heise online

Studie: Proprietäre Software kann nicht sicherer sein als Open-Source

25.07.2023 18:42 Uhr Stefan Krempl

(Bild: Imilian/Shutterstock.com)

Das Entwicklungsmodell an sich erlaube keine Aussage über die Sicherheit, lautet die Analyse. Bei Open Source sei deren Implementierung aber für jeden prüfbar.

Beim Beantworten der Frage, ob **proprietäre oder freie und quelloffene Software bei der IT-Sicherheit die Nase vorn hat [1]**, geht es **oft pauschal zu [2]**. "Viele Argumente für oder gegen die Sicherheit eines bestimmten Entwicklungsansatzes scheinen eher Bauchgefühle, Vorurteile oder Meinungen zu sein", weiß der Bonner Informatiker Marc Ohm. Im Auftrag der Open Source Business (OSB) Alliance hat er daher untersucht, welches Lager recht haben könnte. Sein Fazit lautet: "Das Entwicklungsmodell an sich erlaubt keine Aussage über die Sicherheit." Bei Open-Source-Software sei die Implementierung der Verfahren jedoch zumindest "für jeden prüfbar".

Prinzipiell werde Software "immer mehr zu einem kritischen Punkt, ohne den der gesamte Betrieb zum Stillstand kommt", schreibt Ohm, der über Angriffe auf Software-Lieferketten promoviert hat, **auf den knapp 50 Seiten [3]**. Die Security von Programmen werde so zu einem nicht mehr vernachlässigbaren Aspekt. "Open-Source-Projekte werden in der Regel von mehreren Entwicklern betreut und gepflegt", arbeitet der Forscher heraus. Hauptansprechpartner für Support sei hier meist "die Community, die sich aus Anwendern und Entwicklern zusammensetzt. Da zudem jeder Nutzer

Verbesserungen einreichen kann, erhält die Software regelmäßige Updates und vor allem Sicherheitspatches."

Zuverlässiger mit Hersteller oder Stiftung

Die "reine Möglichkeit zu Verbesserungen" lässt Ohm zufolge aber "nicht direkt auf eine erhöhte Qualität oder Sicherheit schließen". Auch freie Software müsse "zusätzlich aktiv betreut werden". Am zuverlässigsten geschehe dies, "wenn hinter dem Projekt ein Hersteller oder eine Stiftung steht". Ein klarer Vorteil bei quelloffenen Programmen sei aber, dass sie leichter und etwa auch von spezialisierten Stellen "nach Sicherheitsvorgaben zertifiziert werden" könnten. Werde die Open-Source-Software zudem von einem Unternehmen bezogen oder durch einen Dienstleister entwickelt oder gewartet, gälten zudem die kaufrechtlichen Bestimmungen, einschließlich der Gewährleistung wie Nachbesserungspflicht und Haftung.

"Die Sicherheit von proprietärer Software hängt allein vom Hersteller ab", legt der Autor dar. "Anwender müssen warten, bis ein entsprechender Patch zur Verfügung gestellt wird." Falls vorhanden, könne ein dediziertes Entwicklerteam solche Probleme gezielt angehen und lösen. Dies sei von außen aber nicht ersichtlich. Auch bei Open Source könnten durch ein spezielles Team oder das gemeinsame Engagement der Community Programmierfehler und Schwachstellen gefunden und behoben werden. Daher bewerte das Bundesamt für Sicherheit in der Informationstechnik (BSI) solche Programme "als mindestens genauso sicher wie proprietäre Software".

Die Basis für proprietäre Programme

"Ohne Einblick in den Quelltext kann die Sicherheit einer Software niemals sichergestellt werden", führt Ohm aus. Selbst Code-Reviews schützten in beiden Bereichen aber nicht vollumfänglich etwa vor Sabotage. Typischerweise bilde Open-Source-Software jedoch mittlerweile die Grundlage für proprietäre Programme, da durch erstere Basisfunktionen effektiv und kostengünstig bereitgestellt werden könnten. Typischerweise setzten Firmen so auf Open Source auf und fügten dieser einen "marktdifferenzierenden Teil als proprietäre Komponente" hinzu. Ein Beispiel dafür sei der freie Browser Chromium, auf dem Google Chrome und Microsoft Edge basierten.

Ohm zieht daraus den Schluss, dass angesichts der weitgehenden Durchdringung "proprietäre Software nicht sicherer als Open-Source-Software sein kann". Eine Trennung beziehungsweise Unterscheidung beider Ansätze scheine gar nicht mehr sinnvoll. Es müssten gemeinsame Maßstäbe angelegt werden, um die Vertrauenswürdigkeit eines Projekts zu bestimmen. Gefragt seien möglichst konkrete Messgrößen, die sich explizit auf die Sicherheit des Entwicklungsprozesses und auf die allgemeine Qualität einer Softwareinitiative bezögen.

Community-Projekt plus Hersteller als Königsweg

Entsprechende Metriken zur Auswahl von qualitativ hochwertiger und sicherer Software stellt der Wissenschaftler vor. Als abstraktes Rahmenwerk verweist er etwa auf das **Secure Software Development Framework [4]** (SSDF) des National Institute of Standards and Technology (NIST). Ferner gebe es Empfehlungen für die Lieferkette in Form spezieller Supply Chain Levels for Software Artifacts (**SLSA [5]**), **für Entwickler [6]** sowie Instrumente und Standards. Dazu gehörten

Sicherheitstests mithilfe von Lintern, Static Application Security Testing (SAST), **Software Component beziehungsweise Composition Analysis [7]** (SCA), Dynamic Application Security Testing (DAST), Fuzzern, Detektoren für hartcodierte Geheimnisse und Generatoren für Inventare der verwendeten Komponenten.

Als Königsweg empfiehlt der Experte den Einsatz kommerzieller Open-Source-Software, die auf der Basis von Community-Projekten von Unternehmen vorangetrieben, unterstützt und rechtssicher vertrieben wird. Stehe eine Firma hinter einer Entwicklung, könne sie zusätzlichen positiven Einfluss nehmen und Mehrwert generieren. Aber selbst die Einhaltung aller bewährten Praktiken garantiere keine vollkommen sichere Software. Es bleibe immer ein Restrisiko, das nur bei Open Source unabhängig messbar sei.

"Quelloffenheit und kollaborative Entwicklungsmodelle helfen, Software sicherer zu machen", begrüßt Elmar Geese, Sprecher der Arbeitsgruppe Security in der OSB Alliance, die Ergebnisse. "Das alleine reicht natürlich nicht aus. Mit unserer Studie wollen wir Impulse setzen, diese Lücke zu schließen. Besonders die kommerziellen Verwerter von Software sind hier in der Pflicht."

(**axk [8]**)

URL dieses Artikels:

https://www.heise.de/-9226451

Links in diesem Artikel:

[1] https://www.heise.de/tipps-tricks/Ist-Open-Source-Software-wirklich-sicherer-3929357.html

[2] https://www.heise.de/news/Umfrage-IT-Entscheider-halten-Open-Source-Software-fuer-sicherer-5072472.html

[3] https://osb-alliance.de/wp-content/uploads/2023/01/Studie-zum-Vergleich-der-Sicherheit-von-Open-Source-Software-und-proprietaerer-Software.pdf

[4] https://csrc.nist.gov/Projects/ssdf

[5] https://slsa.dev

[6] https://github.com/ossf/wg-best-practices-os-developers

[7] https://www.heise.de/hintergrund/Software-Composition-Analysis-Tools-in-der-Marktuebersicht-7543879.html

[8] mailto:axk@heise.de