OPENCHAIN

OpenChain ~~Conformance~~ Specification

Version 1.1 pc11 (DRAFT)

DRAFT: This is the near final draft of the 1.1 version of the OpenChain Specification. We have recently completed a round of feedback from the OpenChain community. It is now being circulated more broadly for public comments as the final step which will conclude on March 19th 2017. You can send your feedback to project mailing list at openchain@lists.linuxfoundation.org or Mark.Gisi@WindRiver.com if you prefer to provide comments anonymously. A summary of the changes made over the previous version (1.0) of the specification can be found on page 2. If you are new to OpenChain and would like obtain a better understanding of the goals and the context within which the Specification was developed, we recommend first reviewing the specification FAQs found at:

https://wiki.linuxfoundation.org/openchain/specification-questions-and-answers

We look forward to your feedback.

Changes over Specification 1.0 version:

- Page 3: Added the CC by 4.0 license notice – the license that governs the spec
- Page 5: DELETED Definition: Distributed Compliance Artifacts definition. Definite was moved to section 4.1 - the sole section where it is used. It is now referred to as Compliance Artifacts.
- Page 6 Cleaned up requirement 1.1 wording. Semantics remained the same.
- Page 6: Added new requirement 1.3 – to ensure a process exists for reviewing and identifying license obligations.
- Page 8: Cleaned up requirement 2.1 Verification Artifact wording. Some clarifications were made. See yellow highlights.
- Page 9: Cleaned up requirement 3.1 wording. Semantics remained the same.
- Page 9: Added Verification Artifact 3.1.2.
- Page 9: Cleaned up requirement 3.2 wording. Semantics remained the same.
- Page 10: Cleaned up requirement 4.1 wording. Semantics remained the same.
- Page 11: Cleaned up requirement 5.1 wording. Semantics remained the same.
- Page 11: Cleaned up requirement 5.2 wording. Removed contribution process considerations to simplify the requirement. Although discussed extensively, a major consensus was not reach on how to expand section 5 guidance. The discussion will be revisited during the next revision.
- Page 12: Added new requirement 6.2 – requires program re-validation after 18 months in order to retain spec conformance status.
- Page 13: Added Append I – Notes that translations are welcome and provides web page where to find additional information on how to contribute a new translation.

The previous version of the specification (1.0) can be found here:
https://wiki.linuxfoundation.org/_media/openchain/openchainspec-1.0.pdf

[THIS PAGE WILL BE REMOVE FROM THE FINAL DRAFT]

Contents

Introduction

The OpenChain Initiative began in 2013 when a group of software supply chain open source practitioners observed two emerging patterns: 1) significant process similarities existed among organizations with mature open source compliance programs; and 2) there still remained a large number of organizations exchanging software with less developed programs. The latter observation resulted in a lack of trust in the consistency and quality of the compliance artifacts accompanying the software being exchanged. As a consequence, at each tier of the supply chain, downstream organizations were frequently redoing the compliance work already performed by other upstream organizations.

A study group was formed to consider whether a standard program specification could be created that would: i) facilitate greater quality and consistency of open source compliance information being shared across the industry; and ii) decrease the high transaction costs associated with open source resulting from compliance rework. The study group evolved into a work group, and in April 2016, formally organized as a Linux Foundation collaborative project.
The Vision and Mission of the OpenChain Initiative are as follows:

- **Vision**: A software supply chain where free/open source software (FOSS) is delivered with ~~trusted~~ trustworthy and consistent compliance information.

- **Mission**: Establish requirements to achieve effective management of free/open source software (FOSS) for software supply chain participants, such that the requirements and associated collateral are developed collaboratively and openly by representatives from the software supply chain, open source community, and academia.

In accordance with the Vision and Mission, this specification defines a set of requirements that if met, would significantly increases the probability that an open source compliance program had achieved a sufficient level of quality, consistency and completeness; although a program that satisfies all the specification requirements does not guarantee full compliance. The requirements represent a base level (minimum) set of requirements a program must satisfy to be considered OpenChain Conforming. The specification focuses on the “what” and “why” qualities of a compliance program as opposed to the “how” and “when” considerations. This ensures a practical level of flexibility that enables different organizations to tailor their policies and processes to best fit their objectives.

Section 2 introduces definitions of key terms used throughout the specification. Section 3 presents the specification requirements where each one has a list of one or more Verification Artifacts. They represent the evidence that must exist in order for a given requirement to be considered satisfied. If all the requirements have been met for a given program, it would be considered OpenChain Conforming in accordance with version 1.1 of the specification. Verification Artifacts are not intended to be public, but could be provided under NDA or upon private request from the OpenChain organization to validate conformance.

Definitions

FOSS (Free and Open Source Software) - software subject to one or more licenses that meet the Open Source Definition published by the Open Source Initiative (OpenSource.org) or the Free Software Definition (published by the Free Software Foundation) or similar license.

FOSS Liaison - a designated person who is assigned to receive external FOSS inquires.

Identified Licenses - a set of FOSS licenses identified as a result of following an appropriate method of identifying such licenses.

OpenChain Conforming – a program that satisfies all the requirements of this specification.

Software Staff - any employee or contractor that defines, contributes to or has responsibility for preparing Supplied Software. Depending on the organization, that may include (but is not limited to) software developers, release engineers, quality engineers, product marketing and product management.

SPDX or Software Package Data Exchange – the format standard created by the SPDX Working Group for exchanging license and copyright information for a given software package. A description of the SPDX specification can be found at ww.spdx.org.

Supplied Software – software that an organization delivers to third parties (e.g., other organizations or individuals).

Verification Artifacts - evidence that must exist in order for a given requirement to be considered satisfied.

Requirements

G1: Know Your FOSS Responsibilities

1.1 A written FOSS policy exists that governs FOSS license compliance of the Supplied Software distribution. The policy must be internally communicated.

Verification Artifact(s):

- ☐ 1.1.1 A documented FOSS policy exists.
- ☐ 1.1.2 A documented procedure exists that makes all Software Staff aware of the existence of the FOSS policy (e.g., via training, internal wiki, or other practical communication method).

Rationale:
Ensure steps were taken to create, record and make Software Staff aware of the existence of a FOSS policy. Although no requirements are provided here on what should be included in the policy, other sections may impose requirements on the policy.

1.2 Mandatory FOSS training for all Software Staff exists such that:

- **The training, ~~as~~ at a minimum, covers the following topics:**
  - **The FOSS policy and where to find a copy;**
  - **Basics of ~~IP~~ Intellectual Property law pertaining to FOSS and FOSS licenses;**
  - **FOSS licensing concepts (including the concepts of permissive and copyleft licenses);**
  - **FOSS project licensing models;**
  - **Software Staff roles and responsibilities pertaining to FOSS compliance specifically and the FOSS policy in general; and**
  - **Process for identifying, recording and/or tracking of FOSS components contained in Supplied Software.**
- **Software Staff must have completed FOSS training within the last 24 months (to be considered current). A test may be used to allow Software Staff to satisfy the training requirement.**

Verification Artifact(s):

- ☐ 1.2.1 FOSS training materials covering the above topics exists (e.g., slide decks, online course, or other training materials).
- ☐ 1.2.2 Method of tracking the completion of the training for all Software Staff.
- ☐ 1.2.3 At least 85% of the Software Staff are current, as per the definition in above section.

Rationale:
Ensure the Software Staff have recently attended FOSS training and that a core set of relevant FOSS topics are covered. The intent is to ensure a core base level set of topics are covered but a typical training program would likely be more comprehensive than what is required here.

1.3 A process exists for reviewing the Identified Licenses to determine the obligations, restrictions and rights granted by each license.

Verification Artifact(s):

- ☐ 1.3.1 A documented procedure exists to review and document the obligations, restrictions and rights granted by each Identified License governing the Supplied Software.

Rationale:

To ensure a procedure exists for reviewing and identifying the license obligations for each Identified License for the various use cases.

G2: Assign Responsibility for Achieving Compliance

2.1 Identify FOSS Liaison Function ("FOSS Liaison").

- **Assign individual(s) responsible for receiving external FOSS inquiries;**
- **FOSS Liaison must make commercially reasonable efforts to respond to FOSS compliance inquiries as appropriate; and**
- **Publicly identify a means by which one can contact the FOSS Liaison.**

Verification Artifact(s):

- ☐ 2.1.1 FOSS Liaison function is publicly identified (e.g., via a published contact email address, or the Linux Foundation's Open Compliance Directory).
- ☐ 2.1.2 An internal documented procedure exists that assigns responsibility for receiving FOSS compliance inquiries.

Rationale:
Ensure there is a reasonable way for third parties to contact the organization with regard to FOSS compliance inquiries and that this responsibility has been effectively assigned.

2.2 Identify Internal FOSS Compliance Role(s).

- **Assign individual(s) responsible for managing internal FOSS compliance. The FOSS Compliance role and the FOSS Liaison ~~can~~ may be the same individual.**
- **FOSS compliance management activity is sufficiently resourced:**
  - **Time to perform the role has been allocated; and**
  - **Commercially reasonable budget has been allocated.**
- **Assign responsibilities to develop and maintain FOSS compliance policy and processes;**
- **Legal expertise pertaining to FOSS compliance is accessible to the FOSS Compliance role (e.g., could be internal or external); and**
- **Escalation path is available for resolution of FOSS compliance issues.**

Verification Artifact(s):

- ☐ 2.2.1 Name of persons, group or function in FOSS Compliance role(s) internally identified.
- ☐ 2.2.2 Identify source of legal expertise available to FOSS Compliance role(s) which could be internal or external.
- ☐ 2.2.3 A documented procedure exists that assigns internal responsibilities for FOSS compliance.
- ☐ 2.2.4 A documented procedure exists for handling review and remediation of non-compliant cases.

Rationale:
Ensure certain FOSS responsibilities have been effectively assigned.

G3: Review and Approve FOSS Content

3.1 A process exists for creating and managing a FOSS component bill of materials which includes each component (and its Identified Licenses) in a Supplied Software release.

Verification Artifact(s):

- ☐ 3.1.1 A documented procedure exists that describes the process of identifying, tracking and archiving information about the collection of FOSS components from which a Supplied Software release is comprised.
- ☐ 3.1.2 FOSS component records exist for each Supplied Software release which demonstrates the documented procedure was properly followed.

Rationale:
To ensure a procedure exists for creating and managing a FOSS component bill of materials used to construct the Supplied Software. A bill of materials is needed to support the systematic review of each component's license terms to understand the obligations and restrictions as it applies to the distribution of the Supplied Software.

3.2 The FOSS program must be capable of handling common FOSS license use cases encountered by Software Staff for Supplied Software, which may include the following use cases (note that the list is neither exhaustive, nor may all of the use cases apply):

- **distributed in binary form;**
- **distributed in source form;**
- **integrated with other FOSS such that it may trigger copyleft obligations;**
- **contains modified FOSS;**
- **contains FOSS or other software under an incompatible license interacting with other components within the Supplied Software; and/or**
- **contains FOSS with attribution requirements.**

Verification Artifact(s):

- ☐ 3.2.1 A procedure has been implemented that handles ~~is capable of handling~~ the common FOSS license use cases for the FOSS components of each Supplied Software release ~~encountered by Software Staff for Supplied Software~~.

Rationale:
To ensure the FOSS management program is sufficiently robust to handle an organization's common FOSS license use cases ~~as a result of that organization's business practices~~. That a procedure exists to support this activity and that the procedure is followed.

G4: Deliver FOSS Content Documentation and Artifacts

4.1 Prepare the set of artifacts which represent the output of the of the FOSS review program for each Supplied Software release. This set is referred to as the Compliance Artifacts which may include (but are not limited to) one or more of the following: source code, attribution notices, copyright notices, copy of licenses, modification notifications, written offers, SPDX documents and so forth.

Verification Artifact(s):

- ☐ 4.1.1 A documented procedure exists describing a process that ensures the Compliance Artifacts are prepared and distributed with Supplied Software release as required by the Identified Licenses.

- ☐ 4.1.2 Copies of the Compliance Artifacts of the Supplied Software release are archived and easily retrievable, and the archive is planned to exist for at least as long as the Supplied Software is offered or as required by the Identified Licenses (whichever is longer).

Rationale:

Ensure the complete collection of Compliance Artifacts accompany the Supplied Software as required by the Identified Licenses that govern the Supplied Software along with other reports created as part of the FOSS review process.

G5: Understand FOSS Community Engagement

5.1 A written policy exists that governs contributions to FOSS projects by the organization. The policy must be internally communicated.

Verification Artifact(s):

- ☐ 5.1.1 A documented FOSS contribution policy exists;
- ☐ 5.1.2 A documented procedure exists that makes all Software Staff aware of the existence of the FOSS contribution policy (e.g., via training, internal wiki, or other practical communication method).

Rationale:
Ensure an organization has given reasonable consideration to developing a policy with respect to publicly contributing to FOSS. The FOSS contribution policy can be made a part of the overall FOSS policy of an organization or be its own separate policy. In the situation where contributions are not permitted at all, a policy should exist making that position clear.

5.2 **If an organization permits contributions to FOSS projects then a process must exist that implements the FOSS contribution policy outlined in Section 5.1.**

Verification Artifact(s):

- ☐ 5.2.1 Provided the FOSS contribution policy permits contributions, a documented procedure exists that describes the FOSS contribution process.

Rationale:
Ensure an organization has a documented procedure for how the organization publicly contributes FOSS. A policy may exist such that contributions are not permitted at all. In that situation it is understood that no procedure may exist and this requirement would nevertheless be met.

G6: Certify Adherence to OpenChain Requirements

6.1 In order for an organization to be OpenChain certified, it must affirm that it has a FOSS program that meets the criteria described in this OpenChain ~~Conformance~~ Specification version 1.1.

Verification Artifact(s):

- ☐ 6.1.1 The organization affirms that a program exists that meets all the requirements of this OpenChain ~~Conformance~~ Specification version 1.1.

Rationale:
To ensure that if an organization declares that it has a program that is OpenChain Conforming, that such program has met all the requirements of this specification. The mere meeting of a subset of these requirements would not be considered sufficient to warrant a program be OpenChain certified.

6.2 Conformance with this version of the specification will last 18 months from the date conformance validation was achieved. Conformance validation requirements can be found on the OpenChain project's website.

Verification Artifact(s):

- ☐ 6.2.1 The organization affirms that a FOSS compliance program exists that meets all the requirements of this OpenChain ~~Conformance~~ Specification version 1.1 within the past 18 months of achieving conformance.

Rationale:
It is important for the organization to remains current with the specification if they want to assert conformance overtime. This requirement ensures that the program's supporting processes and controls do not erode if they want to continue to assert conformance with the specification overtime.

Appendix I: Language Translations

To facilitate global adoption we welcome efforts to translate the specification into multiple languages. Because OpenChain functions as an open source project translations are driven by those willing to contribute their time and expertise to perform translations under the terms of the CC-BY 4.0 license and the project's translation policy. The details of the policy and available translations can be found on the OpenChain project specification webpage.