July 7, 2022

Open Source Software Policy Guidance

By Michael S. Pavento, Partner

Introduction

Open Source Software (OSS) is software that is freely available in source-code form for anyone to use, copy, modify, and distribute. Generally speaking, however, OSS is not "public domain" software. Like any other software, OSS is copyrighted intellectual property. Authors have the right to control or condition the use of their original OSS code, and typically do so through license agreements.

Unlike traditional software licenses that seek to limit or prohibit further dissemination of the licensed software and certainly the underlying source code, OSS license agreements generally seek to encourage dissemination and to ensure that the source code remains open and accessible to all. Like any other software acquired from an outside source, the applicable license agreement is the starting point for understanding and managing the obligations imposed upon and the risks undertaken by an organization with respect to OSS.

Types of Licenses

Some OSS license agreements are more permissive and some are more restrictive with respect to the obligations imposed on the licensee. The basic concept common to all OSS license agreements is to ensure all downstream users have the freedom to use, modify, and distribute the licensed OSS. Permissive OSS licenses impose minimal obligations on the licensee, such as obligations to maintain attribution and legal notices and to provide a copy of the license terms. These agreements typically permit modifications to the OSS and allow such modifications to be distributed under any license of the licensee's choosing, whether the same or a different OSS source license or a commercial license.

Highly restrictive OSS licenses, often referred to as "strong copyleft" or "viral" license agreements, impose obligations on the licensee not only with respect to the licensed OSS but also with respect to any works derived from it. "Copyleft" refers to an obligation to make source code available. Strong-copyleft licenses require the licensee to make available the source code of the OSS and the source code of any derivative work thereof, which can include the source code of other software with which the OSS is linked or otherwise combined. The source code must be made available free of charge to downstream recipients with broad permissions to modify and redistribute it.

Some OSS licenses fall between the permissive and highly restrictive ends of the OSS license spectrum. These license are usually referred to as "weak copyleft" license agreements. They require the licensee to make the source code of the licensed OSS available to those who

acquire the OSS from the licensee. A weak copyleft requirement, also known as a file-level copyleft requirement, means the obligation to provide source code applies only to the OSS itself and any modifications the licensee might make to the OSS. The requirement does not extend to other software that might be combined, e.g., through dynamic linking, with the OSS.

Distribution & SaaS

It is important to note that virtually all OSS license agreements impose obligations and conditions on the licensee only when the licensee distributes the OSS and/or modifications of it. Therefore, if a company acquires OSS and uses and/or modifies it solely for internal purposes, the company will not be required to take any affirmative actions to comply with the applicable OSS license agreement. Providing software as a service ("SaaS") to customers, where the customers do not receive a copy of the underlying software, is not considered a distribution of the software. Therefore, most OSS licenses do not impose conditions of use on the licensee in a SaaS use case. However, some OSS licenses, most notably the *GNU Affero General Public License* and the *Server-Side Public License*, impose requirements, including copyleft requirements on the licensee, even in SaaS use cases.

OSS Policies

Compliance with OSS licensing requirements is generally rather simple: when OSS is distributed or, if applicable, used in a SaaS offering, alone or as part of a larger product, the licensee must maintain or reproduce attribution and license notices, provide disclaimers to down-stream users, and, when required by the applicable license, make source code available, etc. The hard part: management of the compliance process and minimizing risk associated with noncompliance.

It is critical to ensure OSS license compliance prior to product ship; non-compliance is expensive. Failure to comply with OSS license requirements could subject the company to liability for breach of contract claims and/or IP infringement claims. Remediation requires the reallocation of costly resources and legal expenditures. A structured compliance program is therefore a must. The program must be managed by, or at least include input from, legal and technical personnel who understand license-specific nuances and technical issues.

A best practice it to establish and publish throughout the company a formal OSS policy and required usage and compliance procedures. This involves establishing the infrastructure, work flow and culture, defining risk tolerance and educating employees. When building an OSS policy, the following considerations should be taken into account:

- Approval and exception request process

  In many cases, corporate policies will leave at least some of the decision making authority to the software development team. For example, an OSS policy may include a list of pre-approved OSS licenses and use cases. Use of OSS governed by well-known

permissive licenses may be pre-approved for all uses. Use of OSS governed by weak copyleft licenses may be pre-approved for some use cases, e.g., when the OSS is a dynamically linked library and is used unmodified. Use of OSS in a SaaS offering may be pre-approved under all but a few stated OSS licenses. 3rd party source code that does not carry a license and is not explicit dedicated to the public domain may not, in fact, be OSS and should not be pre-approved for any use cases. A sample listing of OSS license pre-approvals is provided in the next section.

The policy may describe an internal process for seeking approval for use of OSS that is not pre-approved, such as escalation of an approval request through one or more layers of management and IP counsel. In other instances, the policy may be coded into an OSS management system that includes or operates in conjunction with a ticketing mechanism for routing approval requests to appropriate team members, e.g., IP counsel.

- Scanning and auditing

  An OSS policy should require scanning and auditing of critical codebases to identify OSS usage. More often than not, a codebase will include far more OSS than the developers realize. A best practice is to scan the codebase(s) early and often. All types of codebases should be considered: the company's proprietary code, 3rd party code, sample code, firmware, OSS, etc. An audit should utilize both objective and subjective information to validate automated scan results. Linkage analysis and transitive dependencies should be taken into account.

- Use of proprietary 3rd party software

  3rd party software can provide material OSS risk simply because the user is at the mercy of the 3rd party for disclosure of any included OSS and attendant license obligations. A good OSS policy will define and require relevant parties to use reasonable efforts to ensure that any 3rd party suppliers include appropriate representations, warranties, and indemnifications in their commercial software license agreements. If a supplier is unwilling to provide sufficient representations, warranties and indemnifications regarding use OSS in its product, someone within the licensee's organization should be tasked with evaluating whether the use case for the applicable 3rd party software requires scanning of the 3rd party software code to ensure compliance with corporate policy. If the supplier will not provide or permit a source code scan, the responsible party must evaluate whether alternative 3rd party solutions are available on more favorable terms.

  Similarly, the policy should require an OSS scan and audit whenever the company acquires another entity having software-based products or services. Company IP counsel or its designee should be tasked with leading the due diligence review and working with outside counsel as necessary to assess risk associated with onboarding the target entity's software. Input from software developers is usually required.

- Security and vulnerability risks

  Use OSS also presents cybersecurity and data privacy risks. Hackers have access to the OSS source code and can leverage vulnerabilities to gain access to backend system, personal or proprietary data, etc. An OSS policy should require the development team to periodically (ideally, often) check for an upgrade to new versions of and patches to approved OSS. Developers or other compliance personnel should be required to monitor public vulnerability notices, such as Black Duck Security Advisories, and notices published by the National Vulnerability Database (NVB), and the CVE Program. A thorough policy will also specify procedures for monitoring public OSS repositories, such as GitHub, for posting of the company's own proprietary code, security credentials and other sensitive data.

- Contributing to OSS projects

  A company may wish to participate in OSS projects for a variety of reasons, including to help improve critical OSS, to perpetuate the company's own technology and strategies, and/or for public notoriety. Some companies promote their participation in OSS projects in an effort to attract talented software developer recruits. An OSS policy should define a process for approving and tracking employee participation in OSS projects. The policy should require legal and business review to consider, among other things, the nature of a proposed contribution, e.g. simple bug fixes versus robust functional code, the business value or risks associated with participating in the particular OSS project, whether the contribution implicates any of the company's proprietary intellectual property, and whether the company and/or the developer will be required to sign a contributor license agreement or provide a developer certificate of origin (DCO).

- Record keeping

  Maintaining records of OSS used and approved for use by the company and/or contributed by the company to an OSS project is important for many obvious reasons. An OSS policy should specify how, where and for how long OSS usage records should be stored and who can access to those records. The OSS policy should require the use of a central repository for storing copies of all OSS used and any modified versions created by the company. A bill of materials (BOM) should be created before any company software product or service is released or otherwise put into production.

- Compliance procedures

  A strong OSS policy will also outline and designate parties responsible for implementation of OSS license compliance procedures. When software is distributed to

customers or other third parties, someone must be responsible for ensuring that appropriate copyright notices and legal terms are provided for all included OSS. When required by the applicable license, source code or an offer to provide source code must also be provided with the software distribution. If an OSS license imposes conditions on use of the OSS in a SaaS environment, someone must be responsible for understanding and ensuring compliance with those conditions.

Sample OSS License Pre-Approvals (*provided for illustration purposes only; consult with IP counsel – guidance may differ based on company's risk profile*)

Internal Use of OSS is pre-approved in all cases, unless a license violation is flagged by the applicable OSS scanning tool.

* The OSS scanning tool should flag OSS licenses that permit only non-commercial use of the OSS. Use of OSS governed by non-commercial licenses are strictly prohibited.

* In some cases, internal use of OSS, *e.g.*, in a development environment, may result in OSS being imported into or otherwise incorporated within an externally-facing product or service. In those cases, use of OSS in or with the externally-facing product or service must be separately evaluated per the externally-facing use cases defined above and the chart provided below.

Permissive OSS Licenses – The following OSS licenses are pre-approved for all use cases:

- Apache License 2.0
- Apache Software License 1.1
- ASM License
- Boost Software License
- BSD licenses (all versions)
- Creative Commons 0 (CC0)
- Creative Commons Attribution (CC BY)
- DOM4j License (like BSD)
- Eiffel Forum License, v2.0
- EU DataGrid License
- Fair License
- Historical Permission Notice and Disclaimer
- ICU License
- ISC License
- Jaxen License
- Jcup License
- JDOM License
- Jlex License
- jMock License
- MIT license
- MX4J License
- Open SSL License
- Open SSL License + SSLeay License
- PHP License
- Python Software Foundation License
- RelaxNGDatatype License
- SIL Open Font License, Version 1.1
- SSLeay License
- Sun MSV License
- The Legion Of The Bouncy Castle
- The PostgreSQL License
- University of Illinois / NCSA Open Source License
- W3C License
- X.Net

Strong or Weak Copyleft OSS Licenses – The following OSS licenses are pre-approved for use in externally-facing SaaS applications only when the OSS is used as a standalone component (e.g., not linked or interfacing with the SaaS application):

- Affero GNU Public License (AGPLv3)
- European Union Public License v 1.2 (EUPLv1.2)
- Common Public Attribution License 1.0 (CPALv1.0)
- Open Software License 3.0 (OSL-3.0)
- Server-side Public License (SSPL)

Strong or Weak Copyleft OSS Licenses – The following OSS licenses are pre-approved for externally distributed software only in the indicated use cases:

OSS License	SaaS	Stand-Alone	Dynamic Library	Static Library	Snippet
Academic Free License 3.0 (AFL-3.0)	NO	NO	NO	NO	NO
Affero GNU Public License (AGPLv3)	NO	OK	NO	NO	NO
Artistic License 2.0	OK	OK	OK	OK	OK
Carnegie Mellon University License	OK	OK	OK	OK	OK
CECILL-2.1	NO	NO	NO	NO	NO
CNRI Python License	NO	NO	NO	NO	NO
Common Development and Distribution License version 1.1 (CDDLv1.1)	NO	NO	NO	NO	NO
Common Public Attribution License 1.0 (CPALv1.0)	NO	OK	NO	NO	NO
Common Public License	NO	NO	NO	NO	NO
Creative Commons Attribution-ShareAlike (CC BY-SA)	NO	NO	NO	NO	NO
Creative Commons Attribution-NonCommercial-ShareAlike (CC BY-NC-SA)	NO	NO	NO	NO	NO
Eclipse Public License, version 1	NO	NO	NO	NO	NO
Eclipse Public License, version 2	OK	OK	OK	NO	NO
European Union Public License v 1.2 (EUPLv1.2)	NO	OK	NO	NO	NO
GNU General Public License, version 2.0 (GPLv2)	OK	OK	NO	NO	NO
GNU General Public License, version 2.0 (GPLv2) with Classpath Exception	OK	OK	OK	NO	NO
GNU General Public License, version 3.0 (GPLv3)	OK	OK	OK	NO	NO
GNU Library General Public License, version 2.0 (LGPLv2)	OK	OK	OK	NO	NO
GNU Lesser General Public License, version 2.1 (LGPLv2.1)	OK	OK	OK	NO	NO

OSS License	SaaS	Stand-Alone	Dynamic Library	Static Library	Snippet
GNU Lesser General Public License (LGPLv3)	OK	OK	OK	NO	NO
IBM Public License version 1.0	OK	OK	OK	NO	NO
Lucent Public License Version 1.0	NO	NO	NO	NO	NO
Lucent Public License Version 1.02	NO	NO	NO	NO	NO
Microsoft Public License (Ms-PL)	OK	OK	OK	OK	OK
Microsoft Reciprocal License (Ms-RL)	OK	OK	OK	NO	NO
Mozilla Public License version 1.1 (MPLv1.1)	NO	NO	NO	NO	NO
Mozilla Public License 2.0 (MPLv2)	OK	OK	OK	NO	NO
Open Software License 3.0 (OSL-3.0)	NO	OK	NO	NO	NO
Server-side Public License (SSPL)	NO	OK	NO	NO	NO
Simple Public License (SimPL-2.0)	OK	OK	NO	NO	NO
Sun Public License	NO	NO	NO	NO	NO