DSI Industrial & Policy Recommendations (IPR) Series

Security record of open source and free software

Martin Schallbruch (Digital Society Institute, ESMT Berlin)

Issue 5, 2017

In April 2017, the Digital Society Institute hosted a workshop entitled "How secure is free software? Security record of open source and free software." The workshop included contributions from Matthias Kirschner (Free Software Foundation Europe), Kathrin Noack (Karlsruhe Institute of Technology, Projekt secUnity), Michael Kranawetter (Microsoft) and Carl-Daniel Hailfinger (German Federal Office for Information Security).

1. Status

Many IT decision makers are convinced that free and open source software offers security advantages over proprietary software. The overwhelming number of cyber attacks are conducted by exploiting vulnerabilities in proprietary software. On the other hand, serious security incidents such as Heartbleed and Shellshock have shown that there are specific security issues with free software.

Three factors are relevant to assessments of the security characteristics of free software – even in comparison with proprietary software: first, the quality of development; second, life-cycle management, in particular the patching of security vulnerabilities; and third, the security assessments of somewhat independent third-party security researchers.

Systematic studies of the number of vulnerabilities and the mean detection time in free and proprietary software by IT security companies (e.g., Veracode 2016) and scientists (e.g., Schryen 2011, 130-140) do not show much difference. The Coverity report (Synopsys 2015) finds a slightly lower defect density of free software versus proprietary software; in the area of Web applications, however, the exact opposite result was found, which is essentially due to the security weakness of PHP.

Various studies have shown, however, that the vulnerability frequency and vulnerability detection time depend upon the size and complexity of the software. Particularly significant differences are also observed in the patch behavior of the manufacturers or communities. Free software is characterized by the fact that its use, analysis, distribution, and improvement is, in principle, free of licensing, regardless of the specific licensing models. In addition, there are significant differences between different free software products, such as the scope of license protection or the developer's motivation. For example, a large number of employed developers are involved in the further development of free software on behalf of their employers. It is also important to note that free software is increasingly being provided in virtualized form by commercial providers, for example in the form of Linux cloud instances by Microsoft or AWS.

Large differences between various free software products also result from the respective community governance. While both the Linux kernel community and the Apache Software Foundation basically work according to the principle of a "meritocracy," the internal structures are different. The Linux kernel community is very focused on Linus Torvalds, while the Apache Software Foundation is based on a "constitution" with a chosen board and collaborative delegation of responsibility. Software quality and thus also the prevention of security flaws is an essential theme of the communities. Investigations into the extent to which different community governance models influence the

development quality of the software on the one hand and the prevention of malicious exploitation of security flaws on the other are not yet available.

A central question for the security of free software is the transparency of the development stages. In particular, it is sometimes difficult to identify the main development branch of a product, because there are many forks. An important issue for the safety of free software products is also the relationship between the relevance of a product and the stability of the developer community. In the case of heavily used, but little known, software with small developer communities, there is a risk that the quality of the further development can no longer be adequately ensured.

A decisive advantage of free software is the verifiability of the code by independent security researchers, who also have the opportunity to create patches for vulnerabilities. These possibilities for review are clearly superior to the source code view offered by proprietary software vendors regarding access to code, tool use in analysis, and legal risks when publishing vulnerabilities.

Another advantage of free software is the non-existent "bunker mentality" in the detection of vulnerabilities. For proprietary software vendors, the presence of vulnerabilities, possibly difficult to patch, may have significant economic implications, so it cannot be ruled out that the handling of the vulnerability is affected by extraneous motives and the closing of the vulnerability is delayed or not done.

However, free software can only realize its advantage if a "producer-like" community exists for the respective product, which is transparent, quality-oriented, stable, and sufficiently financed over a long period of time.

2. Recommendations

Recommendations for companies and authorities

IT enterprises

Complexity is a key driver of uncertainty. Among the key development guidelines for IT companies should be to ensure the appropriateness of code scope and code quality and to avoid incorporating unneeded functionality and modules. As IT companies integrate free software modules into their products, they should be concerned with the security features of these modules and set up their own management process, which continuously processes this. In particular, care should be taken to determine whether the used modules continue to correspond to the main development branch, whether the module is actively developed and used by others, whether the development community of the module is adequately equipped and financed, and how the post-breach behavior develops.

If the modules are critical for important functions of your own product, you should build up your own developer capacity, which is actively involved in the respective community.

User companies and authorities

Security advantages of using free software can be used by user companies and authorities if there is sufficient assessment of the security features of the respective product. In particular, the expertise needs to be built or bought in order to assess the stability, quality orientation, and post-breach behavior of a community. Long-term and intensive observation of the development of the product and the community as well as the handling of vulnerabilities is advisable if the product is used in critical fields.

Policy recommendations

Free software has a significant impact on the software market and is even dominant in some market segments. As a result of the integration of free software modules into proprietary software, free software is critical to the security of information technology. The aim of policy should be to help developer communities and users of free software to improve the framework conditions for the security of free software.

Promoting transparency and verifiability

The consideration of security goals in community governance has not yet been adequately explored. This also applies to the implementation of security life cycle processes for free software. By encouraging appropriate research, transparency about security features during development and further development of free software can be improved.

At the same time, policy can promote the testing of significant free software products by independent security researchers. This is particularly advisable if such software systems are used in especially relevant fields of application, such as state authorities or critical infrastructures.

Securing critical components

In some cases, free software products take on critical functions for the economy and society (e.g., OpenSSL). Such products should be identified and analyzed in their criticality. Policy, together with companies, should then provide the framework for sustainable and resilient development and ensure that security features are regularly and independently examined.

3. References

Schryen (2011). Is Open Source Security a Myth? Communications of the ACM 54 (5), New York, NY, USA: ACM.

Synopsys (2015). Coverity Scan Open Source Report 2014. http://go.coverity.com/rs/157-LQW-289/images/2014-Coverity-Scan-Report.pdf

Veracode (2016). The State of Software Security. Volume 7. https://www.veracode.com/resources/state-of-software-security.

The DSI Industrial & Policy Recommendations (IPR) Series is published by the Digital Society Institute of ESMT Berlin, http://dsi.esmt.org.

DSI Industrial & Policy Recommendations (IPR) Series

Sicherheitsbilanz von Open-Source- und freier Software

Martin Schallbruch (Digital Society Institute, ESMT Berlin)

Ausgabe 5, 2017

Im April 2017 war das Digital Society Institute Gastgeber für einen Workshop „Wie sicher ist freie Software? Sicherheitsbilanz von Open-Source- und freier Software“. Impulse zu dem Workshop steuerten Matthias Kirschner (Free Software Foundation Europe), Kathrin Noack (Karlsruher Institut für Technologie, Projekt secUnity), Michael Kranawetter (Microsoft) und Carl-Daniel Hailfinger (Bundesamt für Sicherheit in der Informationstechnik) bei.

1. Sachstand

Viele IT-Entscheider sind davon überzeugt, dass freie und Open-Source-Software Sicherheitsvorteile gegenüber proprietärer Software bietet. Die überwiegende Zahl aller Cyberangriffe erfolgt durch das Ausnutzen von Schwachstellen in proprietärer Software. Andererseits haben schwerwiegende Sicherheitsvorfälle wie zum Beispiel Heartbleed und Shellshock gezeigt, dass es bei freier Software spezifische Sicherheitsfragen gibt.

Für die Bewertbarkeit der Sicherheitseigenschaften von freier Software – auch im Vergleich zu proprietärer Software – sind insgesamt drei Faktoren relevant: Erstens die Qualität der Entwicklung, zweitens das Life-Cycle-Management, insbesondere das Verfahren der Schließung von Sicherheitslücken, und drittens die Bewertbarkeit der Sicherheit durch Dritte, etwa unabhängige Sicherheitsforscher.

Systematische Untersuchungen der Anzahl von Schwachstellen und der mittleren Entdeckungszeit in freier und proprietärer Software durch IT-Sicherheitsunternehmen (z.B. Veracode 2016) und Wissenschaftler (z.B. Schryen 2011, 130-140) ergeben keine großen Unterschiede. Der Coverity Report (Synopsis 2015) hat eine leicht geringere Defektdichte freier Software gegenüber proprietärer Software aufgezeigt; im Bereich der Webanwendungen wurde jedoch das genau umgekehrte Ergebnis gefunden, was im Wesentlichen auf der Sicherheitsschwäche von PHP beruht.

Aufgezeigt werden in verschiedenen Studien jedoch Abhängigkeiten der Schwachstellenhäufigkeit und -entdeckungszeit von der Größe und Komplexität der Software. Besonders signifikante Unterschiede sind auch bei dem Patch-Verhalten der Hersteller beziehungsweise Communities zu verzeichnen. Freie Software zeichnet sich dadurch aus, dass Ihre Verwendung, Analyse, Verbreitung und Verbesserung unabhängig von den konkreten Lizenzmodellen grundsätzlich genehmigungsfrei ist. Darüber hinaus gibt es erhebliche Unterschiede zwischen verschiedenen freien Softwareprodukten, etwa im Hinblick auf die Reichweite des Lizenzschutzes oder die Motivlagen der Entwickler. Beispielsweise wirkt eine Vielzahl angestellter Entwickler an der Weiterentwicklung freier Software im Auftrag ihres Arbeitgebers mit. Zu beachten ist auch, dass freie Software zunehmend in virtualisierter Form von kommerziellen Anbietern bereitgestellt wird, etwa in Form von Linux-Cloud-Instanzen durch Microsoft oder AWS.

Große Unterschiede zwischen verschiedenen freien Softwareprodukten ergeben sich auch aus der jeweiligen Community Governance. Während sowohl die Linux Kernel Community als auch die Apache Software Foundation grundsätzlich nach dem Prinzip

einer „Meritokratie“ arbeiten, sind die internen Strukturen unterschiedlich. Bei der Linux Kernel Community erfolgt eine Zuspitzung auf Linus Torvalds, während der Apache Software Foundation eine „Verfassung“ zu Grunde liegt mit gewähltem Board und arbeitsteiliger Verantwortungsübertragung. Softwarequalität und damit auch die Vermeidung von Sicherheitsmängeln ist ein wesentliches Thema der Communities. Untersuchungen, inwieweit unterschiedliche Community Governance-Modelle die Entwicklungsqualität der Software einerseits und die Verhinderung des böswilligen Einpflegens von Sicherheitslücken andererseits beeinflussen, liegen bislang nicht vor.

Eine zentrale Fragestellung für die Überprüfbarkeit der Sicherheit freier Software ist die Transparenz der Entwicklungsstände. Insbesondere ist es bisweilen schwierig, den „Hauptentwicklungszweig“ des Produkts zu erkennen, weil sehr viele Forks existieren. Eine bedeutende Fragestellung für die Sicherheit freier Softwareprodukte ist zudem das Verhältnis zwischen der Relevanz eines Produkts und der Stabilität der Entwicklergemeinschaft. Bei stark genutzter, aber wenig bekannter Software mit kleinen Entwicklergemeinschaften besteht die Gefahr, dass die Qualität der Weiterentwicklung nicht mehr ausreichend sichergestellt werden kann.

Ein entscheidender Vorzug freier Software ist die Überprüfbarkeit des Codes durch unabhängige Sicherheitsforscher, die auch gleich die Möglichkeit haben, Patches für Schwachstellen zu erstellen. Diese Möglichkeiten der Überprüfung sind hinsichtlich des Zugangs zum Code, des Tooleinsatzes bei der Analyse und der rechtlichen Risiken bei der Veröffentlichung von Schwachstellen den von Herstellern proprietärer Software bereitgestellten Angeboten zur Source-Code-Einsicht deutlich überlegen.

Ein weiterer Vorteil freier Software ist die nicht vorhandene „Bunker-Mentalität“ bei der Entdeckung von Schwachstellen. Für Hersteller proprietärer Software kann das Vorhandensein von – möglicherweise schwer zu patchenden – Schwachstellen erhebliche wirtschaftliche Auswirkungen haben, weshalb nicht auszuschließen ist, dass der Umgang mit der Schwachstelle von sachfremden Motivlagen beeinflusst ist und das Schließen der Schwachstelle zeitverzögert oder nicht erfolgt.

Allerdings kann freie Software ihren Vorteil nur dann ausspielen, wenn für das jeweilige Produkt eine „herstellerähnliche“ Gemeinschaft existiert, die transparent, qualitätsorientiert, über längere Zeit stabil und ausreichend finanziert ist.

2. Empfehlungen

Empfehlungen für Unternehmen und Behörden

IT-Unternehmen

Komplexität ist zentraler Treiber von Unsicherheit. Eine der wesentlichen Entwicklungsleitlinien für IT-Unternehmen sollte es sein, Angemessenheit des Code-Umfangs und der Code-Qualität sicherzustellen und das Einbinden nicht benötigter Funktionalitäten und Module zu vermeiden. Soweit IT-Unternehmen freie Softwaremodule in ihre Produkte integrieren, sollten sie sich mit den Sicherheitseigenschaften dieser Module beschäftigen und einen eigenen Management-Prozess aufsetzen, der dies fortlaufend bearbeitet. Hierbei ist insbesondere darauf zu achten, ob die selbst verwendeten Module weiterhin dem Hauptentwicklungszweig entsprechen, ob das Modul aktiv entwickelt und von anderen verwendet wird, ob die Entwicklungsgemeinschaft des Moduls angemessen ausgestattet und finanziert ist und wie sich das Post-Breach-Verhalten entwickelt.

Sofern die Module für wichtige Funktionen des eigenen Produkts kritisch sind, sollte eigene Entwicklerkapazität aufgebaut werden, die sich an der jeweiligen Community aktiv beteiligt.

Anwenderunternehmen und Behörden

Sicherheitsvorteile des Einsatzes freier Software können von Anwenderunternehmen und Behörden genutzt werden, wenn ausreichende Beurteilungsfähigkeit zu den Sicherheitseigenschaften des jeweiligen Produktes vorhanden ist. Insbesondere muss die Expertise aufgebaut oder eingekauft werden, um die Stabilität, Qualitätsorientierung und das Post-Breach-Verhalten einer Community beurteilen zu können. Eine dauerhafte und intensive Beobachtung der Entwicklung des Produkts und der Community sowie des Umgangs mit Schwachstellen ist anzuraten, sofern das Produkt in kritischen Feldern zum Einsatz kommt.

Empfehlungen für die Politik

Freie Software hat eine erhebliche Bedeutung für den Softwaremarkt und ist in manchen Marktsegmenten sogar marktbeherrschend. Dadurch aber vor allem auch durch die Integration freier Softwaremodule in proprietäre Software hat freie Software erhebliche Bedeutung für die Sicherheit der Informationstechnik. Ziel der Politik sollte es sein, den Entwicklergemeinschaften und den Anwendern freier Software zu helfen, die Rahmenbedingungen für die Sicherheit freier Software zu verbessern.

Transparenz und Überprüfbarkeit fördern

Die Berücksichtigung von Sicherheitszielen in der Community Governance ist noch unzureichend erforscht. Dies gilt auch für die Implementierung von Security-Life-Cycle-Prozessen bei freier Software. Durch die Förderung entsprechender Forschung kann die Herstellung von Transparenz über die Sicherheitseigenschaften bei Entwicklung und Weiterentwicklung freier Software verbessert werden.

Gleichzeitig kann die Politik die Prüfung bedeutsamer freier Softwareprodukte durch unabhängige Sicherheitsforscher fördern. Dies ist insbesondere dann anzuraten, wenn entsprechende Softwaresysteme in besonders relevanten Anwendungsfeldern zum Einsatz kommen, etwa staatlichen Behörden oder kritischen Infrastrukturen.

Kritische Komponenten sichern

Freie Softwareprodukte übernehmen in manchen Fällen kritische Funktionen für Wirtschaft und Gesellschaft (z.B. OpenSSL). Solche Produkte sollten identifiziert und in ihrer Kritikalität analysiert werden. Anschließend sollte die Politik gemeinsam mit Unternehmen den Rahmen dafür schaffen, dass eine dauerhafte und belastbare Weiterentwicklung sichergestellt und die Sicherheitseigenschaften regelmäßig unabhängig untersucht werden.

3. Referenzen

Schryen (2011). Is Open Source Security a Myth? Communications of the ACM 54 (5), New York, NY, USA: ACM.

Synopsys (2015). Coverity Scan Open Source Report 2014. http://go.coverity.com/rs/157-LQW-289/images/2014-Coverity-Scan-Report.pdf

Veracode (2016). The State of Software Security. Volume 7. https://www.veracode.com/resources/state-of-software-security.

Die DSI Industrial & Policy Recommendations (IPR) Series wird herausgegeben vom Digital Society Institute der ESMT Berlin, http://dsi.esmt.org.