

SLSA: Securing the Software Supply Chain

All SUSE Products

Marcus Meissner, Distinguished Engineer Solutions Security (SUSE)

Jana Jaeger, Project Manager Technical Documentation (SUSE)

SLSA: Securing the Software Supply Chain

This document details how SUSE, as a long-time champion and expert of software supply chain security, prepares for SLSA L4 compliance.

Disclaimer: This document is part of the SUSE Best Practices series. All documents published in this series were contributed voluntarily by SUSE employees and by third parties. If not stated otherwise inside the document, the articles are intended only to be one example of how a particular action could be taken. Also, SUSE cannot verify either that the actions described in the articles do what they claim to do or that they do not have unintended consequences. All information found in this document has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Therefore, we need to specifically state that neither SUSE LLC, its affiliates, the authors, nor the translators may be held liable for possible errors or the consequences thereof.

Contents

- 1 Software supply chain security at SUSE 4
- 2 Background: Threats to the supply chain and their mitigation 4
- 3 SLSA components 6
- 4 Outlook 16
- 5 Legal notice 17
- 6 GNU Free Documentation License 18

1 Software supply chain security at SUSE

Securing our software supply chain is a top priority for SUSE to protect our customers from security risks, known and zero-day vulnerabilities. Ensuring that no threat actor can inject malicious code into our build service systems is attested by industry-leading security certifications. Our teams continually work to certify all SUSE products, and develop security solutions to offer our customers the highest level of trust and reliability.

A new industry standardization effort named SLSA (Supply chain Levels for Software Artifacts), started by Google and driven by several industry stakeholders, aims to protect the integrity of the software supply chain.

SLSA defines four levels of assurance, going from basic requirements at level 1 to strict rules and documentation requirements at level 4.

While the SLSA standard is still in development, SUSE already considers it as a great representation of needs for a secure product build environment, and we are adjusting our processes and tooling to meet the requirements of the highest assurance level 4.

2 Background: Threats to the supply chain and their mitigation

The software supply chain contains several known points of attack. They are illustrated by the graphic below. For an exhaustive list of threats and their mitigation, refer to <https://slsa.dev/spec/v0.1/threats>.

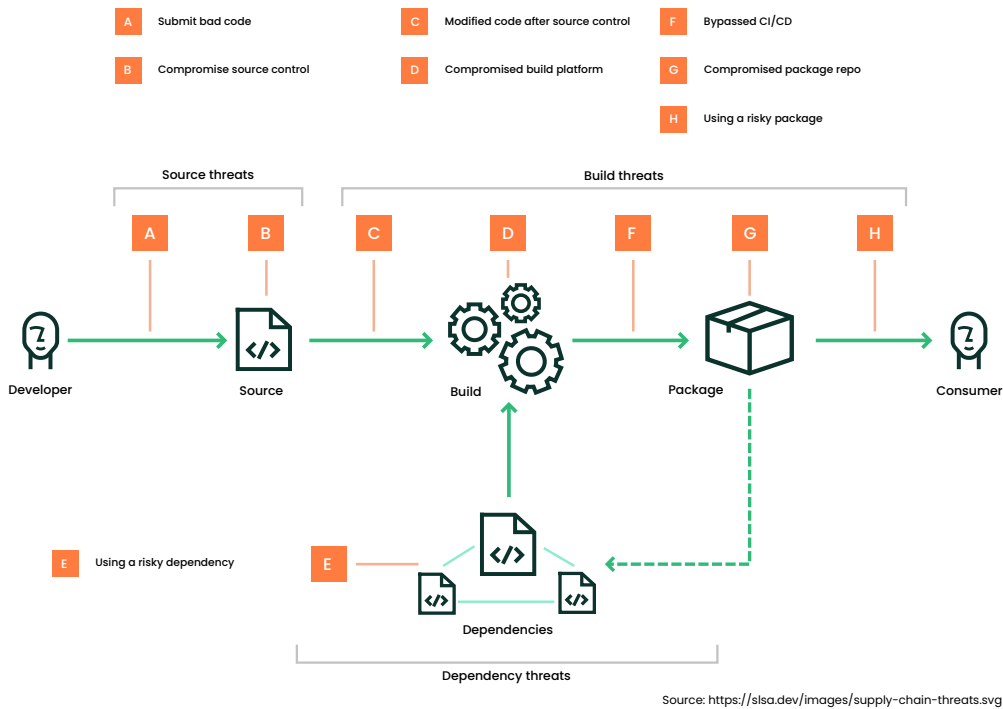


FIGURE 1: SOFTWARE SUPPLY CHAIN AND POSSIBLE THREATS

Source threats

Typical source code threats include bad code that introduces vulnerabilities or a compromised source control system. To address bad code injection, SLSA mandates two-person reviews. To prevent source code from getting compromised, SLSA mandates strong measures to secure the source control systems.

Build threats

Build threats include code commits to the build that were not tracked by the source control system, a compromised build platform, bypassing the CI/CD system, a compromised package repository, and injecting bad packages. Most build threats are mitigated by maintaining a controlled build environment, where each build is also fully encapsulated on its own, not influenceable from outside, or even reproducible. To prove this to the outside, detailed provenance data can be generated, which allows for the external inspection of the builds. Strong security controls ensure that the build platform is not easily compromised.

Dependency threats

Dependency threats come into play where risky dependencies are used. SLSA addresses this kind of threat by mandating provenance for all artifacts (files, git commits, directories of files, container images, ...), so one would have an indication that this dependency was not built from the proper builder or out of the designated GitHub repository.

3 SLSA components

SLSA is designed to standardize the four key components of a software supply chain. Learn about these components, the respective SLSA requirements, and whether and how SUSE's build process meets these requirements today.

For a complete summary of SLSA requirements, refer to the official SLSA project pages: <https://slsa.dev/spec/v0.1/requirements>.



Note: Reading the requirement tables

To understand the SLSA requirement tables, check the following legend:

✓	Yes
✗	No
○	Best effort

3.1 Source code management

Keeping source code integrity is the key aspect of supply chain integrity. Source code integrity needs to be defended against all threats originating from insider or outsider attacks.

SLSA mandates the following criteria to be met to ensure source code integrity:

TABLE 1: SLSA4 SOURCE REQUIREMENTS AND SUSE'S OBS

Requirement	Required at SLSA4	Met by OBS/SUSE
Version-controlled	✓	✓
Verified history	✓	✓
Retained indefinitely	✓	✓
Two-person reviewed	✓	✓

See how SUSE's current software building pipeline that uses the Open Build Service (OBS) meets the above requirements:

Version-controlled source

- **SLSA L4 requirement.** Changes to the source must be tracked in a version control system which, in turn, needs to provide a *change history* that contains both the identities of the uploader and the reviewers, time stamps of the reviews/submissions, a comprehensive description of the change, its content, and parent reviews. The version control system must also provide an *immutable reference* to each revision. This requirement is met by most current version control systems, for example, git.
- **SUSE's processes.** OBS meets this requirement as it assigns numeric identifiers to commits. The commit stores information about the author, the commit time, a comment describing the commit, and other information. The commit also contains an identification of the source content, like the *tree* object in git. OBS uses an MD5 sum for identification purposes, but also makes sure that there is no collision with other source content.

Verified revision history

- **SLSA L4 requirement.** The history of a revision must be verifiable, that is, it must contain a time stamp and carry at least one strongly authenticated actor identity. Actors in this context are the author, submitter, or reviewer of the revision. Identities themselves must be verifiable and use a two-step verification process, or something similar.
- **SUSE's processes.** Each commit stores information about the author and commit time. It also contains a comment describing the commit and other information. Identities of all actors are verifiable and use two-factor authentication.

Revision and change history are retained indefinitely

- **SLSA L4 requirement.** The revision and its change history must be preserved indefinitely and cannot be deleted. Exceptions can be made if there is a legal or policy requirement that mandates the obliteration of this revision/change history.
- **SUSE's processes.** SUSE retains all sources, but also any shipped binary indefinitely. A retraction of a shipped update from the customer's channels does not lead to its removal from the build system. Furthermore, all source references and binary shipments including bug tracker references (CVE, bugzilla) are tracked by the build system for each shipment channel.

Two-person reviewed

- **SLSA L4 requirement.** Two trusted persons (see above) need to agree to every change in the revision history before it is submitted. Both of them must be strongly authenticated.
- **SUSE's processes.** The workflows for creating a new product and delivering maintenance updates involve multiple parties, like the core code review team and the maintenance or product release managers as a minimum. Furthermore, there are reviews by subject matter experts and additional checks for quality assurance and legal aspects. These reviews are enforced by the OBS, and a single decline rejects the entire release process.

3.2 Building and build system

The next part of the integrity chain is the actual build process that turns sources to binaries. The entire build process must be secured against any kind of unknown or outside influence to avoid possible tampering with the builds. Builds must be reproducible to allow verification and checking of build results.

TABLE 2: SLSA4 BUILD (PROCESS) REQUIREMENTS AND SUSE'S OBS

Requirement	Required at SLSA4	Achieved
Scripted build	✓	✓
Build service	✓	✓

Requirement	Required at SLSA4	Achieved
Build as code	✓	✓
Ephemeral environment	✓	✓
Isolated	✓	✓
Parameterless	✓	✓
Hermetic	✓	✓
Reproducible	○	✗

See how SUSE's current software building pipeline with the Open Build Service (OBS) meets the above requirements:

Scripted build

- **SLSA L4 requirement.** Define all build steps in any kind of build script. Manual interaction is limited to invoking the build script itself.
- **SUSE's processes.** This is achieved by the SUSE build script also used by the OBS. Even the decision to invoke a build is made by OBS based on the submitted code changes or other builds.

Build service

- **SLSA L4 requirement.** Some build service runs the build steps. Running builds on developers' workstations does not qualify as a build service.
- **SUSE's processes.** Met by OBS (runs in a build cluster in the SUSE data center).

Build as code

- **SLSA L4 requirement.** Store the build definition and configuration in text files and keep them under version control. They must be executed by the build service.
- **SUSE's processes.** The recipe files defining the build process are part of the sources of the individual packages, for example the RPM spec file, its sources and patches, or image and container description files. The build environment configuration (*project config*) is also under source control in OBS.

Ephemeral environment

- **SLSA L4 requirement.** Build steps are run in an ephemeral environment such as a VM or container. These environments must not be recycled from a prior build and must be provisioned solely for this build.
- **SUSE's processes.** Met by OBS by using an isolated KVM instance for each build. Access to the outside is not possible (no network), only the sources and binaries prepared by OBS can be used. OBS also decides which pieces of the build artifacts are used. This includes running Linux kernel in the VM.

Isolated

- **SLSA L4 requirement.** Build steps must be run isolated from each other. No build instance, either prior or concurrent, may influence others.
- **SUSE's processes.** Met by OBS (builds in KVM).

Parameterless

- **SLSA L4 requirement.** The build is fully and solely defined through the build script and nothing else. Builds cannot be affected by any user input/parameters (except for build entry point and top-level source location).
- **SUSE's processes.** The build happens completely decoupled from any user interaction. Any parameter must be part of any source submission. No input is possible during the build, which is ensured by the KVM setup. The only output during the build is the build log.

Hermetic

- **SLSA L4 requirement.** All build steps must be shielded from any kind of external or internal influences. All transitive build steps, sources and dependencies are declared up front with immutable references and run without network access.
- **SUSE's processes.** Met by OBS (builds in a KVM guest without network, and everything required for the build is injected before the build instance is brought up).

Reproducible

- **SLSA L4 requirement.** When a build is rerun with identical input, it must yield identical results (bit-for-bit). Builds that cannot meet this requirement must provide a justification why they cannot be made reproducible. This requirement is *best effort*.
- **SUSE's processes.** The OBS build system tracks all used binaries for each build and can reproduce the build environment of any released binary. The binaries used are also referenced in in-toto provenance files and made available together with the sources starting from SUSE Linux Enterprise 15 SP4 builds. Older builds may have missing binaries because of the nature of the bootstrapping process. It is notable that the SUSE Linux Enterprise 15 code base is not enforcing binary identical reproducibility yet. Instead, builds are compared and known good differences are accepted (for example time stamps or build host name). This validation is done by the code in the build-compare package.

3.3 Provenance

A key aspect of supply chain security is the ability to prove that a build has been completed / a package built according to all SLSA4-mandated requirements. This *provenance* is established by means of providing metadata that proves compliance to SLSA.

The SLSA requirements for provenance can be grouped into:

- process requirements on provenance generation and consumption
- requirements on the contents of the provenance

For provenance generation and consumption, SLSA mandates the following requirements:

TABLE 3: SLSA4 BUILD PROVENANCE PROCESS REQUIREMENTS AND SUSE'S OBS

Requirement	Required at SLSA4	Achieved
Available	✓	✓
Authenticated	✓	✓
Service-generated	✓	✓
Non-falsifiable	✓	✓
Dependencies complete	✓	✓

Learn how SUSE's OBS meets the SLSA4 provenance process requirements:

Available

- SLSA L4 requirement. Provide provenance in a format that is accepted by the consumer.
- SUSE's processes. Provided by SUSE.

Authenticated

- SLSA L4 requirement. Consumer can verify the authenticity and integrity of the provenance.
- SUSE's processes. The provenance records are signed by OBS after the build.

Service-generated

- SLSA L4 requirement. Provenance data must be provided by the build service.
- SUSE's processes. The build service creates the provenance file.

Non-falsifiable

- SLSA L4 requirement. Build service users cannot falsify provenance.
- SUSE's processes. Signing is done outside of the build in the build service and cannot be influenced by the user.

Dependencies complete

- SLSA L4 requirement. Record all build dependencies that were available at the build time (includes initial state of container, VM, build worker or machine).
- SUSE's processes. The component downloading the build environment generates the dependency part of the provenance file. Nothing else is available for the build process.

For provenance contents, SLSA mandates the following requirements:

TABLE 4: SLSA4 BUILD PROVENANCE CONTENT REQUIREMENTS AND SUSE'S OBS

Requirement	Required at SLSA4	Achieved
Identifies artifact	✓	✓

Requirement	Required at SLSA4	Achieved
Identifies builder	✓	✓
Identifies build instructions	✓	✓
Identifies source code	✓	✓
Identifies entry point	✓	✓
Includes all build parameters	✓	✓
Includes all transitive dependencies	✓	✓
Includes reproducible info	✓	✓
Includes metadata	○	✓

Learn how SUSE's OBS meets the SLSA4 provenance content requirements:

Identifies artifact

- **SLSA L4 requirement.** Identify the output artifact by at least one cryptographic hash included in the provenance.
- **SUSE's processes.** OBS stores SHA2 hashes of all build artifacts in the provenance files.

Identifies builder

- **SLSA L4 requirement.** Provide an identifier for the entity that performed the build with the provenance.
- **SUSE's processes.** The ID of the builder is added by OBS into the provenance files.

Identifies build instructions

- **SLSA L4 requirement.** Provide the top-level instructions that were used to execute the build.
- **SUSE's processes.** The build instructions are the RPM spec file, which is specified in the provenance files.

Identifies source code

- **SLSA L4 requirement.** Include the repository origin/origins for the source code used in the build in the provenance.
- **SUSE's processes.** Met by OBS.

Identifies entry point

- **SLSA L4 requirement.** Identify the entry point of the build definition used to drive the build (for example the source repo the configuration was taken from).
- **SUSE's processes.** The provenance files specify the RPM spec file or container description file as entry point.

Includes all build parameters

- **SLSA L4 requirement.** Include all build parameters that are under a user's control in the provenance.
- **SUSE's processes.** Met by OBS.

Includes all transitive dependencies

- **SLSA L4 requirement.** Include list of all transitive dependencies in the provenance.
- **SUSE's processes.** Met by OBS.

Includes reproducible info

- **SLSA L4 requirement.** Include information on whether the build was intended to be reproducible. If so, also include all information needed to reproduce the build.
- **SUSE's processes.** Met by OBS, but builds are currently marked as non-reproducible.

Includes metadata

- **SLSA L4 requirement.** Include metadata in the provenance to help with debugging and investigations.
- **SUSE's processes.** OBS includes additional metadata into the provenance.

3.4 Common

Every system involved in the supply chain must adhere to a set of strict, common security standards.

TABLE 5: SLSA4 COMMON REQUIREMENTS AND SUSE'S OBS

Requirement	Required at SLSA4	Achieved
Security	✓	✓
Access	✓	✓
Superusers	✓	✓

Learn how SUSE's software supply chain is kept secure by securing all systems involved:

Security

- **SLSA L4 requirement.** The source control and build systems must meet certain security standards to ensure they cannot be easily compromised.
- **SUSE's processes.** Red team scans for vulnerabilities, build team patches build environment. Machine identity is enforced, user isolation by policy.

Access

- **SLSA L4 requirement.** Make sure that physical and/or remote access is rare and tightly regulated. This includes logging and multi-party approval.
- **SUSE's processes.** Multi-party approval is done when the user is logged in via SSH auth key which is salted.

Superusers

- **SLSA L4 requirement.** Restrict the number of admins allowed to override the above-listed criteria to a bare minimum. Whenever such interaction is required, a second admin must give their approval.
- **SUSE's processes.** The number of superusers in OBS is very small.

4 Outlook

SUSE is up to a great start with its effort to attain SLSA L4 compliance, as SLSA requirements partly overlap with those of Common Criteria EAL4+. This means that several SLSA criteria were met by SUSE's supply chain processes right from the start. The core part of this is our certified and proven build and integration process which uses the Open Build Service technology. Over the past few months, SUSE has been working on improving and tightening processes and technologies to be able to claim full SLSA L4 compliance.

New to SUSE is providing the information of the build environment (*provenance*) to the outside world. So far, this kind of information has been kept internal, but now it will be delivered in the *in-toto* format, allowing external parties to verify the full build environment used for building all SUSE deliverables.

5 Legal notice

Copyright ©2006-2025 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”.

SUSE, the SUSE logo and YaST are registered trademarks of SUSE LLC in the United States and other countries. For SUSE trademarks, see <http://www.suse.com/company/legal/>. Linux is a registered trademark of Linus Torvalds. All other names or trademarks mentioned in this document may be trademarks or registered trademarks of their respective owners.

Documents published as part of the **SUSE Best Practices** series have been contributed voluntarily by SUSE employees and third parties. They are meant to serve as examples of how particular actions can be performed. They have been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. SUSE cannot verify that actions described in these documents do what is claimed or whether actions described have unintended consequences. SUSE LLC, its affiliates, the authors, and the translators may not be held liable for possible errors or the consequences thereof.

Below we draw your attention to the license under which the articles are published.

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. Secondly, 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 noncommercially, 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 <http://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.