Open Source Software Audit – Do it now or pay later!

Jonathan F. Ariano
Osborn Maledon, PA

What is Open Source Software (OSS)?

Open source software (OSS) or free and open source software (FOSS) is computer software for which the human-readable source code is made available to the public under license terms permitting the study, modification, and distribution of the software to anyone and for any purpose. The "free" of FOSS refers to the liberty to study, modify and distribute – it does not require that it be made available without cost. The fact that OSS and FOSS can be studied, modified and distributed has created open source communities in which developers continually modify, enhance and troubleshoot the software and contribute these changes back to the community.

As a result, the vast majority of enterprises use OSS components in their own proprietary software applications. Doing so brings a host of benefits, including accelerating development time, reducing costs and producing more stable and secure code. While the use of OSS has seen a significant increase over the last 10 years, compliance with the OSS license terms often has not kept up.

OSS License Structure

Many users who download an OSS component may not appreciate that doing so obligates them to abide by a specific software license associated with its use. Unlike commercial software applications or software-as-a-service (SaaS) offerings, click-through acceptance of OSS license terms and conditions are not typically required to gain access to the OSS code. Instead, the license terms are merely referenced on the download site or embedded in a text file within the download package.

One might ask: if the user does not assent to the license terms, then how are they binding on the user? Let's back into the answer to that question. Software code is protected by copyright laws upon creation. Copyright laws give the author of the code a certain bundle of rights, including the right to make copies, publicly display, modify and make derivative works. Unless the author clearly declares the code as part of the public domain, copying (including downloading), publicly displaying, modifying or creating derivative works of such code is an infringement of the rights of the copyright holder. Therefore, one's use of an OSS component must either be copyright infringement or done pursuant to the terms of the license associated with the OSS component.

While there are hundreds of different OSS licenses, they can be broken down into a small subset of license types. They range from: the most permissive licenses, such as Berkley Software Distribution (BSD), MIT and Apache, which permit almost any use of the OSS component and simply limit warranties and disclaim liabilities; to weakly protective licenses, such as the Mozilla Public License (MPL); to the most restrictive "copyleft"-style licenses, such as the GNU General

Public License (GPL) or the GNU Lesser General Public License (LGPL). Copyleft licenses can require that any software with which they are packaged must also be distributed under the same copyleft license terms. Use of the more restrictive licenses can therefore have a viral effect, requiring disclosure of, and permitting third parties to use, the source code of one's own proprietary code.

Further, different licenses impose different requirements on modification and distribution of the OSS component. For example, the LGPL license requires users to distribute the source code (or otherwise make it available) if they distribute the OSS component to third parties. Other license types require that users identify changes they have made to the OSS component in a header or other text file included in the distribution package.

Risks of Noncompliance

The consequences of breach of a commercial software license are relatively clear: the commercial software vendor can sue for damages resulting from the breach of the license terms. But in the OSS context, what are the real risks as a practical matter? Technically, the copyright owners of an OSS component could sue for infringement and/or breach of the OSS license terms. While these suits have been filed, given the fact that a particular OSS component could easily have hundreds of contributors and hence hundreds of copyright owners without a common voice, litigation is not typical. Instead, compliance with OSS license terms becomes critical in the context of many important transactions, such as financings and mergers and acquisition transactions.

A sophisticated investor or acquirer in any significant financing or M&A transaction will always demand a representation and warranty of OSS compliance. Non-compliance creates potential ambiguity around ownership of a material asset and potential post-closing costs of compliance. That ambiguity and associated remediation costs may affect not only the value of the transaction, but also the decision whether to proceed with the transaction at all. How an enterprise manages its use of OSS can speak volumes as to its policies, procedures, structure and culture, all of which are relevant to successful transaction due diligence.

To avoid the potential for OSS issues to negatively impact an important transaction, enterprises must develop and follow processes to inventory their use of OSS components, analyze their degree of compliance and remediate any non-compliance long before the term sheet stage of any transaction.

OSS Audit

Before any enterprise commences an OSS audit, it needs to educate both developers and management on the benefits and risks of incorporating OSS into their proprietary software applications. Asking individuals to participate in an audit where their prior actions could come under a microscope will be far more successful if they understand and appreciate the importance of the outcome.

OSS Component Inventory

The most significant effort with any OSS audit is inventorying existing use of OSS components. That inventory can occur manually by interviewing each developer and asking them to identify the OSS components they have downloaded and used in the development process. For long standing development teams that have never been through this exercise, that can be a difficult, if not impossible, task. While it may be easy to remember OSS components that were recently incorporated into a development project, that may not be true for software developed years ago.

Alternatively, third party software products exist that can automate the inventory process. Products from vendors such as Black Duck, Palamdia, Protecode and others can scan your software code and, using sophisticated pattern matching algorithms, identify the various OSS components present in your software code. While some vendors require that source code be uploaded to their cloud environment for processing, others can operate entirely on-premises or using hashed values of the source code to avoid the risk of source code disclosure outside the enterprise. The cost of such automated processes is dependent on a number of factors, including the number of lines of code and location of inventory process (SaaS v. on premises) and can range from a few thousand to tens of thousands of dollars.

Unless an enterprise is in the early stages of its development process or has kept an accurate running list of OSS components, the automated process will be far more accurate and complete. Further, in more and more significant financing and M&A transactions, the investors or acquirers themselves are using such automated tools as part of the due diligence process, so to simply assume that the manual inventory process will be "good enough" may be misguided. Representing compliance and possession of an accurate list of OSS components, only to later find out from a counterparty using an automated tool that this is not the case, can be just as bad, if not worse, than not having completed an audit at all.

OSS License Inventory

Whether done manually or via an automated process, the first step is to complete an accurate and complete inventory of each OSS component present in the software. Next, one must identify the license type associated with each OSS component. If the inventory was done manually, then this will require a search of the source website of each OSS component to determine the applicable license type. If the inventory was automated, the automated tools typically identify the license type for you based on their extensive database of OSS components. While there are hundreds of different OSS license types, most OSS inventories tend to consist of only a few dozen license types.

Use Characteristics Inventory

Next, for certain OSS license types, the enterprise must determine how the OSS component is used. These use characteristics typically include:

a) *Was the OSS component modified?* Certain OSS license types require that changes made to the OSS component be identified in a header or other text file. Certain other OSS

license types have different distribution requirements if the OSS component was modified.

b) *Is the OSS component distributed to third parties?* Certain OSS license types, especially the copyleft style licenses, require that the source code of the OSS component be distributed or made available to third parties.

c) *How was the OSS component incorporated into the enterprise's code?* While this can vary based on different programming language, OSS components can typically be incorporated into a developer's code in one of three ways. First, the source code of the OSS component can be copied and pasted into the developer's source code (as is done with the enterprise's own proprietary code). Second, the OSS component can be combined with the developed object code at compile time (when the source code is converted into machine-readable code and combined with other components). Third, the OSS component can be combined with the developed proprietary code at runtime (when the compiled code is combined with other compiled code and executed to produce the running program). Each manner of incorporation can invoke different requirements under the various OSS licenses.

While available automated tools can determine if the OSS component was modified, the other use characteristics are typically determined by interviewing developers and gaining an understanding the development process.

OSS Inventory Report

The OSS component inventory, OSS license inventory and use characteristics are typically summarized in a large spreadsheet with each OSS component on its own line with the following column headings:

- OSS Component Name
- OSS License Type
- Source location of website for OSS component
- Link to the applicable OSS License
- Modified? (Yes/No)
- Distributed? (Yes/No)
- How Incorporated? (source, compile, runtime)

The OSS inventory report is then shared with intellectual property counsel to analyze compliance with the various OSS licenses.

OSS Compliance Analysis and Implementation

Intellectual property counsel will then review each OSS license against the use characteristics of each OSS component to determine the requisite compliance obligations. Depending upon the license type, those obligations may include:

- A notice of original copyright and/or attribution to the original author in header files, documentation or other user interface elements, such as "about" boxes.
- A copy of the OSS license in any distribution package to third parties.
- A copy of the source code of the OSS component in any distribution to third parties.
- Limitations on use of trademarked names without permission.
- Identification of changes made to the OSS component.

While this may seem like a daunting task, because most OSS components fall under a handful of OSS license types, experienced intellectual property counsel can accomplish the task in less time than one might think. The output from this analysis will typically be a list of recommended actions, such as:

- A complete list of all original copyright and/or attribution notices to include in header files, documentation or other user interface elements, such as "about" boxes.
- A complete copy of all OSS licenses to be included in any distribution to third parties.
- A mechanism to share or make available copies of the source code of certain OSS components.
- A list of trademarked names that should not be used with the software product.
- A complete list of changes made to certain OSS components.

The recommended actions should then be incorporated into the enterprise's software development process to ensure compliance with all applicable OSS licenses on an ongoing basis.

OSS License Incompatibility

As part of the OSS audit, it is common to come across certain types of OSS licenses that could be fatal to an enterprise's future direction or that are simply incompatible with one another.

For example, as part of the OSS audit, counsel might discover that a developer has included a certain OSS component licensed under GPL that was incorporated with proprietary code at compile time. Based on the terms of the GPL license, the entire software package must be licensed under the GPL license and ALL of the software package's source code, including the proprietary code, must be disclosed to third parties. For most software companies, the source code of proprietary software is the crown jewel of the enterprise and any required disclosure would significantly reduce the value of the company.

In addition, there are certain OSS license types that are inherently incompatible with others. For example, an OSS component licensed under MPL combined with an OSS component licensed under GPL cannot be distributed without violating the terms of one of the licenses.

In either situation, the OSS component causing the compliance issue will typically need to be replaced with another OSS component with a compatible license or rewritten with proprietary code. The potential that this remediation work will be required is a prime reason to commence and compete an OSS audit well in advance of a significant transaction.

Implement an Open Source Policy

After the initial OSS audit is completed and the enterprise is in compliance with all of the licenses applicable to the OSS components used in its software products, management should implement an Open Source Policy. In its simplest terms, an Open Source Policy is a set of written rules that governs the management of OSS (both use of and contribution to) with detailed specifications as to how an enterprise will implement these rules on a daily basis. That policy should include a list of permitted and banned OSS license types that developers can reply upon when choosing whether to use and incorporate a specific OSS component. Any OSS license types not on the permitted or banned list should be reviewed with intellectual property counsel prior to their use. As part of the policy, the OSS inventory report created from the initial audit should be continually updated as a living document.

The policy should also incorporate periodic training of OSS considerations for developers. This will keep an open dialogue between developers and intellectual property counsel as new OSS license types emerge and existing OSS licenses are tested in the court systems.

Conclusion

Given that the OSS audit process can easily take months to complete, learning about OSS compliance issues on the eve of, or even worse, in the middle of, a financing or M&A transaction can cost the enterprise significantly in terms of decreased value, as well as put the entire transaction at risk. Alternatively, completing the OSS audit process at a time with minimal outside pressures and being able to provide an accurate and complete OSS inventory report as part of the due diligence process can help ensure a successful transaction.