OPENCHAIN

기업 공개SW 거버넌스 가이드

OpenChain 해설서

OpenChain 해설서

SK텔레콤 장학성

OPENCHAIN

OPENCHAIN

목 차

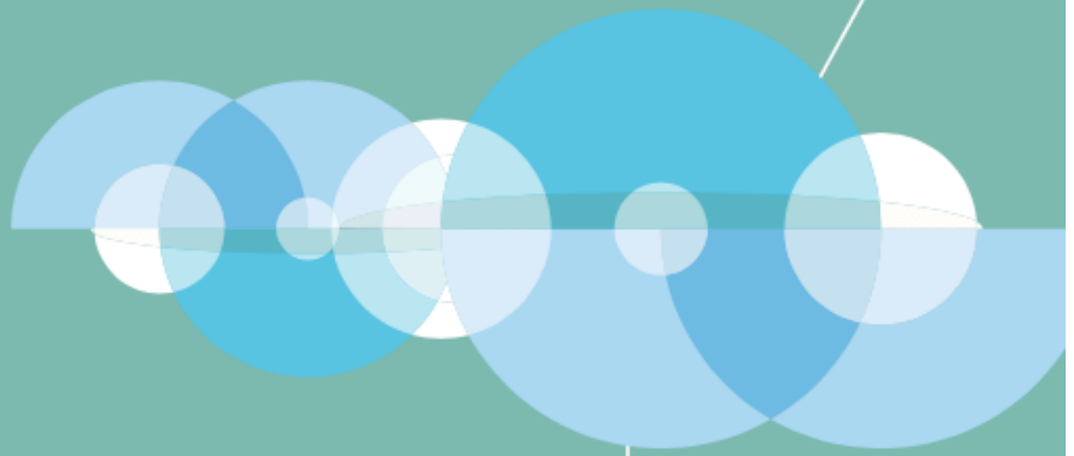
Ⅰ. OpenChain Project란?

오늘날 소프트웨어는 갈수록 그 규모와 복잡도가 커지고 있다. 하나의 소프트웨어를 개발하기 위해서는 자체 개발하는 소프트웨어뿐 아니라 오픈소스, 3rd party Software, 반도체 벤더의 SDK 등 소프트웨어 공급망에 걸친 다양한 소프트웨어가 사용될 수 있기 때문이다.

이러한 복잡한 소프트웨어 공급망의 조직 중 한 곳이라도 라이선스 의무를 준수하지 않거나, 올바른 오픈소스 정보를 제공하지 못한 경우, 최종 소프트웨어를 배포하는 기업은 라이선스 준수에 실패하고 이로 인해 제품 판매가 중단되는 상황이 발생할 수 있다. 실제로 2009년 12월, Busybox라는 오픈소스 관련된 소송이 있었다. Busybox는 임베디드 시스템에 광범위하게 사용되고 있는 GPL-2.0 라이선스가 적용된 오픈소스인데, 두 곳의 국내 회사를 포함하여 총 14개 회사가 소송 대상이 되었다. 이 사례에서 주목할만한 점은 이 중에는 제품을 직접 개발하지 않고 배포만 한 회사도 소송을 당했다는 점이다.

이와 같은 복잡한 소프트웨어 공급망 환경에서는 어느 한 기업이 아무리 훌륭한 프로세스를 갖추고 있다고 해도 자체적으로 완벽한 오픈소스 컴플라이언스를 달성하는 건 매우 어렵다. 결국 소프트웨어를 최종 배포하는 기업이 오픈소스 컴플라이언스를 제대로 이행하기 위해서는 소프트웨어 공급망의 모든 구성원이 라이선스 의무를 준수하고 올바른 오픈소스 정보를 제공하여 공급망 전체에 신뢰가 구축되어야 한다.

OPENCHAIN



II. Openchain Specification 준수 방법

1. 프로그램 설립 (Program Foundation)
2. 관련 업무 정의 및 지원
   (Relevant Tasks Defined and Supported)
3. 오픈소스 콘텐츠 검토 및 승인 (Open Source
   Content Review and Approval)
4. 컴플라이언스 결과물 생성 및 전달
5. 오픈소스 커뮤니티 참여에 대한 이해
6. 설명서 요건 준수

Ⅱ. OpenChain Specification 준수방법

OpenChain Specifiation에서는 오픈소스 컴플라이언스를 위한 핵심 요구 사항을 정의한다. OpenChain Specification을 준수한다고 인정받은 기업은 소프트웨어 솔루션을 배포하는 조직간에 신뢰를 제공할 수 있게 된다. 여기에서는 기업들이 OpenChain Specification을 준수하기 위해 충족해야 하는 여섯가지 주요 요건과 그 방법을 세부적으로 설명한다.

1 프로그램 설립 (Program Foundation)

1) 1.1 정책 (Policy)

오픈소스를 이용하여 소프트웨어를 개발하고 배포하는 기업이라면 오픈소스를 관리하기 위한 정책과 프로세스를 구축하고, 이를 위한 인력과 자원을 할당해야 한다. OpenChain에서는 이러한 일련의 활동을 관리하는 체계를 오픈소스 프로그램이라고 부르고, OpenChain Specification을 준수하기 위한 첫번째 요건은 바로 이 프로그램을 설립해야 하는것이다. 여기서 오픈소스 프로그램이란 정책, 프로세스, 인원 등 기업이 오픈소스 컴플라이언스 활동을 수행하기 위한 일련의 관리 체계를 의미한다.

OpenChain Specification에서는 이를 입증하기 위한 자료로 우선 문서화된 오픈소스 정책을 요구한다. 이 안내서에서는 참고를 위해 OpenChain Specification의 요건을 충족하는 오픈소스 정책 문서 예시를 "[부록 01] 오픈소스 정책 for OpenChain 2.0 (예시)" 에서 제공한다. OpenChain Specification은 이어지는 장에서 오픈소스 프로그램이 갖춰야할 요건들을 설명하고 있다.

OPENCHAIN

[부록 01]
오픈소스 정책 for OpenChain 2.0 (예시)

이 오픈소스 정책 for OpenChain 2.0(샘플)은 다음 두가지 자료를 참고하여 작성하였다.

1. https://github.com/OpenChain-Project/curriculum/tree/master/policy-template
2. https://github.com/todogroup/policies/blob/master/linuxfoundation/lf_compliance_generic_policy.pdf

○○회사 오픈소스 정책

1 목 적

이 정책은 오픈소스를 사용하는 조직 전체가 오픈소스 컴플라이언스 활동을 수행하도록 수립되었다. 또한 이 정책은 직원들이 오픈소스의 가치를 이해하게 하고, 오픈소스 커뮤니티에 기여하기 위한 방법을 제공한다.

〈○○회사〉의 직원은 이 정책의 근거와 내용을 이해하고 필요한 활동을 충실히 수행함으로써 정책의 효과 및 회사의 컴플라이언스 수준 향상에 기여한다.

이 정책을 준수하는 것은 중요하다. 준수하지 않을 경우 다음과 같은 상황을 초래할 수 있다.

- 사용 중인 코드에 대한 저작권 또는 기타 지식재산권 보유자의 법적 클레임
- 고객으로부터의 클레임
- 회사 독점 코드의 의도치 않은 공개
- 라이선스 의무 위반으로 인한 벌금 부과
- 평판 손실
- 수익 손실
- 공급업체 및 고객과의 계약 위반

5) 1.5 라이선스 의무 (License Obligations)

OpenChain Specification

1.5 라이선스 의무

각 라이선스에 의해 부여된 의무, 제한 및 권리를 결정하기 위해 식별된 라이선스를 검토하는 프로세스가 존재한다.

입증 자료:

1.5.1 각 식별된 라이선스에 의해 부과되는 의무, 제한 및 권리를 검토하고 문서화하기 위한 문서화된 절차.

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):

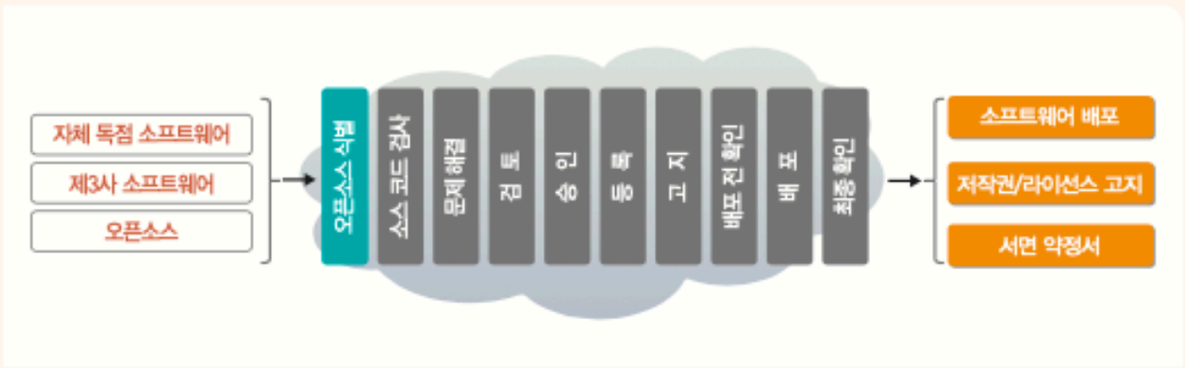
1.4.1 A written statement that clearly defines the scope and limits of the Program.

오픈소스의 사용 가능 여부를 판단하기 위해서는 먼저 오픈소스의 라이선스가 무엇인지 식별하고, 라이선스가 요구하는 의무사항을 검토하고 확인해야 한다. 오픈소스 프로그램은 소프트웨어 개발팀에서 오픈소스 라이선스가 부여하는 의무, 제한 및 권리를 검토할 수 있도록 오픈소스 라이선스 의무 요약 자료를 제공하는 것이 좋다. 공개SW 라이선스(https://www.oss.kr/oss_license)에서는 주요 오픈소스 라이선스의 의무, 제한 및 권리를 자세히 설명한다.

오픈소스를 사용하기에 앞서 라이선스 검토하고 이를 문서화하는 절차는 "[부록 02] 오픈소스 컴플라이언스 프로세스 (예시)" 절차의 오픈소스 식별 단계에 해당한다.

오픈소스 컴플라이언스 프로세스(예시)

Step 1. 오픈소스 식별 (Identification of Open Source)



오픈소스 식별 단계는 오픈소스 컴포넌트를 식별하기 위한 검토 단계이다. 자체 독점 소프트웨어인지, 제3자 소프트웨어인지 여부에 관계 없이 공급 대상 소프트웨어에 포함된 오픈소스를 모니터링한다. 오픈소스 식별 방법은 다음과 같다.

- 오픈소스 사용 요청 접수 : SW개발자는 특정 제품에 오픈소스를 사용하고자 함을 오픈소스 책임자 또는 오픈소스 센터에 알리고, 검토 및 승인을 위한 오픈소스 패키지의 용도에 관한 정보를 제공한다.
- 회사 개발 소프트웨어 검사 (Auditing) : 개발자가 오픈소스의 소스코드를 복사해서 가져와 소프트웨어를 개발할 수 있기 때문에 회사가 개발한 소프트웨어에 대해서도 검사를 수행한다.
- 제3자 소프트웨어 실사 (Due diligence)

식별 단계 시작 조건	식별 단계 결과
다음 조건 중 하나에 의해 식별단계를 시작한다. • 개발자로부터 특정 오픈소스 사용 요청 접수 • 개발 프로세스 상 소프트웨어 검사 단계 • 제3자 소프트웨어 입수 및 개발소프트웨어로의 통합	• 오픈소스에 대한 컴플라이언스 기록 생성 (Jira 등 활용) • 소스코드 스캔 대상 선정 및 요청

3 오픈소스 컨텐츠 검토 및 승인 (Open Source Content Review and Approval)

1) 3.1 BOM (Bill of Materials)

OpenChain Specification

3.1 BOM

공급 대상 소프트웨어를 구성하는 각 오픈소스 컴포넌트(및 식별된 라이선스)를 포함하는 BOM을 작성하고 관리하는 프로세스가 있다.

입증 자료:

3.1.1 공급 대상 소프트웨어를 구성하는 오픈소스 컴포넌트 모음에 대한 정보를 식별, 추적, 검토, 승인 및 보관하는 문서화된 절차

3.1.2 공급 대상 소프트웨어에 대해 문서화된 절차가 적절히 준수되었음을 입증하는 오픈소스 컴포넌트 기록.

3.1 Bill of Materials

A process exists for creating and managing a bill of materials that includes each Open Source component (and its Identified Licenses) from which the Supplied Software is comprised.

Verification Material(s):

3.1.1 A documented procedure for identifying, tracking, reviewing, approving, and archiving information about the collection of Open Source components from which the Supplied Software is comprised.

3.1.2 Open Source component records for the Supplied Software that demonstrates the documented procedure was properly followed.

오픈소스 컴플라이언스 활동의 가장 기본은 바로 공급 대상 소프트웨어에 포함된 오픈소스 현황을 파악하는 것이다. 공급 대상 소프트웨어에 포함된 오픈소스와 그 라이선스를 식별하여 그 정보를 담고있는 BOM을 작성하고 관리하는 프로세스를 구축해야 한다. 공급 대상 소프트웨어마다 어떤 오픈소스가 포함되어 있는지 알고 있어야 소프트웨어를 배포할 때 각 라이선스가 요구하는 의무 사항을 준수할 수 있기 때문이다. 모든 오픈소스는 배포 대상 소프트웨어에 통합하기 전에 검토 및 승인되어야 한다. 오픈소스의 기능, 품질 뿐만 아니라 출처, 라이선스 요건을 충족하는지 검토가 되야 한다. 이를 위해 검토 요청 → 리뷰 → 승인 과정이 필요하다. [부록 02]에서는 기업의 오픈소스 컴플라이언스를 위한 프로세스 전과정에 대해 설명하고 있다. 식별부터 등록까지의 과정을 통해 BOM을 작성하고 관리하게 된다.

공급 대상 소프트웨어에 포함된 오픈소스 목록은 문서화하여 보관해야 한다. Eclipse 재단에서 후원하는 오픈소스 프로젝트인 SW360(https://projects.eclipse.org/proposals/sw360)은 공급 대상 소프트웨어별로 포함하고 있는 오픈소스 목록을 트래킹할 수 있는 기능을 제공한다. SW360 사용 방법은 [부록 03]을 참고할 수 있다.

오픈소스 컴플라이언스 프로세스의 모든 과정과 결과는 문서화가 되어야 한다. 이메일을 사용하는 것 보다는 Jira, Bugzilla 등의 이슈 트래킹 시스템을 이용하는 것이 이러한 과정을 효율적으로 문서화 할 수 있다.

2) 3.2 라이선스 컴플라이언스

OpenChain Specification

3.2 라이선스 컴플라이언스

프로그램은 공급 대상 소프트웨어에 대해 소프트웨어 공급 담당자가 접하게 되는 일반적인 오픈소스 사용 사례를 관리할 수 있어야 하며, 다음과 같은 사례가 포함될 수 있다(이 목록이 완전한 것은 아니며, 모든 사용 사례가 적용되어야 하는 것은 아니다).:

- 바이너리 형태로 배포;
- 소스 형태로 배포;
- Copyleft 의무를 발생시킬 수 있는 다른 오픈소스와 통합;
- 수정한 오픈소스를 포함;
- 공급 대상 소프트웨어 내에서 상호 작용하는 다른 컴포넌트와 호환되지 않는 라이선스 하의 오픈소스 또는 기타 소프트웨어를 포함;
- 저작자 표시 요건이 있는 오픈소스를 포함.

입증 자료:

3.2.1 공급 대상 소프트웨어의 오픈소스 컴포넌트에 대해 일반적인 오픈소스 라이선스 사용 사례를 처리하기 위한 문서화된 절차.

3.2 License Compliance

The Program must be capable of managing 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):

A documented procedure for handling the common Open Source license use cases for the Open Source components of the Supplied Software.

오픈소스 라이선스를 제대로 준수하기 위해서는 오픈소스 라이선스 별로 요구하는 사항에 대해 정확히 알고 있어야 한다. 개별 소프트웨어 개발자가 이를 일일이 파악하는 것은 어렵기 때문에 오픈소스 책임자는 자주 사용되는 오픈소스 라이선스 들에 대해 일반적인 사용 사례별 요구사항/주의사항을 정리하여 회사 내부에 공유하는 것이 좋다. 오픈소스 책임자는 자주 사용되는 오픈소스 라이선스별로 일반적인 사용 사례에 대한 의무 요약 자료를 제공한다. 오픈소스 라이선스에 대한 일반적인 가이드와 라이선스 의무 요약 자료는 NIPA에서 제공하는 "공개SW 라이선스 가이드"를 참고할 수 있다. (https://www.oss.kr/oss_license) [부록 2] 오픈소스 컴플라이언스 프로세스 (예시)의 오픈소스 컴플라이언스 프로세스의 식별, 검사, 문제해결, 리뷰, 승인 단계를 통해 공급 대상 소프트웨어의 오픈소스 컴포넌트에 대해 일반적인 오픈소스 라이선스 사용 사례를 처리할 수 있다.

식별 및 검사 단계에서는 소스코드 스캔 도구를 사용할 수 있다. 소스코드 스캔 도구는 무료로 사용할 수 있는 오픈소스 기반 도구부터 상용 도구까지 다양하게 있다. 각 도구들은 특장점 들이 있지만 어떤 하나도 모든 문제를 해결할 수 있는 완벽한 기능을 제공하지 않는다. 따라서 기업은 제품의 특성과 요구사항에 맞는 적합한 도구를 선택해야 한다. 많은 기업들이 이러한 자동화된 소스 코드 스캔 도구와 수동 검토를 병행하여 이용한다. Linux Foundation의 FOSSology Project는 오픈소스로 공개된 소스 코드 스캔 도구로서 기업들이 손쉽게 무료로 사용할 수 있다. 사용 방법은 [부록 03]의 FOSSology 사용 방법을 참고할 수 있다.

(https://www.fossology.org/)

[부록 03]

오픈소스 도구(FOSSology, SW360)

오픈소스 컴플라이언스 활동을 위해서는 정책, 프로세스나 교육자료뿐만 아니라 소스코드 스캔, Dependency 분석, 오픈소스 BOM 관리 등을 위한 다양한 도구와 시스템도 요구된다. 때문에 다수의 기업이 이러한 도구와 시스템을 도입하고 활용하는데 많은 리소스를 투입하고 있다. 특히 오픈소스 컴플라이언스를 처음 시작하는 기업은 프로세스뿐 아니라 비용 측면에서도 어려움을 겪고 있다.

이런 어려움을 해결하기 위해, 2019년 6월, OpenChain 프로젝트에 참여하고 있는 지멘스, 보쉬, 도시바, 후지쓰, 히타치 등의 오픈소스 컴플라이언스 도구 전문가들을 주축으로 OpenChain Tooling Work Group이 시작되었다.

OpenChain Tooling Work Group은 여러 기업의 오픈소스 전문가들이 이슈를 함께 해결하고 결과물을 공유해 오픈소스 컴플라이언스 비용을 절감하고 양질의 컴플라이언스 결과물을 만들어 내기 위해 구성되었다.

구체적으로는 FOSSology, SW360, Software Heritage, ClearlyDefined, SPDX 등의 기존 오픈소스 프로젝트를 활용하여 통합(turn-key) 오픈소스 툴 체인을 만들고, 모든 기업이 이를 자유롭게 사용할 수 있도록 하는 것을 목표로 삼고 있다.
(https://groups.io/g/oss-based-compliance-tooling)

여기서는 FOSSology와 SW360에 대해 소개 및 간단한 사용 방법에 대해 알아본다.

Question?

OPENCHAIN