**OpenChain Specification**

Version 2.0

* Translation Draft Englisch / Deutsch –

Inhaltsverzeichnis

Copyright © 2016-2019 Linux Foundation. Dieses Dokument ist unter der Creative Commons Attribution 4.0 International (CC-BY 4.0) Lizenz lizenziert. Eine Kopie des Lizenztexts finden Sie unter https://creativecommons.org/licenses/by/4.0/.

Einleitung

|  |  |  |
| --- | --- | --- |
| 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 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 Materials. 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 1.2 of the specification. Verification Materials are not intended to be public, but could be provided under NDA or upon private request from the OpenChain organization to validate conformance.  Additional clarification on how to interpret the specification can be obtained by reviewing the Specification Frequently Asked Questions (FAQs) located at: <https://www.openchainproiect.org/specification-faq> |  | Die OpenChain Initiative wurde im Jahr 2013 ins Leben gerufen, nachdem eine Gruppe von Open-Source-Anwendern zwei wiederkehrende Muster in Open-Source-Software-Lieferketten beobachtet hatte: 1) die für den Umgang mit Open Source Software definierten Prozesse wiesen bei Organisationen mit ausgereiften Open-Source-Compliance-Programmen erhebliche Ähnlichkeiten auf; und 2) es gab noch eine große Anzahl von Organisationen, die Software im Rahmen weniger weit entwickelter Programme austauschte. Diese nachrangig genannte Erkenntnis veranlasste dazu, der Konsistenz und Qualität der Compliance-Artefakte, die mit zugelieferter Software zur Verfügung gestellt werden, nur ein geringes Vertrauen zu schenken. Infolgedessen führen auf jeder Stufe der Lieferkette Organisationen Compliance-Arbeit erneut durch, auch wenn sie bereits von Zulieferern ausgeführt wurde.  Es wurde eine Forschungs- und Arbeitsgemeinschaft gebildet, um zu prüfen, ob gemeinsame Standard-Spezifikationen für Compliance-Programme identifiziert und definiert werden könnten, die: i) zu einer verbesserten Qualität und Konsistenz der Open-Source-Compliance-Informationen führen, die in der gesamten Industrie geteilt werden; und ii) die hohen Transaktionskosten im Zusammenhang mit Open Source Software reduzieren, die sich aus der Wiederholung von Compliance-Arbeit ergeben. Die Forschungsgemeinschaft entwickelte sich zu einem Arbeitskreis, der im April 2016 offiziell als Linux Foundation-Kooperationsprojekt eingerichtet wurde.  Die OpenChain Initiative basiert auf folgender Vision und Mission:   * Vision: Eine Software Supply Chain, in der Freie und Open Source Software (FOSS) mit vertrauenswürdigen und konsistenten Compliance-Informationen zugeliefert wird. * Mission: Etablieren von Anforderungen die einen effektiven Umgang mit Freier und Open-Source-Software (FOSS) durch Mitglieder der Software Supply Chain ermöglichen, indem die Anforderungen und die entsprechenden Begleitmaterialien gemeinsam und offen von Vertretern der Software Supply Chain, Open Source Community und Hochschulen entwickelt werden.   In Übereinstimmung mit der Vision und der Mission definiert diese Spezifikation eine Reihe von Anforderungen, die, wenn sie erfüllt werden, die Wahrscheinlichkeit erhöhen, dass ein Open-Source-Compliance-Programm ein ausreichendes Maß an Qualität, Konsistenz und Vollständigkeit erreicht hat; auch wenn ein Programm, das alle Anforderungen der Spezifikation erfüllt, keine vollständige Compliance garantiert. Die definierten Anforderungen stellen einen Katalog von grundlegenden (Minimal-) Anforderungen auf, die ein Programm erfüllen muss, um als “OpenChain Conforming” anerkannt zu werden. Die Spezifikation konzentriert sich auf die "Was" und "Warum" -Beschaffenheit eines Compliance-Programms, statt "Wie" und "Wenn" -Überlegungen anzustellen. So wird ein praktisches Maß an Flexibilität sichergestellt, welches es Organisationen aller Art ermöglicht, ihre Richtlinien und Prozesse optimal an ihre Ziele anzupassen.  Im folgenden Abschnitt 2 werden zunächst Definitionen von Schlüsselbegriffen eingeführt, die in der gesamten Spezifikation verwendet werden. Abschnitt 3 enthält die Anforderungen der Spezifikation. Jeder Anforderung ist eine Liste von Verifikationsmaterial zugeordnet. Dieses stellt den oder die Nachweis(e) dar, die vorhanden sein müssen, damit eine bestimmte Anforderung als erfüllt angesehen werden kann. Wenn ein bestimmtes Programm alle Anforderungen erfüllt, gilt es als OpenChain Conforming gemäß Version 1.2 der Spezifikation. Es ist nicht beabsichtigt, dass das jeweilige Verifikationsmaterial öffentlich zugänglich sein muss. Es kann vielmehr im Rahmen einer Verschwiegenheitsvereinbarung offengelegt werden oder auf individuelle Anfrage von der OpenChain Initiative zur Bestätigung der Konformität bereitgestellt werden.  Zusätzliche, klarstellende Informationen zur Auslegung der Spezifikation können dem Spezifikations-FAQ unter <https://www.openchainproject.org/specification-faq> entnommen werden. |

Definitionen

|  |  |  |
| --- | --- | --- |
| **Compliance Artifacts** - a collection of artifacts which represent the output of the FOSS management program for a Supplied Software release. The collection 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, FOSS component bill of materials, SPDX documents and so forth.  **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 licenses that govern the Supplied Software.  **OpenChain Conforming** Program - 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 [www.spdx.org](http://www.spdx.org).  **Supplied Software** - software that an organization delivers to third parties (e.g., other organizations or individuals).  **Verification Materials** - evidence that must exist in order for a given requirement to be considered satisfied. | **Compliance Artifacts -** a collection of artifacts which represent the output of the Open Source management program for a Supplied Software release. The collection 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, Open Source component bill of materials, SPDX documents and so forth.  **Open Source Software (Open Source)** - 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.  **Open Source Liaison** - a designated person who is assigned to receive external Open Source inquires.  **Identified Licenses** - a set of Open Source Software licenses identified as a result of following an appropriate method of identifying licenses that govern the Supplied Software.  **OpenChain Conforming Program** (Program) - 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 www.spdx.org.  **Supplied Software** - software that an organization delivers to third parties (e.g., other organizations or individuals).  **Verification Materials** - evidence that must exist in order for a given requirement to be considered satisfied. | **Compliance-Artefakte –** Eine Zusammenstellung von Artefakten, die für eine Version Zugelieferter Software das Arbeitsergebnis des FOSS-Management-Programms darstellen.  Die Zusammenstellung kann eines oder mehrere der folgenden Elemente enthalten (ist aber nicht auf diese beschränkt): Quellcode, Benennung des Autors, Urheberrechtshinweise, Kopien der Lizenzbedingungen, Bearbeitungshinweise, schriftliche Angebote, eine FOSS-Komponenten-Stückliste („FOSS Bill of Materials“ bzw. „FOSS-BoM"), SPDX-Dokumente, etc.  **FOSS (Freie und Open Source Software)** - Software, die einer oder mehreren Lizenzen unterliegt, die den Anforderungen der Open Source Definition der Open Source Initiative (OpenSource.org) oder der Free Software Definition der Free Software Foundation entsprechen.  **FOSS Liaison** - eine konkrete Person, die für den Erhalt externer FOSS Anfragen bestimmt wird.  **Identifizierte Lizenzen** - eine Reihe von FOSS-Lizenzen, die aufgrund einer geeigneten Methode zur Identifizierung derjenigen FOSS-Lizenzen, die einer Zugelieferten Software zugrunde liegen, identifiziert wurden.  **OpenChain Conforming Program** - ein Programm, das alle Anforderungen dieser Spezifikation erfüllt.  **Software-Mitarbeiter** - jeder Mitarbeiter oder Auftragnehmer, der die Vorgaben für zugelieferte Software festlegt, zu ihr beiträgt oder für ihre Vorbereitung verantwortlich ist. Abhängig von der jeweiligen Organisation sind dies insbesondere Software-Entwickler, Release-Ingenieure, Qualitätsprüfer, Produkt-Marketing und Produkt-Management.  **SPDX oder Software Package Data Exchange** - der von der SPDX-Arbeitsgruppe erstellte Format-Standard für den Austausch von Lizenz- und Urheberrechtsinformationen für ein bestimmtes Softwarepaket. Eine Beschreibung der SPDX-Spezifikation finden Sie unter [www.spdx.org](http://www.spdx.org).  **Zugelieferte Software** - Software, die eine Organisation an Dritte weitergibt (z. B. andere Organisationen oder Einzelpersonen).  **Verifikationsmaterial** - Nachweise, die vorhanden sein müssen, damit eine bestimmte Anforderung als erfüllt angesehen werden kann. |

Requirements

|  |  |  |
| --- | --- | --- |
| Goal 1: Know Your FOSS Responsibilities | **Part 1: Program Foundation** | Ziel 1: Erkennen und verstehen Sie Ihre Verpflichtungen bei der Nutzung von FOSS |
| 1.1 A written FOSS policy exists that governs FOSS license compliance of the Supplied Software distribution. The policy must be internally communicated.  Verification Material(s):  1.1.1 A documented FOSS policy.  1.1.2 A documented procedure that makes Software Staff aware of the existence of the FOSS policy (e.g., via training, internal wiki, or other practical communication method).  Rationale:  To ensure steps are 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, at a minimum, covers the following topics:   * The FOSS policy and where to find a copy; * Basics of 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 (“Currently Trained“). A test may be used to allow Software Staff to satisfy the training requirement.  Verification Material(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 Documented method for tracking the completion of the training for the Software Staff.  1.2.3 At least 85% of the Software Staff are Currently Trained, as per the definition above. The 85% may not necessarily refer to the entire organization, but to the totality Software Staff governed by the OpenChain Conforming program.  Rationale:  To ensure the Software Staff have recently attended FOSS training and that a core set of relevant FOSS topics were covered in the training. 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 Material(s):  1.3.1 A documented procedure to review and document the obligations, restrictions and rights granted by each Identified License governing the Supplied Software.  Rationale:  To ensure a process exists for reviewing and identifying the license obligations for each Identified License for the various use cases. | **1.1 Policy**  **A written Open Source policy exists that governs Open Source license compliance of the Supplied Software distribution. The policy must be internally communicated.**  **Verification Material(s)**:  1.1.1 A documented Open Source policy.  1.1.2 A documented procedure that makes Software Staff aware of the existence of the Open Source policy (e.g., via training, internal wiki, or other practical communication method).  **Rationale**:  To ensure steps are taken to create, record and make Software Staff aware of the existence of a Open Source 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 Competence**  **The organization shall:**   * **Identify the roles and the corresponding responsibilities of those roles that affects the performance and effectiveness of the Program;** * **Determine the necessary competence of person(s) fulfilling each role** * **Ensure that these persons are competent on the basis of appropriate education, training, and/or experience;** * **Where applicable, take actions to acquire the necessary competence** * **Retain appropriate documented information as evidence of competence**   **Verification Material(s):**  1.2.1 A documented list of roles with corresponding responsibilities for the different participates in the Open Source compliance program;  1.2.2 A documented that identifies the competencies for each role  1.2.3 Documented evidence of assessed competence for each program participant  **Rationale**:  To ensure that the program participants have obtain a sufficient level of competence for their respected roles and responsibilities.  1.3 Awareness  The organization shall ensure that persons doing work are aware of:  a) The Open Source policy;  b) Relevant Open Source objectives;  c) Their contribution to the effectiveness of the Open Source compliance program;  d) The implications of not conforming to the Open source compliance program requirements.  Verification Material(s):  1.3.1 Documented evidence of assessed awareness for each program participant including implications of non-conformance.  Rationale:  To ensure program participants have obtain a sufficient level of awareness for their respected roles and responsibilities.  1.4 Program Scope  Different compliance programs may be governed by different levels of scope. For example, a program could govern a single product line, an entire department or an entire organization. The scope designation needs to be declared for each program seeking conformance.  Verification Material(s):  1.4.1 A written statement that clearly defines the scope of the program.  Rationale:  Provide the flexibility to construct a compliance program that best fits the scope of an organization’s needs. Some organizations could choose to maintain a compliance program for a specific product line while others could choose the program scope to govern software releases of the entire organization. Large organizations may prefer the former example while smaller organizations may prefer the latter.  1.5 License Obligations  A process exists for reviewing the Identified Licenses to determine the obligations, restrictions and rights granted by each license.  Verification Material(s):  1.5.1 A documented procedure to review and document the obligations, restrictions and rights granted by each Identified License.  Rationale:  To ensure a process exists for reviewing and identifying the license obligations for each Identified License for the various use cases. | 1.1 Es existiert eine schriftliche FOSS-Richtlinie, in der die Anforderungen an die FOSS-Lizenz-Compliance innerhalb der Supplied Software Distribution geregelt ist. Die Richtlinie muss innerhalb des Unternehmens kommuniziert werden.  Verifikationsmaterial:  1.1.1 Eine schriftlich dokumentierte FOSS-Richtlinie.  1.1.2 Ein dokumentiertes Verfahren, welches die Software-Mitarbeiter auf die Existenz der FOSS-Richtlinie aufmerksam macht (z. B. über Training, internes Wiki oder eine andere im Unternehmen relevante Kommunikationsmethode).  Begründung:  Es soll sichergestellt werden, dass die notwendigen Schritte unternommen wurden, um Software-Mitarbeiter auf die Existenz der FOSS-Richtlinie hinzuweisen. Obwohl an dieser Stelle keine inhaltlichen Vorgaben an die FOSS-Richtlinie gestellt werden, können solche inhaltlichen Vorgaben an anderer Stelle dieser Spezifikation genannt werden.  1.2 Zwingend vorgeschriebene FOSS-Schulungen für alle Software-Mitarbeiter:  Die Schulung umfasst mindestens folgende Themen:   * Die FOSS-Richtlinie der Organisation und wo man eine Kopie findet; * Grundlagen zu gesetzlichen Vorgaben im Bereich Intellectual Property in Bezug auf FOSS und FOSS-Lizenzen; * FOSS-Lizenzkonzepte (einschließlich der Konzepte von permissiven und copyleft-Lizenzen); * Lizenzmodelle für FOSS Projekte; * Rollenverteilung und Verantwortlichkeiten der Software-Mitarbeiter im Zusammenhang mit der FOSS-Richtlinie im Allgemeinen und FOSS-Compliance im Besonderen; und * Verfahren zur Identifizierung, Dokumentation und / oder Nachverfolgung von FOSS-Komponenten, die in der Zugelieferten Software enthalten sind.   Alle Software-Mitarbeiter müssen in den vergangenen 24 Monaten eine FOSS-Schulung absolviert haben, um als „Aktuell Geschult“ zu gelten. Um die Anforderungen an die FOSS-Schulung zu erfüllen, kann ein Test zu absolvieren sein.  Verifikationsmaterial:  1.2.1 FOSS-Schulungsunterlagen, die die oben genannten Themen abdecken (z. B. Präsentationen, Online-Kurse oder andere Schulungsunterlagen).  1.2.2 Ein dokumentiertes Verfahren zur Nachverfolgung der Schulungsteilnahme für die Software-Mitarbeiter.  1.2.3 Mindestens 85% der Software-Mitarbeiter gelten nach der obenstehenden Definition als „Aktuell Geschult“. Die 85% müssen sich hierbei nicht auf die gesamte Organisation, jedoch auf die Gesamtheit der Software-Mitarbeiter beziehen, auf die sich das OpenChain Conforming Program erstreckt.  Begründung:  Es soll sichergestellt werden, dass die Software-Mitarbeiter zeitnah an einer FOSS-Schulung teilgenommen haben und dass die Schwerpunkte der relevanten FOSS-Themen durch die FOSS-Schulung abgedeckt wurden. Ziel ist es, sicherzustellen, dass alle Schwerpunkte relevanter FOSS-Themen abgedeckt sind, wobei die einzelnen Schulungsprogramme wahrscheinlich sehr viel umfassender ausfallen als hier gefordert.  1.3 Es besteht ein Verfahren zur Überprüfung der Identifizierten Lizenzen um die jeweiligen Rechte, Einschränkungen und Verpflichtungen zu erkennen.  Verifikationsmaterial:  1.3.1 Ein dokumentiertes Verfahren zur Überprüfung und Dokumentation der Rechte, Beschränkungen und Verpflichtungen, die durch die jeweiligen Identifizierten Lizenzen an der bzw. in Bezug auf die Zugelieferte Software bestehen.  Begründung:  Es soll sichergestellt werden, dass ein Prozess besteht, in dem die Lizenzpflichten für die verschiedenen Anwendungsfälle geprüft und identifiziert werden. |

|  |  |  |
| --- | --- | --- |
| Goal 2: Assign Responsibility for Achieving Compliance | Goal 2: Assign Roles and Responsibilities | Ziel 2: Weisen Sie die Verantwortung für die Erfüllung der License Compliance zu |
| 2.1 Identify External 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 Material(s):  2.1.1 Publicly visible identification of FOSS Liaison (e.g., via a published contact email address, or the Linux Foundation's Open Compliance Directory).  2.1.2 An internal documented procedure that assigns responsibility for receiving FOSS compliance inquiries.  Rationale:  To 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 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  A process exists for the resolution of FOSS compliance issues  Verification Material(s):  2.2.1 Name of persons, group or function in FOSS Compliance role(s) internally identified.  2.2.2 Identification of legal expertise available to FOSS Compliance role(s) which could be internal or external.  2.2.3 A documented procedure that assigns internal responsibilities for FOSS compliance.  2.2.4 A documented procedure for handling review and remediation of non-compliant cases.  Rationale:  To ensure certain FOSS responsibilities have been effectively assigned. | 2.1 Identify External Open Source Liaison Function ("Open Source Liaison").  Assign individual(s) responsible for receiving external Open Source inquiries;  Open Source Liaison must make commercially reasonable efforts to respond to Open Source compliance inquiries as appropriate; and  Publicly identify a means by which one can contact the Open Source Liaison.  Verification Material(s): 2.1.1 Publicly visible identification of Open Source Liaison (e.g., via a published contact email address, or the Linux Foundation's Open Compliance Directory).  2.1.2 An internal documented procedure that assigns responsibility for receiving Open Source compliance inquiries.  Rationale:  To ensure there is a reasonable way for third parties to contact the organization with regard to Open Source compliance inquiries and that this responsibility has been effectively assigned.  2.2 Identify Internal Open Source Compliance Role(s).  Assign individual(s) responsible for managing internal Open Source compliance. The Open Source Compliance role and the Open Source Liaison may be the same individual.  Open Source 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 Open Source compliance policy and processes;  Legal expertise pertaining to Open Source compliance is accessible to the Open Source Compliance role (e.g., could be internal or external); and  A process exists for the resolution of Open Source compliance issues.  Verification Material(s):  2.2.1 Name of persons, group or function in Open Source Compliance role(s) internally identified.  2.2.2 Identification of legal expertise available to Open Source Compliance role(s) which could be internal or external.  2.2.3 A documented procedure that assigns internal responsibilities for Open Source compliance.  2.2.4 A documented procedure for handling the review and remediation of non-compliant cases.  Rationale:  To ensure internal Open Source responsibilities have been effectively assigned. | 2.1 Identifikation der Funktion eines FOSS-Ansprechpartners ("FOSS Liaison"), der auch außerhalb der Organisation bekannt gegeben wird.  Ernennung einer oder mehrerer Einzelpersonen, die für die Entgegennahme und Bearbeitung externer FOSS-Anfragen verantwortlich sind;  Die FOSS Liaison muss kommerziell vernünftige Anstrengungen unternehmen, um auf FOSS-Compliance-Anfragen zu reagieren; und öffentlich den Kommunikationskanal bekannt machen, über den er/sie kontaktiert werden kann.  Verifikationsmaterial:  2.1.1 Eine öffentlich sichtbare Bekanntgabe der FOSS-Liasion (z. B. durch Veröffentlichen einer Kontakt-E-Mail-Adresse oder Aufnahme in das Open Compliance-Verzeichnis der Linux Foundation).  2.1.2 Ein intern dokumentiertes Verfahren, das die Verantwortung für den Empfang und die Bearbeitung von FOSS-Compliance-Anfragen zuweist.  Begründung:  Es soll sichergestellt werden, dass es für Dritte eine angemessene Möglichkeit gibt, sich mit der Organisation in Bezug auf FOSS-Compliance-Anfragen in Verbindung zu setzen und dass die Verantwortung zur Bearbeitung entsprechender Anfragen effektiv zugeordnet wird.  2.2 Identifikation der internen FOSS-Compliance Rolle(n).  Zuweisen der Verantwortung zur Verwaltung und zum Management der internen FOSS-Compliance an individuelle Mitarbeiter. Die FOSS-Compliance-Verantwortlichen und die FOSS-Liaison können dieselben Personen sein.  FOSS-Compliance-Management-Aktivitäten verfügen über ausreichende Ressourcen:   * Weisen Sie angemessene Zeit für die Ausführung der Rolle zu; und * Weisen Sie ein angemessenes Budget zu.   Zuweisen der Verantwortlichkeiten zur Entwicklung und Pflege von FOSS-Compliance-Richtlinie und -Prozessen;  Gewährleisten, dass juristische Expertise in Bezug auf die FOSS-Compliance vorhanden und für die FOSS-Compliance-Verantwortlichen (intern oder extern) zugänglich ist; und  Sicherstellen, dass ein Prozess für die Lösung von FOSS-Compliance-Problemen zur Verfügung steht.  Verifikationsmaterial:  2.2.1 Name der Personen, Gruppe oder Funktionen des/der FOSS-Compliance-Verantwortliche (n) sind intern identifiziert.  2.2.2 Benennung der juristischen Expertise, die dem/den FOSS-Compliance-Verantwortlichen intern oder extern zur Verfügung steht.  2.2.3 Ein dokumentiertes Verfahren, das interne Verantwortlichkeiten für die FOSS-Compliance zuweist.  2.2.4 Ein dokumentiertes Verfahren zur Prüfung und Behebung von Fällen der Nichterfüllung von FOSS-Compliance-Anforderungen.  Begründung:  Es soll sichergestellt sein, dass einzelnen Mitarbeitern konkrete FOSS-Verantwortlichkeiten verbindlich zugewiesen wurden. |

|  |  |  |
| --- | --- | --- |
| Goal 3: Review and Approve FOSS Content | Goal 3: Review and Approve Open Source Content | Ziel 3: Überprüfen und genehmigen Sie 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) from which the Supplied Software is comprised.  Verification Material(s):  3.1.1 A documented procedure for 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 for each Supplied Software release which demonstrates the documented procedure was properly followed.  Rationale:  To ensure a process 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 Material(s):  3.2.1 A documented procedure for handling the common FOSS license use cases for the FOSS components of the Supplied Software.  Rationale:  To ensure the program is sufficiently robust to handle an organization’s common FOSS license use cases. That a procedure exists to support this activity and that the procedure is followed. | 3.1 A process exists for creating and managing a Open Source component bill of materials which includes each component (and its Identified Licenses) from which the Supplied Software is comprised.  Verification Material(s):  3.1.1 A documented procedure for identifying, tracking and archiving information about the collection of Open Source components from which a Supplied Software release is comprised.  3.1.2 Open Source component records for each Supplied Software release which demonstrates the documented procedure was properly followed.  Rationale:  To ensure a process exists for creating and managing a Open Source 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 Open Source management program must be capable of handling common Open Source 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 Open Source such that it may trigger copyleft obligations; * contains modified Open Source; * contains Open Source or other software under an incompatible license interacting with other components within the Supplied Software; and/or * contains Open Source with attribution requirements.   Verification Material(s):  3.2.1 A documented procedure for handling the common Open Source license use cases for the Open Source components of the Supplied Software.  Rationale:  To ensure the program is sufficiently robust to handle an organization’s common Open Source license use cases. That a procedure exists to support this activity and that the procedure is followed. | 3.1 Es existiert ein Prozess zum Erstellen und Verwalten einer Bill of Materials der FOSS-Komponenten, die jede Komponente (und ihre Identifizierten Lizenzen) enthält, aus der sich sie Zugelieferte Software zusammensetzt.  Verifikationsmaterial:  3.1.1 Ein dokumentiertes Verfahren zur Identifizierung, Nachverfolgung und Archivierung von Informationen über die Zusammensetzung von FOSS-Komponenten, aus denen eine Version Zugelieferter Software besteht.  3.1.2 Eine Aufzeichnung der FOSS-Komponenten für jede Version Zugelieferter Software, welche nachweist, dass die dokumentierte Prozedur ordnungsgemäß befolgt wurde.  Begründung:  Es soll sichergestellt werden, dass ein Prozess zum Erstellen und Verwalten einer Bill of Materials der FOSS-Komponenten existiert, anhand dessen die Zugelieferte Software erstellt wird. Die Bill of Materials ist erforderlich, um systematisch die Lizenzbedingungen jeder Komponente mit dem Ziel zu überprüfen, die Lizenzpflichten und -bedingungen mit Blick auf die Verbreitung der Zugelieferten Software zu ermitteln.  3.2 Das FOSS-Programm muss es ermöglichen, die üblichen Anwendungsfälle von FOSS-Lizenzen in Zugelieferter Software abzudecken. Zu den üblichen Fällen zählen dabei insbesondere (beachten Sie allerdings, dass die Liste weder erschöpfend ist, noch alle Anwendungsfälle auf Sie Anwendung finden müssen):   * Verbreitung in Binärform; * Verbreitung in Sourcecodeform; * Integration mit anderer FOSS, so dass die Voraussetzungen des Copyleft vorliegen können; * Enthält bearbeitete FOSS; * Enthält FOSS oder andere Software unter einer inkompatiblen Lizenz, die mit anderen Komponenten innerhalb der Zugelieferten Software interagiert; und / oder * Enthält FOSS mit Verpflichtungen hinsichtlich einer Nennung der Urheberschaft.   Verifikationsmaterial:  3.2.1 Ein dokumentiertes Verfahren, welches es ermöglicht, die üblichen Anwendungsfälle von FOSS-Lizenzen für die FOSS-Komponenten von Zugelieferter Software abzudecken.  Begründung:  Es soll sichergestellt werden, dass das Programm ausreichend robust ist, um die üblichen Anwendungsfälle von FOSS-Lizenzen einer Organisation zu behandeln. Es muss gewährleistet sein, dass ein Verfahren zur Unterstützung dieser Tätigkeit besteht und dass die vorgesehene Prozedur befolgt wird. |

|  |  |  |
| --- | --- | --- |
| Goal 4: Deliver FOSS Content Documentation and Artifacts | Goal 4: Deliver Compliance Artifacts | Ziel 4: Stellen Sie FOSS-Inhaltsdokumentation und Artefakte bereit |
| 4.1 A process exists for creating the set of Compliance Artifacts for each Supplied Software release.  Verification Material(s):  4.1.1 A documented procedure 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:  To 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. | 4.1 A process exists for creating the set of Compliance Artifacts for each Supplied Software release.  Verification Material(s):  4.1.1 A documented procedure that ensures the Compliance Artifacts are prepared and distributed with the 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:  To ensure the complete collection of Compliance Artifacts accompany the Supplied Software as required by the Identified Licenses along with other reports created as part of the Open Source review process. | 4.1 Es existiert ein Prozess, um die Compliance-Artefakte für jede Version einer Zugelieferten Software zu erstellen.  Verifikationsmaterial:  4.1.1 Ein dokumentiertes Verfahren, welches sicherstellt, dass die Compliance-Artefakte mit jeder Version Zugelieferter Software entsprechend den Anforderungen der Identifizierten Lizenzen zusammengestellt und verteilt werden.  4.1.2 Kopien der Compliance-Artefakte der Version Zugelieferter Software werden archiviert und sind einfach wiederauffindbar, und es ist geplant, dass das Archiv mindestens so lange besteht, wie die Zugelieferte Software angeboten wird oder wie es die Identifizierten Lizenzen verlangen (je nachdem, welcher Zeitraum länger ist).  Begründung:  Es soll sichergestellt werden, dass die vollständigen Compliance-Artefakte entsprechend den Anforderungen der Identifizierten Lizenzen, sowie sonstige Berichte, die während der FOSS-Überprüfung erstellt wurden, mit jeder Version der Zugelieferten Software ausgeliefert werden. |

|  |  |  |
| --- | --- | --- |
| Goal 5: Understand FOSS Community Engagement | Goal 5: Understand Open Source Community Engagement | Ziel 5: Verstehen Sie 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 Material(s):  5.1.1 A documented FOSS contribution policy;  5.1.2 A documented procedure 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:  To 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 Material(s):  5.2.1 Provided the FOSS contribution policy permits contributions, a documented procedure exists that governs FOSS contributions.  Rationale:  To ensure an organization has a documented process 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. | 5.1 A written policy exists that governs contributions to Open Source projects by the organization. The policy must be internally communicated.  Verification Material(s):  5.1.1 A documented Open Source contribution policy;  5.1.2 A documented procedure that makes all Software Staff aware of the existence of the Open Source contribution policy (e.g., via training, internal wiki, or other practical communication method).  Rationale:  To ensure an organization has given reasonable consideration to developing a policy with respect to publicly contributing to Open Source. The Open Source contribution policy can be made a part of the overall Open Source policy of an organization or be its own separate policy. In the situation where contributions are limited or not permitted at all, a policy should exist making that position clear.  5.2 If an organization permits contributions to Open Source projects then a process exists that implements the Open Source contribution policy outlined in Section 5.1.  Verification Material(s):  5.2.1 Provided the Open Source contribution policy permits contributions, a documented procedure that governs Open Source contributions.  Rationale:  To ensure an organization has a documented process for how the organization publicly contributes Open Source. 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. | 5.1 Es gibt eine schriftliche Richtlinie, die die Beiträge zu FOSS-Projekten durch die Organisation regelt. Die Richtlinie muss intern kommuniziert werden.  Verifikationsmaterial:  5.1.1 Eine dokumentierte Richtlinie für Beiträge zu FOSS;  5.1.2 Ein dokumentiertes Verfahren, welches alle Software-Mitarbeiter auf die Existenz der Richtlinie für Beiträge zu FOSS aufmerksam macht (z. B. mittels Training, ein internes Wiki oder andere praktische Kommunikationsmethode).  Begründung:  Es soll sichergestellt werden, dass die Organisation der Entwicklung einer Richtlinie für öffentliche Beiträge zu FOSS eine ausreichende Beachtung geschenkt hat. Die Richtlinie für Beiträge zu FOSS kann Teil einer übergreifenden FOSS-Richtlinie oder eine eigene separate Richtlinie sein. In dem Fall, dass Beiträge zu FOSS überhaupt nicht erlaubt sind, sollte es eine Richtlinie geben, die diese Haltung klarstellt.  5.2 Wenn eine Organisation Beiträge zu FOSS-Projekten zulässt, muss ein Prozess existieren, der die in Abschnitt 5.1 skizzierte Richtlinie für Beiträge zu FOSS umsetzt.  Verifikationsmaterial:  5.2.1 Wenn die Richtlinie Beiträge zu FOSS zulässt, muss ein dokumentiertes Verfahren existieren, anhand dessen Beiträge zu FOSS erfolgen.  Begründung:  Es soll sichergestellt werden, dass eine Organisation einen dokumentierten Prozess hat, wie sie öffentlich zu FOSS beiträgt. Es kann eine Richtlinie dergestalt bestehen, dass Beiträge gar nicht gestattet sind. Aus dieser Situation folgt zwingend, dass kein Verfahren existieren kann und, dass diese Anforderung auch ohne Verfahren erfüllt werden würde. |

|  |  |  |
| --- | --- | --- |
| Goal 6: Certify Adherence to OpenChain Requirements | Goal 6: Verify Adherence to OpenChain Requirements | Ziel 6: Zertifizieren der OpenChain-Anforderungen |
| 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 Specification version 1.2.  Verification Material(s):  6.1.1 An affirmation of the existence of a FOSS management program that meets all the requirements of this OpenChain Specification version 1.2.  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.  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 Material(s):  6.2.1 The organization affirms that a FOSS compliance program exists that meets all the requirements of this OpenChain Specification version 1.2 within the past 18 months of achieving conformance validation.  Rationale:  It is important for the organization to remain current with the specification if that organization wants to assert conformance over time. This requirement ensures that the program’s supporting processes and controls do not erode if the conforming organization continues to assert conformance over time. | 6.1 In order for an organization to have an OpenChain Conforming Program, it must affirm that the program that meets the criteria described in this OpenChain Specification version 1.3.  Verification Material(s):  6.1.1 An affirmation of the existence of a program that meets all the requirements of this OpenChain Specification version 1.3.  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.  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 Material(s):  6.2.1 The organization affirms the existence of a program that meets all the requirements of this OpenChain Specification version 1.3 within the past 18 months of achieving conformance validation.  Rationale:  It is important for the organization to remain current with the specification if that organization wants to assert program conformance over time. This requirement ensures that the program’s supporting processes and controls do not erode if an organization continues to assert program conformance over time. | 6.1 Damit eine Organisation ein OpenChain-konformes Programm vorweisen kann, muss sie bestätigen, dass ihr Programm die in dieser OpenChain-Spezifikation Version 1.3 beschriebenen Kriterien erfüllt.  Verifikationsmaterial:  6.1.1 Die Bestätigung des Vorhandenseins eines Programms, welches alle Anforderungen dieser OpenChain Spezifikation Version 1.3 erfüllt.  Begründung:  Es soll sichergestellt werden, dass ein FOSS-Programm alle Anforderungen dieser Spezifikation erfüllt, wenn eine Organisation angibt, ihr Programm sei OpenChain Conforming. Lediglich Teile der Anforderungen zu erfüllen, wird nicht als ausreichend angesehen werden.  6.2 Die Übereinstimmung mit dieser Version der Spezifikation ist ab dem Datum der Validierung der Konformität für 18 Monate gültig. Die Anforderungen der Validierung der Konformität finden Sie auf der Website des OpenChain-Projekts.  Verifikationsmaterial:  6.2.1 Die Organisation bestätigt, dass ein FOSS-Compliance-Programm existiert, das alle Anforderungen dieser OpenChain Spezifikation Version 1.3 während der vergangenen 18 Monate seit Erreichen der Konformitäts-Validierung erfüllt.  Begründung:  Es ist wichtig, dass die Organisation auf einem aktuellen Stand bezüglich der Spezifikation bleibt, wenn sie als Organisation die Programmkonformität auf Dauer behaupten will. Diese Anforderung stellt sicher, dass die die Konformität unterstützenden Prozesse und Kontrollen des Programms nicht abgeschwächt werden, wenn eine Organisation ihre Konformität auf Dauer behaupten möchte. |

|  |  |  |
| --- | --- | --- |
| Appendix I: Language Translations | Appendix I: Language Translations | Anhang I: Sprachübersetzungen |
| 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. | 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. | Um die globale Adoption zu erleichtern, begrüßen wir die Bemühungen, die Spezifikation in mehrere Sprachen zu übersetzen. Da auch die OpenChain Initiative wie ein Open Source Projekt aufgesetzt ist, werden Übersetzungen durch diejenigen gesteuert, die bereit sind, ihre Zeit und ihr Fachwissen zu Übersetzungen unter den Bedingungen der CC-BY 4.0-Lizenz und der Richtlinie des Projekts für Übersetzungen beizutragen. Die Details der Richtlinien und der verfügbaren Übersetzungen finden Sie auf der Spezifikations-Webseite des OpenChain-Projekts. |