Open Source License Compliance - Basics

This document provides the background and an overview of an OSL compliance audit

Software.
Embedded.

Project name: OSL Basics
Version: 2.0

CONTENT

Page

1 Basic Information

1.1 Purpose oft he document

Current document describes goals of the OSL[1] compliance audit, means to achieve, and steps to take in order to eliminate issues.

This document is prepared as an answer to legal requirements to protect intellectual property across the world.

1.2 Protection of intellectual rights

In the information age it is crucial to protect your intellectual property. While the idea of protecting your intellectual property is not new, a wide spread of IT in our lives led to creation of huge amount of new intellectual property, including software. One of the significant differences of software from other intellectual property is that a lot of creators started encouraging usage of their work instead of protecting it from use. Thus, the copyleft licenses appeared as an alternative to copyright.

Copyright is a legal right, existing in many countries, that grants the creator of an original work exclusive rights to determine whether, and under what conditions, this original work may be used by others.

Copyleft, distinguished from copyright, is the practice of offering people the right to freely distribute copies and modified versions of a work with the stipulation that the same rights be preserved in derivative works created later. Copyleft software licenses are considered protective or reciprocal, as contrasted with permissive free-software licenses

1.3 Major Copyleft licenses overview

All following information is taken from https://tldrlegal.com/ and cannot be used as legal advice.

GNU General Public License v2.0 (GPL-2.0)

You may copy, distribute and modify the software as long as you track changes/dates in source files. Any modifications to or software including (via compiler) GPL-licensed code must also be made available under the GPL along with build & install instructions.

GNU Lesser General Public License v2.1 (LGPL-2.1)

This license mainly applies to libraries. You may copy, distribute and modify the software provided that you state modifications and license them under LGPL-2.1. Anything statically linked to the library can only be redistributed under LGPL, but applications that use the library don't have to be. You must allow reverse engineering of your application as necessary to debug and relink the library.

GNU General Public License v3 (GPL-3)

You may copy, distribute and modify the software as long as you track changes/dates in source files. Any modifications to or software including (via compiler) GPL-licensed code must also be made available under the GPL along with build & install instructions.

GNU Lesser General Public License v3 (LGPL-3.0)

This license is mainly applied to libraries. You may copy, distribute and modify the software provided that modifications are described and licensed for free under LGPL. Derivatives works (including modifications or anything statically linked to the library) can only be redistributed under LGPL, but applications that use the library don't have to be.

[1] OSL — Open Source License(s)

MIT License (Expat)

A short, permissive software license. Basically, you can do whatever you want as long as you include the original copyright and license notice in any copy of the software/source. There are many variations of this license in use.

4-Clause BSD

The BSD 4-clause license is a permissive license with a special obligation to credit the copyright holders of the software.

Apache License 2.0 (Apache-2.0)

You can do what you like with the software, as long as you include the required notices. This permissive license contains a patent license from the contributors of the code.

2 Third-party entities and related tools

2.1 OSADL

Open Source Automation Development Lab eG (OSADL) is a German organization intended to promote and coordinate the development of open source software for the machine, machine tool, and automation industry.

OSADL aims to help its members use associated products in compliance with the law. Since violating the conditions of Open Source licenses such as the GNU General Public License (GPL) can lead to the copyright infringement, licensors can forbid the further distribution of non- compliant products. In addition, reputations can be damaged, and related suits can cost a consider- able amount of money.

By using a special auditing process, the OSADL License Compliance Audit (OSADL LCA), companies who use Linux within their embedded systems can determine whether the necessary measures have been taken to satisfy the obligations associated with Open Source licenses. This ensures that Open Source software is used in a compliant manner.

2.2 SPDX

Software Package Data Exchange® (SPDX®) is an open standard for communicating software bill of material information (including components, licenses, copyrights, and security references).

SPDX reduces redundant work by providing a common format for companies and communities to share important data about software licenses, copyrights, and security references, thereby streamlining and improving compliance.

The SPDX specification is developed by the SPDX workgroup, which is hosted by The Linux Foundation. The grass-roots effort includes representatives from more than 20 organizations—software, systems and tool vendors, foundations and systems integrators—all committed to creating a standard for software package data exchange formats.

2.3 FOSSology

FOSSology is an open source license compliance software system and toolkit. As a toolkit you can run license, copyright and export control scans from the command line. As a system, a database and web UI are provided to give you a compliance workflow. In one click you can generate an SPDX file, or a ReadMe with the copyrights' notices from your software. FOSSology deduplication means that you can scan an entire distro, submit a new version, and only the changed files will get rescanned.

2.4 Commercial tools

2.4.1 Black Duck

A commercial utility owned by Synology, advertised as a Complete Open Source Management Solution with next advantages:

- Fully discover all open source in your code
- Map components to known vulnerabilities
- Identify license compliance and component quality risks
- Set and enforce open source policies
- Integrate open source management into your DevOps environment
- Monitor and alert when new threats are reported

In comparison to FOSSology, Black Duck should also look for open source code, copied from some open source product.

2.4.2 WhiteSource

Another commercial utility. Automatically identifies all the open source components and dependencies in the project by constant and automatic cross-referencing of open source components against WhiteSource's definitive database of open source repositories. Provides browser plugin, which reveals any reported bugs, security risks, undesirable licenses (as defined by the company policy) newer versions and more for each component, helping to make better decisions about which component to add to the project.

2.4.3 Snipe IT

The software allows you to manage software licenses and software users across your business. Developers are currently working on easier handling of multi-pack licensing, which will become a powerful tool for large license deployments.

2.4.4 OpenLM

This is an audit and management software specifically designed for managing engineering software licenses. It allows your organization to analyze and monitor software usage, as well as enforce your usage policies.

2.4.5 Other utilities

licensecheck — Debian utility that can scan source code and report found copyright holders and known licenses. Its approach is to detect licenses with a dataset of regex patterns and key phrases (parts) and to reassemble these in detected licenses based on rules. It output results in plain text (with customizable delimiter) or a Debian copyright file format.

Licensecheck[2] (NPM) — A quick way to see the licenses of NPM modules used in the project, recursively.

OSLC[3] — Open Source License Checker is tool for inspection and analysis of license information from open source packages. Not currently supported.

More Linux-specific tools can be found here: https://wiki.debian.org/CopyrightReviewTools.

[2] https://www.npmjs.com/package/licensecheck

[3] https://sourceforge.net/projects/oslc/

3 Development guidelines

When developing software, a license conformity should always be considered by developers, when evaluating libraries and tools to be used in the project. The rule of thumb is that the end-user, upon receiving your binary packages in any form, should also be provided with corresponding information in accordance with license for every Open Source Software (OSS) package included. This can include such items as:

- License text for every utility/library;
- Visible information about the license on start;
- Information about authors of every utility/library;
- Source code;
- Build instructions.

On the other hand, developers should take into consideration the delicacy of handling the intellectual property, which should be guarded against releasing to the public. The most dangerous mistake here can be mixing business logics with open source software and contaminating it with a GPL-like license. Such situation may lead to disclosure of sensitive intellectual property.

Some rules can be formulated as example:

- Binary applications should not be linked with any GPL-software neither statically, nor dynamically, and can only be linked with LGPL-libraries dynamically.
- Firmware contents should be considered wisely, and there is no way to close source code of the Linux driver, for example.
- When delivering SaaS, or using OSS only to build applications (hence, internally), there are no obligations to the end-user.
- When delivering software, guarded with GPL, don't forget to deliver the source code as well (or prepare written offer to provide aforementioned source code, which is usually more complicated approach).

4 Recommendations to the OSL Audit Report contents

In order to understand project state, control risks, and be able to address any accusations swiftly, prepare an OSL Audit Report. There is no need to provide the report together with software, but it can be evaluated by internal structures (e.g. legal team).

Such report can be prepared manually or with use of automated tools. We advise to consider following subjects to be included into the report:

1. OSL Audit Report preparation procedure.
2. High-level architecture of the product.
3. Registry of deliverables (distributed to the end-user).
4. List of 3rd-party software, unmodified, but included as part of deliverables (it is advised to store source code of any such software internally instead of relying on 3rd-party distribution channels).
5. List of 3rd-party software, modified by team, and included as part of deliverables (consider also outlining changes done to the software).
6. List of libraries statically or dynamically linked to the implemented software (it doesn't matter if such libraries are distributed as part of deliverables).
7. Include licensing information for any software and/or libraries, mentioned in the lists (items 4–6), as well as resolutions if the license is followed or not.

Bibliography

[1] https:/tldrlegal.com/

Change History

Version	Description	Author	Date
0.1	Creation	SaM	08.02.2019
1.0	Review and release	DP	11.02.2019
2.0	CI update	DP	04.09.2020