OpenChain Compliance Automation

Automating the generation of OpenChain compliance artefacts

Alexander Murphy

2023-09-06

OpenChain

International standard (ISO/IEC 5230) lists requirements for a quality open source licence compliance program. The project homepage is here: https://www.openchainproject.org/; and the specificaiton and workgroup manage the development publicly on GitHub here: https://github.com/OpenChain-Project/License-Compliance-Specification.

The standard provides the following definition:

> 2.1 - compliance artifacts a collection of artifacts that represent the output of a compliance program and accompany the supplied software
>
> Note: The collection may include (but is not limited to) one or more of the following: attribution notices, source code, build and install scripts, copy of licenses, copyright notices, modification notifications, written offers, open source component bill of materials, and SPDX documents.

These data are what we will automate in this demonstration (attribution notices and licence texts specifically, other artefacts may be required but this is not an exhaustive exercise, regardless, it is highly unlikely that the steps in this example will map exactly to your own use-case).

Tools

OpenChain is a non-prescriptive standard. To (ab)use computer science terminology, OpenChain is declarative in nature, it tells you *what* to do, but you decide *how* to go about it. This example is a generalised method (rather, a collection of tools and processes) used frequently at Orcro to generate compliance artefacts.

Spelling

Artefacts - British engligh, artifacts, american english. Former used throughout.

Scancode toolkit

R

Scripts

GitHub

Overview

We begin by identifying what data we need to automate. This step is usually resource-intensive, so an arbitrary (and vetted) software product will be used for this example.

Then we break-down the full pipeline into "components", and semi-manually run these to illustrate what the tools are doing.

The output of the process (the compliance artefacts) will be shown. These will be in the form of plain text files, which can be compressed if required.

Setting expectations

Implementing the process on a CI system is not covered.

This only covers generation of compliance artefacts for the application itself. It is likely that additional FOSS is used in any shipped product (Docker container, GPU drivers, etc.) but this process is for the application layer only.

We won't consider legal requirements such as how the jurisdiction your code is shipped to may affect your obligations.

Identifying the requried artefacts

Questions to ask

What is *distributed*?

Are there any *dependencies*?

How will the code be shipped?

Are there any *snippets* present?

In this case, there is: