您用于运行 **transactional-update** 命令的外壳中的环境。例如，当前工作目录 (`$PWD`) 不是设为您运行 **transactional-update** 时所在的目录，而是设为 `/`。

从快照内无法访问 `/var` 目录。此目录也不会包含在快照中。但是，有些目录虽然没有包含在快照中，却能从 **transactional-update** 环境中访问，例如 `/root` 目录。

## 2 **transactional-update** 命令的用法

使用 **transactional-update** 命令可以原子方式安装或去除更新。如果所有更新都能成功安装，则会应用更新。**transactional-update** 创建系统的快照，并使用该快照来更新系统。日后，您可以恢复该快照。重引导后，所有更改才会生效。

**transactional-update** 命令的语法如下：

```
transactional-  
update [option] [general_command] [package_command] standalone_command
```



### 注意：不带参数运行 **transactional-update**

如果运行 **transactional-update** 命令时未指定任何参数或选项，系统会更新自身。

下面进一步说明了可能的命令参数。

#### **transactional-update** 选项

`--interactive, -i`

可以与软件包命令结合使用来开启交互模式。

`--non-interactive, -n`

可以与软件包命令结合使用来开启非交互模式。

--continue [number], -c

使用 --continue 选项可在不重引导的情况下对根文件系统进行多项更改。有关更多详细信息，请参见第 3 节“在不重引导的情况下应用多项更改”。

--continue 选项的另一个有用功能是，可让您选择任一现有快照作为新快照的基础。以下示例演示如何运行 **transactional-update** 在基于快照 13 的某个快照中安装新软件包，然后再次运行该命令来安装另一个软件包：

```
> sudo transactional-update pkg install package_1
```

```
> sudo transactional-update --continue 13 pkg install package_2
```

--no-selfupdate

禁用 **transactional-update** 的自我更新功能。

--drop-if-no-change, -d

如果未对根文件系统进行任何更改，则丢弃 **transactional-update** 创建的快照。如果对 /etc 目录进行了更改，这些命令会合并回当前文件系统。

--quiet

**transactional-update** 命令的结果将不输出到 stdout。

--help, -h

列显 **transactional-update** 命令的相关帮助。

--version

显示 **transactional-update** 命令的版本。

## 2.1 常规命令

本节列出了 **transactional-update** 的通用命令。

### **grub.cfg**

使用此命令可重建 GRUB 引导加载程序配置文件。

### **bootloader**

该命令会重新安装引导加载程序。

## **initrd**

使用该命令可重建 initrd。

## **kdump**

如果对硬件或存储设备进行了更改，就可能需要重建 Kdump initrd。

## **shell**

退出前在新快照中打开读写外壳。该命令通常用于执行调试。

## **reboot**

当 transactional-update 命令完成后，系统会重引导。

## **run <command>**

在新快照中运行提供的命令。

## **setup-selinux**

安装并启用针对性的 SELinux 策略。

# 3 在不重引导的情况下应用多项更改

**transactional-update** 命令将更改应用于事务系统上的根文件系统。默认行为是在每次更改发生后基于当前根文件系统创建新快照，然后重引导以应用更改。

要在不重引导的情况下对根文件系统进行多项更改，可以采用多种方法，下面几节中会加以说明：

## 3.1 **transactional-update --continue** 选项

使用带 **--continue** 选项的 **transactional-update** 命令可在不重引导的情况下进行多项更改。每次运行该命令时，都会创建一个不同的快照，该快照会包含先前快照中的所有更改以及您做出的新更改。最终的快照将会包含所有更改。要应用这些更改，请重引导系统，然后最终快照将成为新的根文件系统。

## 3.2 **transactional-update run** 命令

**transactional-update run** 命令通常只会运行一条命令，但您可以使用它在一个事务性会话中运行多条命令，只需在命令外壳（如 **bash**）中串联这些命令即可，例如：

```
> sudo transactional-update run bash -c 'ls && date; if [ true ]; then echo -n "Hello "; echo '\''world'\''; fi'
```



### 注意

**transactional-update run** 命令具有与第 3.3 节 “**transactional-update 外壳**” 中所述的 **transactional-update shell** 命令相同的限制，只不过输入的命令会记录在 `/var/log/transactional-update.log` 文件中。

## 3.3 **transactional-update** 外壳

**transactional-update shell** 命令会在 `transactional-update` 环境中打开一个外壳。在该外壳中，您几乎可以输入任何 Linux 命令来对文件系统进行更改，例如，使用 **zypper** 命令安装多个软件包，或对只读文件系统中的文件进行更改。您还可以校验先前使用 **transactional-update** 命令进行的更改是否正确。



### 重要

事务外壳具有一些限制。例如，您无法使用 `systemd` 命令控制 `start` 或 `stop` 服务，也无法修改 `/var` 分区，因为该分区未挂载。此外，在外壳会话期间输入的命令不会记录到 `/var/log/transactional-update.log` 文件中。

您对文件系统进行的所有更改都是单个快照的组成部分。完成对文件系统的更改并使用 **exit** 命令退出外壳后，需要重引导主机才能应用更改。

## 4 执行快照清理

可以使用 **transactional-update** 清理未使用的文件系统快照和未引用的 /etc 叠加目录。

**transactional-update** 可识别以下清理命令：

### **cleanup-snapshots**

该命令将所有未使用的快照标记为由 Snapper 去除。

### **cleanup-overlays**

该命令去除 /var/lib/overlay 目录中 /etc 的所有未使用叠加层。

### **cleanup**

该命令结合了 **cleanup-snapshots** 和 **cleanup-overlays** 命令的功能。

## 4.1 清理的工作原理

如果您运行 **transactional-update cleanup** 命令，将为所有没有清理算法的旧快照设置一个算法。此外还会标记所有重要快照。该命令还会去除 /var/lib/overlay 中所有未引用的（因此未使用）/etc 叠加目录。

将通过以下参数根据 /etc/snapper/configs/root 中配置的规则删除设置了 number 清理算法的快照：

### **NUMBER\_MIN\_AGE**

定义可自动去除的快照的最短期限（以秒为单位）。

### **NUMBER\_LIMIT/NUMBER\_LIMIT\_IMPORTANT**

定义存储的快照的最大计数。清理算法将删除超过指定最大值的快照，而不考虑快照和文件系统空间的状态如何。这些算法还会删除超过最小值的快照，直到达到快照和文件系统的限制。

systemd 也会定期执行快照清理。



## 5 注册产品

可以使用 **`transactional-update register`** 命令来处理与产品注册及其订阅管理相关的所有任务。您可以提供以下选项：

### **`--list-extensions`**

该命令与此选项结合使用会列出适用于您的系统的扩展。您可以通过输出结果查找产品标识符以激活产品。

### **`-p, --product`**

使用此选项可指定要激活的产品。产品标识符的格式如下：<名称>/<版本>/<体系结构>，例如，`sle-module-live-patching/15.3/x86_64`。对应的命令采用以下格式：

```
> sudo transactional-update register -p sle-module-live-patching/15.3/x86_64
```

### **`-r, --regcode`**

使用提供的注册代码注册系统。该命令会注册订阅并启用软件储存库。

### **`-d, --de-register`**

该选项会取消注册系统，如果将其与 `-p` 选项结合使用，则会取消注册扩展。

### **`-e, --email`**

指定用于在 SUSE Customer Center 中注册系统的电子邮件地址。

### **`--url`**

指定注册服务器的 URL。URL 存储在配置中，并在后续命令调用中使用。例如：

```
> sudo transactional-update register --url https://scc.suse.com
```

### **`-s, --status`**

以 JSON 格式显示当前的注册状态。

### **`--write-config`**

将提供的选项值写入 `/etc/SUSEConnect` 配置文件。

### **`--cleanup`**

去除旧系统身份凭证。

`--version`

列显版本。

`--help`

显示命令的用法。

## 6 管理软件包

可以使用 **`transactional-update`** 来安装、更新或删除软件包。

SLE Micro 从产品注册后提供的储存库中获取软件包。

**`transactional-update`** 使用以下命令来管理软件包。



### 提示：**`pkg`** 命令和 Zypper 选项

对于 **`transactional-update pkg`** 命令，可以使用与所用子命令对应的任何 Zypper 选项。例如，**`zypper install`** 识别的所有选项也能由 **`transactional-update pkg install`** 识别。

### **`pkg install`**

使用 **`zypper install`** 命令从可用渠道安装单个软件包。此命令还可用于安装程序临时修复 (PTF) RPM 文件。此命令的默认选项为 `--interactive`。

```
> sudo transactional-update pkg install package_name
```

或

```
> sudo transactional-update pkg install rpm1 rpm2
```

或者，要安装软件集，请使用以下命令：

```
> sudo transactional-update pkg install -t pattern pattern_name
```

### **`pkg remove`**

使用 **`zypper remove`** 命令从活动快照中去除单个软件包。此命令还可用于去除 PTF RPM 文件。此命令的默认选项为 `--interactive`。

```
> sudo transactional-update pkg remove package_name
```

### **pkg update**

使用 **zypper update** 命令更新活动快照中的单个软件包。只能更新属于基础文件系统快照的软件包。此命令的默认选项为 **--interactive**。

```
> sudo transactional-update pkg update package_name
```

### **patch**

检查并安装可用的补丁。此命令的默认选项为 **--non-interactive**。

### **dup**

执行系统升级。此命令的默认选项为 **--non-interactive**。

### **up**

将安装的软件包更新为更高版本。此命令的默认选项为 **--non-interactive**。

### **migration**

该命令会将系统迁移到所选目标。如果系统已通过 SUSE Customer Center 注册，该命令通常用于升级系统。

## 7 执行系统回滚

GRUB 2 支持从 btrfs 快照引导，因此当新的快照无法正常工作时，您可以使用任何旧的功能快照。

引导快照时，快照中包含的文件系统部分会加载为只读状态；从快照中排除的所有其他文件系统部分会加载为读写状态，并且可以修改。



### **提示：回滚到特定的安装状态**

在初始系统安装结束时会创建初始可引导快照。您随时可以通过引导此快照返回到该状态。该快照可通过 **first root file system** 说明识别。

可以用两种方法执行系统回滚。

- 在正在运行的系统中，可以设置默认快照，具体请参见[过程 2 “从正在运行的系统回滚”](#)。
- 尤其是在当前快照损坏的情况下，您可以引导进入新快照，并将其设置为默认快照。有关细节，请参见[过程 3 “回滚到正常工作的快照”](#)。

如果当前快照可以正常运行，则您可以使用以下过程进行系统回滚。

#### 过程 2：从正在运行的系统回滚

1. 识别要设置为默认快照的快照并记下其编号。

```
> sudo snapper list
```

2. 将该快照设置为默认快照。

```
> sudo transactional-update rollback snapshot_number
```

如果省略 `snapshot_number`，则会将当前快照设置为默认快照。



#### 提示：设置上次正常工作的快照

要将上次正常工作的快照设置为默认快照，请运行 `rollback last`。

3. 重引导系统以引导进入新的默认快照。

如果当前快照损坏并且您无法引导进入其中，请使用以下过程。

#### 过程 3：回滚到正常工作的快照

1. 重引导系统并选择 Start bootloader from a read-only snapshot。
2. 选择要引导的快照。快照根据创建日期排序，最新快照列在最前面。
3. 登录到您的系统，并检查是否一切按预期进行。写入到从快照排除的目录中的数据将保持不变。
4. 如果您引导进入的快照不适合回滚，请重引导系统并选择另一个快照。

如果快照按预期工作，您可以运行以下命令来执行回滚：

```
> sudo transactional-update rollback
```

然后重引导。

## 8 管理自动事务更新

自动更新由每天运行一次的 **systemd.timer** 控制。此操作会应用所有更新，并告知 **rebootmgrd** 应重引导计算机。您可以调整更新运行时间，具体请参见 `systemd.timer(5)` 文档。

### 8.1 禁用自动更新

默认已启用自动更新。但是，您可以使用以下命令将其禁用：

```
> sudo systemctl --now disable transactional-update.timer
```

### 8.2 配置更新失败通知

当自动 **transactional-update** 失败时，将删除发生失败的快照。同时，系统可能会重引导，在这种情况下，您无法知道上次的自动更新已失败。为此，您可以配置 **systemd** 服务，以便在自动 **transactional-update** 失败时收到通知。配置该服务的过程概括如下：

- 安装系统中缺少的必需软件包。有关细节，请参见第 8.2.1 节 “安装必需的软件包”。
- 配置 **systemd-status-mail** 服务。有关细节，请参见第 8.2.2 节 “配置 **systemd-status-mail** 服务”。

#### 8.2.1 安装必需的软件包

必须安装 **mailx** 和 **systemd-status-mail** 软件包才能配置通知。默认情况下，系统中已安装这些软件包。但是，如果未安装这些软件包，可运行以下命令来安装：

```
> sudo transactional-update pkg in systemd-status-mail mailx
```

重新启动系统。

## 8.2.2 配置 `systemd-status-mail` 服务

要配置 `systemd-status-mail` 服务，可以创建配置文件，或使用 `jeos-config` 工具。

### 8.2.2.1 使用 `jeos-config` 配置服务

要配置电子邮件通知，可如下所述使用 `jeos-config` 工具。

1. 要打开配置窗口，请运行以下命令：

```
> sudo jeos-config status_mail
```

2. 在对话框中，根据需要配置项目。
3. 单击确定进行确认。

### 8.2.2.2 通过编辑配置文件来配置服务

默认配置文件位于 `/usr/etc/default/systemd-status-mail` 中。要修改该文件，请在 `/etc/default/` 中创建副本并编辑以下项目：

#### ADDRESS

必填项。指定要将通知发送到的电子邮件地址。例如：

```
ADDRESS="tux@example.com"
```

#### FROM

通知邮件的发件人电子邮件地址。确保该地址有效。例如：

```
FROM="geeko@example.com"
```

#### MAILER

用于发送通知的邮件应用程序类型。如下所示使用 `mailx` 值：

```
MAILER="mailx"
```

#### RELAYHOST

指定 `mailx` 使用的邮件中继。

```
RELAYHOST="mail.example.com:587"
```

## MAILX\_OPTIONS

指定必要的选项，以确保邮件提供商接受通知邮件。

```
MAILX_OPTIONS="-Sverbose -Ssmtp-use-starttls -Ssmtp-auth=login -Ssmtp-auth-user='tux@example.com' -Ssmtp-auth-password='TopSecret' "
```

## 9 法律声明

版权所有 © 2006–2025 SUSE LLC 和贡献者。保留所有权利。

根据 GNU 自由文档许可证 (GNU Free Documentation License) 版本 1.2 或（根据您的选择）版本 1.3 中的条款，在此授予您复制、分发和/或修改本文档的权限；本版权声明和许可证附带不可变部分。许可版本 1.2 的副本包含在题为“GNU Free Documentation License”的部分。

有关 SUSE 商标，请参见 <https://www.suse.com/company/legal/>。所有其他第三方商标分别为相应所有者的财产。商标符号（®、™ 等）代表 SUSE 及其关联公司的商标。星号 (\*) 代表第三方商标。

本指南力求涵盖所有细节，但这不能确保本指南准确无误。SUSE LLC 及其关联公司、作者和译者对于可能出现的错误或由此造成的后果皆不承担责任。

## A GNU 自由文档许可证

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. 允许任何人复制和分发此许可证文档的逐字副本，但禁止对其进行更改。

### 0. 引言

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 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。