Open Source Management Practices Survey

What R&D Teams Are Doing, And Why Their Results Are Poor Despite Their Efforts

Executive Summary

Our research shows that while virtually all developers use open source extensively, and while most companies spend substantial resources on managing their open source inventory (often implicitly), this effort is largely ineffective, resulting in unnecessary risk, as well as too much work and undue cost.

We summarize results from more than 120 respondents.

- Most companies (74%) spend time and sometimes money to ensure they are properly managing their open source adoption; 12.5% have developed some internal tools; and 10% have purchased software for that purpose
- Still, most companies (53%) do not have an up-to-date inventory of all the open source libraries they use. Due to the large effort involved, another 29% produce such inventory only once in a few months, e.g., for a major release, or for M&A or OEM, and so most of the time their inventory is stale. Only 18% become aware of new open source as soon as it is first used by developers. (Note: most of the latter are WhiteSource customers, which was not so beforehand).
    - ➢ Without maintaining an up-to-date and comprehensive inventory, it is impossible to enforce either a licensing policy or a vulnerability patch program, so as to reduce exposure to legal, security, and quality risks.
    - ➢ Late detection of rogue open source may require substantial and expensive work, and is likely to introduce delays into a time-critical release process or major business transactions.
    - ➢ Not knowing about security vulnerabilities in their products (new and previously shipped) may, of course, result in substantial damages and loss of reputation.
- Most companies do not have a clear *policy* with regard to open source licenses, security, and update/patches, or leave these issues to the (often informal) responsibility of individual developers.
    - ➢ When it comes to licensing, 75% do not have a clear policy, and only 9% use automated tools to enforce the policies they have.
    - ➢ When it comes to security vulnerabilities, 74% do not have a process for knowing about security vulnerabilities that arise in open source they use; 13.5% react to such issues when told; and only 12.5% are actively monitoring for security issues (note: most of these respondents are WhiteSource customers that did not have such information beforehand).
- Many companies lack management visibility and consistent governance. In 81% of the cases, open source management is left to individual developers or low-level development teams. Only 19% of companies have central guidance and oversight.
    - ➢ The result is inconsistent treatment, which is a sure recipe for license incompliance, risk to the company's own intellectual property, and substantial risk of untreated defects and security vulnerabilities. Individual developers are not skilled in such matters; furthermore, they don't like to perform these chores, and they cannot be expected to do this well over time.

Background

Many companies spend significant efforts (though they are often unaccounted for) tracking open source usage. These efforts are usually manual and laborious, done by the wrong people (developers!), often at the wrong time (in the crunch of a release or OEM/M&A transaction), and are therefore very expensive and extremely ineffective. Most importantly, judging by the results, they simply do not do the job.

While helping companies better manage their open source usage, we at WhiteSource constantly encounter the following situations:

- There is always a significant gap between what open source people think they use, and what they actually use. Clearly, without knowing which open source components go into their product, they cannot adequately manage and protect themselves against legal, security, and hence business risks.
- Most companies do not have a well-designed open source adoption policy, and those that have such a policy find it very difficult to enforce it across development teams.
- Prior to using WhiteSource, virtually none of our customers had a mechanism for knowing about open source security vulnerabilities that were discovered – even *after* they were already part of their product (i.e., after the software was released).

Earlier WhiteSource research (please ask us for these reports) has shown that

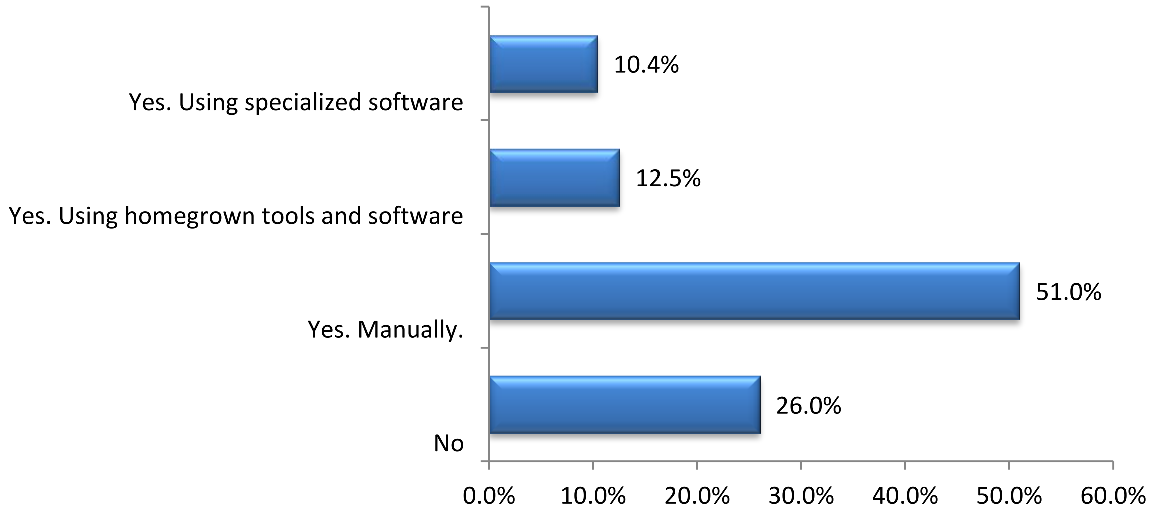
- 91% of software projects contain indirect open source dependencies, and in 64% of these, dependencies have a license different from the root library.
- 24% of software projects contain open source with known security vulnerabilities; in 98% of these cases there exists a later version with a fix
- 85% of software projects relied on at least one open source library that was out of date.

So there remain the questions: “What do R&D teams do today?”, “Why it is so ineffective?”, and “What can we do better?”

In this survey, we set out to understand common practices employed by software development companies when it comes to adopting open source components/libraries, and how well they are prepared to address potential legal, security, and technical issues. We also try to explain why those efforts are so ineffective, and what can be done better.

Sample Questions and Answers Distribution

Does your company manage the use of open source components by developers?

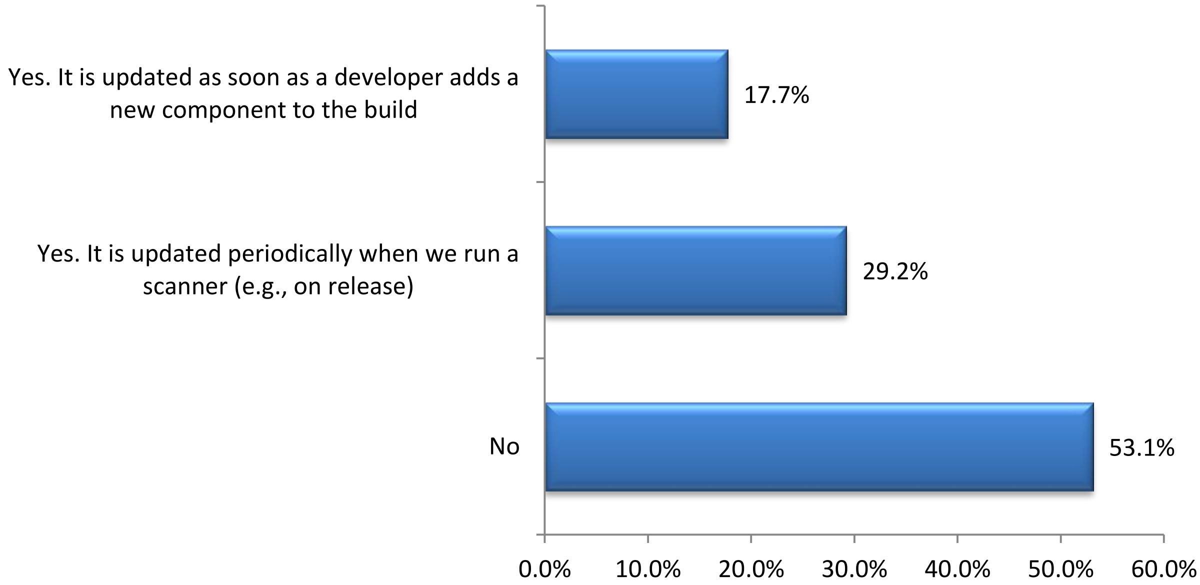


Our Analysis:
Most respondents (77%) manage open source manually, or claim they do not manage it at all (in many cases, there is no awareness of local efforts in individual projects). In most cases, we find that team leaders are managing "lists" of open source that their developers tell them they have adopted.

- WhiteSource Audits invariably find these lists to be inaccurate. First, developers often forget to report on new libraries they have adopted. Second, when they do report, they almost always report the library they chose to use, but not its many dependencies (according to White Source earlier research, 64% of such dependencies will have a license that is different from the reported parent). And third, these lists are always stale, except maybe when they are first created.
- Creating these lists by individual development teams requires substantial coordination, and takes significant research by developers. This is not an easy, nor a pleasant job, and so developers often are unable to find or otherwise do not report all details concerning all libraries in use. This makes it impossible to enforce reasonable open source adoption policies. Not to mention that maintaining these lists is outright impossible.
- Even if it were somehow possible to ensure these lists are accurate and well maintained, without automation it would still be very difficult to enforce a cross-organization policy with regard to licenses' acceptance and usage guidelines (e.g., dynamic linking, attribution requirements, etc.).
- And even if licensing were fully managed, we were never able to find a person who continues to follow every open source, so as to raise a flag when a security vulnerability is discovered, or if there is an update that fixes such a vulnerability, bug, or performance issue, etc.

Last, there is actually a good reason why most companies have not yet adopted professional open source management solutions. Until recently, available solutions were based on source code scanning technology, and usually removed from the software development lifecycle. By their nature, these solutions are very expensive to deploy, and extremely difficult and expensive to operate.

Do you track an up-to-date inventory of open source components used in your product?



Our Analysis:
Most companies (53%) do not track their up-to-date inventory of open source libraries. And if they do, they only do so occasionally, e.g., for a release, or for an M&A or OEM deal.

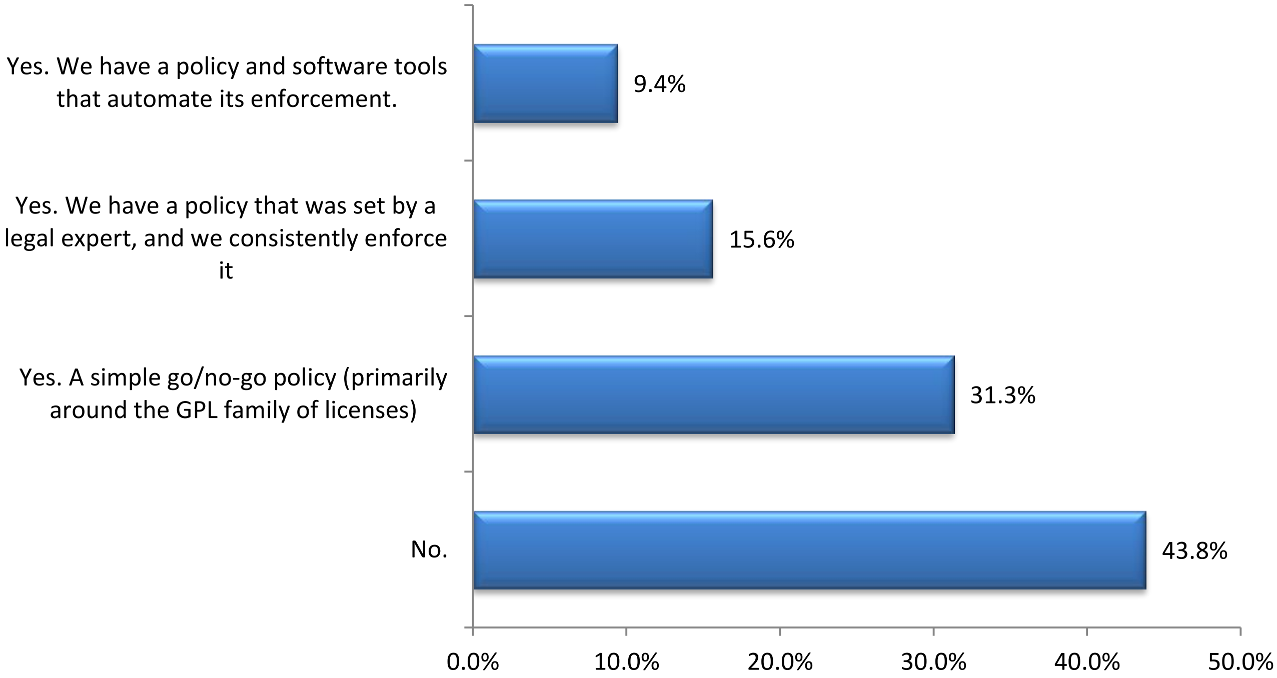
One downside to this oversight is the significant licensing and security risks. But in addition, let's consider what happens if you use a scanner once every few months. Suppose you discover an open source that represents a licensing or security issue. You now need either to find a perfect replacement, or spend a great deal of time to develop the same capabilities on your own. In either case, you would be forced to throw away all the effort of embedding the replaced component, and developing new interfaces, etc.

So why are companies not keeping a continuously updated inventory of their open source components? We believe that this has to do with the fact that traditional technologies, based on source code scanning, are very difficult and expensive to run. Often times, a code scanner will identify thousands of "potential matches" between the code written by the company's developers and one of millions of open source projects. The developers, or release managers, will then have to sift through these just to determine if any of them are for real (usually not).

This is obviously too difficult a task to perform on a daily basis. Fortunately, new technologies such as WhiteSource plug directly into most common continuous integration and build servers, and automate open source detection as part of the software development lifecycle. In doing so, the entire burden is removed from developers, who can continue to focus on coding, while at the same time ensuring that the company has an up-to-the-minute inventory of all open source modules that are in use in each of its specific products.

Once you have a continuously updated inventory, it becomes much easier to truly enforce a licensing policy, acceptance criteria, security patches, etc.

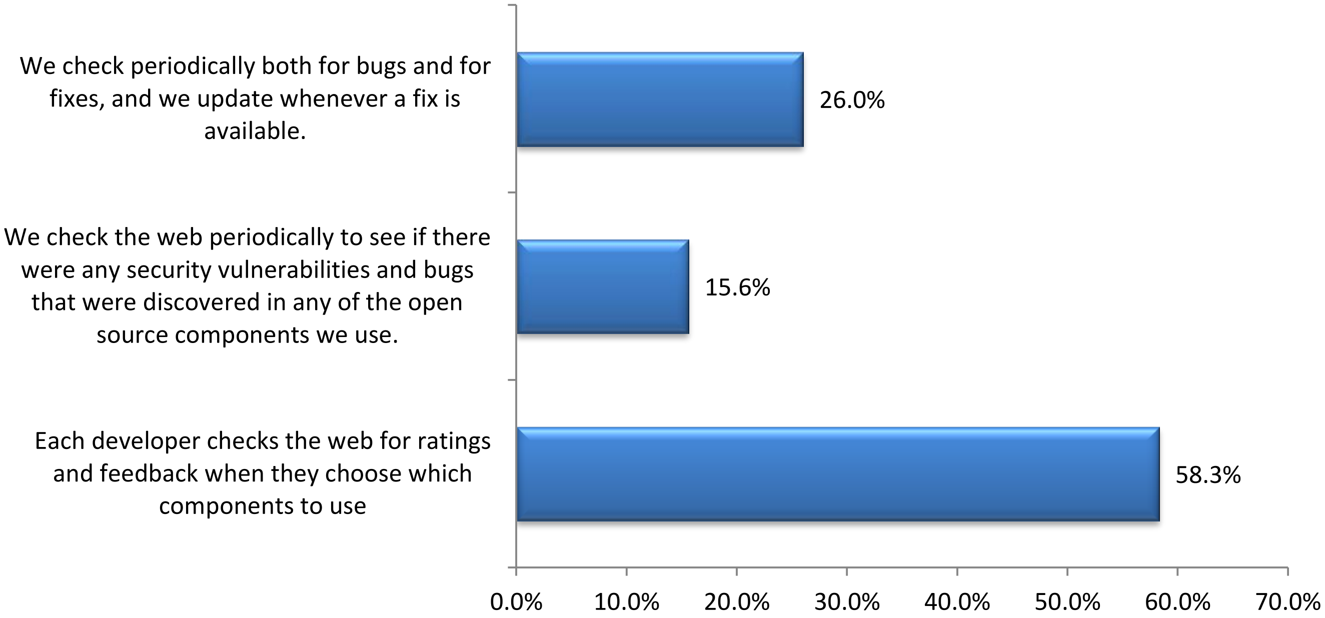
Do you have governance processes to enforce a license policy?



Our Analysis:
By now, it should not be surprising that most companies do not enforce licensing policies. After all, it doesn't make sense to enforce a policy when you don't have up-to-date information, or when the information is stale most of the time. Indeed 44% of companies do not have any policies, and 31% have a simple go/no-go policy which is based primarily on what developers have heard is the "right way" to handle open source. Many developers are aware of the potential issues with GPL-licensed projects, and they simply avoid those without much additional deliberation and without considering the risks with other open source licenses.

Only 25% of companies consulted with legal experts to develop a policy that is suitable for their organization. But what good is a policy if it cannot be systematically enforced, and with the right timing? Very few companies (9.4%) have a policy *and* use tools to automatically enforce it. (Again, most responders here are WhiteSource customers).

Do you check quality/security of open source components you use?

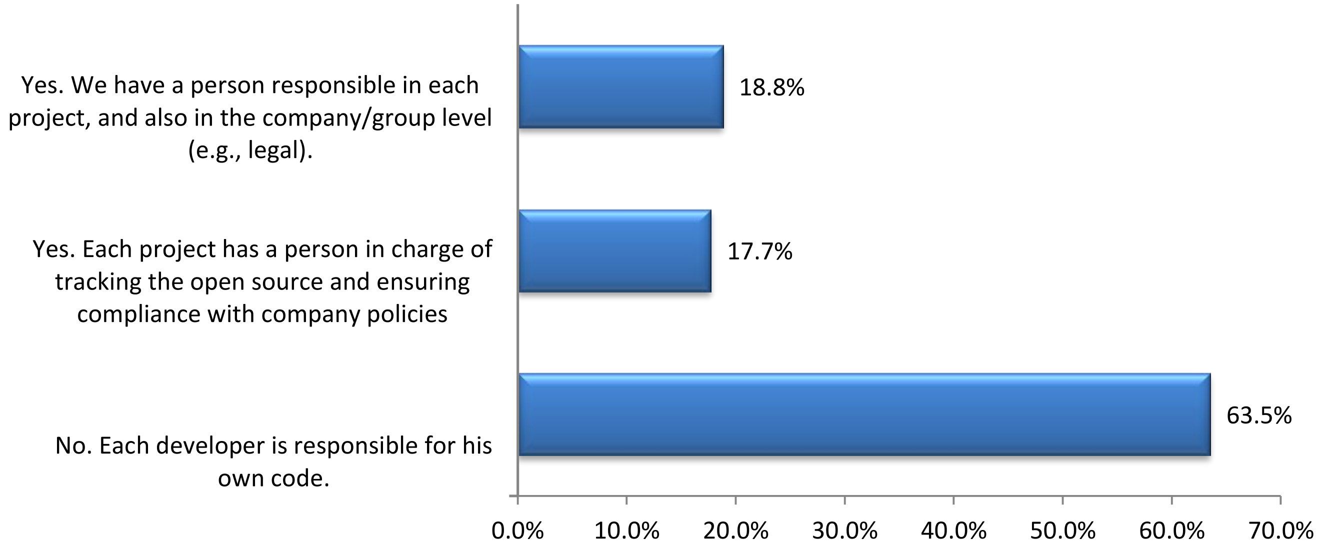


Our Analysis:
Since open source is code like any other, occasionally, defects and security vulnerabilities are discovered within it. However, while most companies test their own code, few have such processes for third-party code, including open source.

Many companies (58%) leave that job to developers. While developers likely check for quality and security reputation before bringing in an open source library, it is extremely rare for them to continue to monitor for new defects and vulnerabilities from that point onward. As a result, it is often the case that later vulnerabilities will be missed, and will not be updated, even if a patch were actually produced by the open source community. In recent WhiteSource research covering 3000 real projects, 24% of projects were found to contain a vulnerable library, even though in 98% of cases a fix had already been available for some time.

Clearly, having developers constantly on the watch for future defects and vulnerabilities is not reasonable. But with an open source management solution like WhiteSource, which is always fully aware of the entire and precise open source content in each of your products, it is only natural to expect that it would automatically alert you in case such issues ever arise, and especially so if a fix is available in a newer version.

Do you have a designated person in charge of open source management?



Our Analysis:
Last but not least is the question of who should be responsible for properly managing open source inventories.

As already mentioned, in most companies it is no one's responsibility, though it is implicitly expected of every developer or team leader.

In some companies (18%), open source is managed within the context of a specific product/project team. There clearly are advantages for doing so, as the requirements and limitations may depend on the specific characteristics of each product. However, as discussed earlier, developers, and even team leaders, lack the knowledge and also the appetite for dealing with legal and policy issues; to a great extent, it is also wasteful of their time.

In larger companies, it becomes necessary to have a uniform policy with regard to open source adoption (maybe adjusted for specific project needs). It also makes sense to leverage some of the information and expertise across projects, e.g., in case a certain library was already reviewed and accepted/rejected by another team.

Conclusion

It is almost impossible today to think of a software project that does not extensively use open source components. However, most companies' usage is completely out of control, for they severely undermanage their open source usage.

Most companies still rely on unorganized manual and laborious processes, often performed by individual developers or by a designated person on a development team. The result is a greater than necessary, but often hidden, cost in (1) developer time; (2) delays to release schedules; and (3) significant legal risks, due to lack of knowledge and inconsistent execution. Security risks are even greater since few developers continuously track the relevant sources to identify new security vulnerabilities that are discovered in the open source they use.

A few companies have adopted source code scanning technologies. However, beyond the high costs of acquiring and deploying such solutions, they require much manual effort in reviewing thousands of false positive "potential matches." As a result, most companies use them once every few months, usually just before a release, or as part of a due-diligence process. This leaves the company exposed between scans, and may require expensive replacements in case rogue open source is discovered in the scan.

New technologies such as WhiteSource make it easy to continuously track open source usage, and automatically enforce licensing and security policies. WhiteSource plugs into the build server and becomes a native part of the software development lifecycle without burdening developers. New open source modules are discovered as soon as they are added by developers. Their licenses (and those of all of their dependencies) are automatically compared to the company licensing policies, initiating the appropriate approve/reject workflow if necessary. WhiteSource continues to track each open source in use, and will proactively notify each project manager in case of new vulnerabilities or patches.

About WhiteSource

WhiteSource provides easy-to-use solutions for managing the usage of open source components by developers, to ensure license compliance and reduce security and quality risks.

WhiteSource easily plugs itself into the software development lifecycle, and automatically detects new open source components as soon as they are entered by developers. Thereafter, WhiteSource continuously provides (1) comprehensive and up-to-date open source inventory reports (down to the last dependency); (2) license risks analysis and compliance reports; and (3) proactive alerts on security vulnerabilities whenever discovered, as well as available fixes.

WhiteSource is easy to setup, requires no training to use, and completely removes the burden from developers. The service is affordable to companies of all sizes.

For more information, please visit: http://www.WhiteSourceSoftware.com

Or, contact Patricia.Johnson@WhiteSourceSoftware.com