// diconium

Open-Source-Software und Quellcode aus dem Internet rechtssicher einbinden und nutzen

XP Days 2022, 7. Oktober 2022, 10.55 Uhr

Dr. Falk W. Müller, RA, FAITR

Über Falk

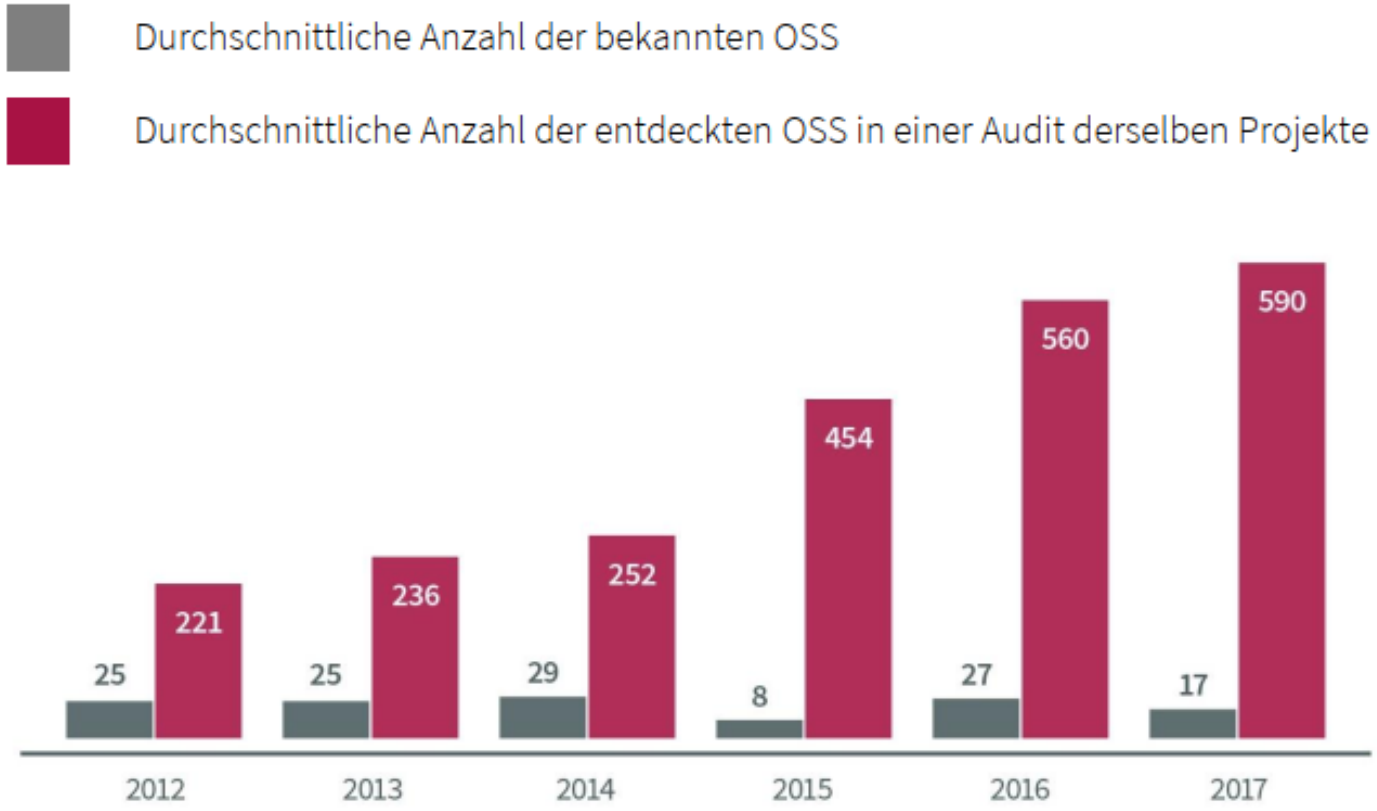
- 39 Jahre alt, 1 Sohn
- Jura-Studium, Fachrichtung „Rechtsinformatik“
  2002 – 2007, später jur. Vorbereitungsdienst
- Softwareentwickler seit 1997
- Bundesweit aktiver Rechtsanwalt,
  Fachanwalt für IT-Recht –
  v. a. **Scrum/Agil**, **Open Source**, Urheberrecht, **Datenschutz**
- Regelmäßiger Speaker auf Konferenzen, vor Unternehmen,
  auf Fachanwaltslehrgängen
- Lehrbeauftragter der Hochschule für Polizei Baden-Württemberg

Einführung und Praxisrelevanz

Wer soll mich schon verklagen?

- Risiko in der Tat zwar überschaubar, aber tückisch, denn:
  - Wenn es dazu kommt, sehr schnelle, oft überstürzte Einführung umfassender Prozesse nötig (wegen Unterlassungsverpflichtungen)
  - Haftungsfragen
  - Habe ich wirklich alles so lizenziert, wie es erforderlich wäre?
- Der Fall Patrick McHardy

Wie viel Open Source nutze ich denn schon auch?



Vom Werk
zum Urheberrecht
zur Lizenz

Aus Sicht des Open Source-Entwicklers: Wann entsteht meine „Open Source-Software“?

- *„Ich will eine Bibliothek entwerfen, die beliebige Fahrraddaten wie Geschwindigkeit, Trittfrequenz, … an ein Sportportal senden kann“*
  - → Ich möchte meinen Code Open Source machen, damit andere ihn nutzen und ggf. verbessern können

Typische Fragestellungen:

- Habe ich da ein Urheberrecht drauf? Wie entsteht es?
- Bin ich irgendwie haftbar, komme ich für irgendwas in den Knast?
- Was kann ich tun, um meine Software unabhängig zu halten?

Urheberrecht im Stil von „Der Sendung mit der Maus“

§ 1 Urheberrechtsgesetz (UrhG):

„Die Urheber von Werken der Literatur, Wissenschaft und Kunst genießen für ihre Werke Schutz nach Maßgabe dieses Gesetzes.“

Geistiges Gut ist keine Sache i. S. d. Gesetzes und daher nicht über das Sachenrecht geschützt.

Aber: Rechtsordnung belohnt die schöpferische Tätigkeit, die ein neues geistiges Gut hervorbringt, indem sie ein eigentumsähnliches Recht an diesem Gut gewährt.

- Schöpferprinzip: Urheber ist der Schöpfer als natürliche Person, gilt auch im Arbeitsverhältnis
  - © mein Arbeitgeber?
    - → Im Arbeitsverhältnis üblicherweise Regelung im Arbeitsvertrag mit Übertragung aller denkbarer Urheberrechte exklusiv an Arbeitgeber

Entstehung

- Erreichen der Schöpfungshöhe: Die „Untergrenze des Urheberrechtsschutzes“
  - Klassischerweise niedrig anzusetzen („kleine Münze“)
- Wenn mehrere Personen gemeinsam ein Werk schaffen: **Miturheberschaft**
- Urheberrechtsschutz entsteht direkt mit dem Schöpfungsakt unabhängig von einer Veröffentlichung und vom Erscheinen
- Urheberschaft wird durch Kennzeichnung auf dem Werk vermutet (z.B. ©),
  - Hat historische Gründe, ist nicht erforderlich

Schutzfähig

- **Alle Formen** von Computerprogrammen sind geschützt (vor allem geschriebener und binärer Code), jedoch nur wenn das Programm tatsächlich **von Menschen** (bzw. aufgrund von menschlichen Anweisungen) geschrieben ist.
  - Umstritten bei HTML-Code und SQL-Statements: eher nicht schutzfähig
- Geschützt werden können grundsätzlich auch **Benutzeroberflächen** bzw. **Datenbanken (Datensammlungen)**.

- Schutzrechtsinhaber hat das **ausschließliche Recht** zur Verbreitung, Vervielfältigung und Bearbeitung des Programms.
- Nutzungs- und Verwertungsrechte, z. B.
  - Bearbeitungsrecht,
  - Vervielfältigungsrecht,
  - Verbreitungsrecht,
  - Recht der öffentlichen Zugänglichmachung.
- Einfache oder exklusive Nutzungsrechte

Lizenzen und Lizenzrecht

- Rechte welcher Art auch immer werden erteilt über sog. Lizenzen
- Lizenzrecht ist das, was der Lizenznehmer tun darf
  - → Wird geregelt im Lizenzvertrag

Haftung ist eine Frage des anwendbaren Rechts

- Nach deutschem Recht Open Source / Freeware eine sog. Schenkung, § 516 BGB
  - Schenkende(r) haftet grundsätzlich nur in einem Fall:
    „Verschweigt der Schenker arglistig einen Fehler der verschenkten Sache, so ist er verpflichtet, dem Beschenkten den daraus entstehenden Schaden zu ersetzen.“
    (§ 524 Abs. 1 BGB)
  - Sonst (grundsätzlich) nicht – gute Stellung des Schenkenden.

- US-amerikanisches Recht: „Jeder ist seines Glückes Schmied“
  - → ganz viele Freiheiten, aber zugleich auch schwer vorhersehbar

im Gegensatz zu

- Deutsches (und auch europäisches) Recht: „Der Bürger ist doof. Es ist daher alles ganz genau zu regeln und jeder zu schützen. Und zwar vor allem vor der bösen Wirtschaft.“
  - → Freiheiten deutlich eingeschränkt, aber Rechtslage leichter bestimmbar (auch wenn Gerichte immer wieder recht kreativ sind)

Und daher…

- Nach US-amerikanischem Recht einige Kreativität (und halt auch Rechtskenntnis…) gefordert, weswegen sich viele Entwickler für vorgefertigte Lizenztexte entscheiden
  - → So kommt es zum Entwurf und massiven Einsatz von Open Source-Lizenzen

Und daher…

- Natürlich nutzen auch deutsche Entwickler vorgefertigte Lizenztexte
  - Greifen aber durch AGB-Recht nur bedingt:
    - Definitionen dienen als Anhaltspunkte für Vertragsauslegung
    - Haftungsausschlüsse greifen an sich nicht, aber Schenkung hat ohnehin kaum eine
    - Lizenzrechte werden in deutsches Recht übertragen und greifen

Und dann war da noch: Der gutgläubige Erwerb

- Ich verkaufe (ohne Berechtigung) das Notebook meiner Freundin F sehr günstig an Käufer K. K freut sich und möchte das Notebook haben und auch behalten.

  F möchte das Notebook aber auch oder zumindest Geldersatz.

  An wen wendet sie sich?

- Wie ist es bei Lizenzen?

Open Source-Lizenzen

Überblick

- Die bekanntesten Open Source-Lizenzen sind
  - GPL (Linux-Kernel, an sich ganz viel um Linux herum)
    - LGPL, Affero GPL, GPL mit Appendizes
  - BSD
    - Apache License v2.0 (Android, Apache, OpenOffice)
    - MIT (jQuery, X Window)
  - Mozilla Public License MPL (Netscape -> Firefox)
    - Common Development and Distribution License (CDDL) (gerne als Alternativlizenz zu GPL)
  - Eclipse Public License (EPL)

Kurze Übersicht über Lizenzen

- BSD, MIT und Apache recht ähnlich (de facto von BSD abgeleitet), so ungefähr „Mach mit meiner Software was du willst – aber verklag mich nicht“

- GPL und Derivate zusätzlich: „Ich habe meine Software Open Source gemacht – und du solltest das auch tun. Oder mir zumindest die Freiheit lassen, mit meiner Software zu machen, was ich möchte.“

Gemeinsamkeiten aller Lizenzen

- Beschreibung der Lizenzrechte
  - Üblicherweise Möglichkeit der beliebigen, auch kostenpflichtigen Weitergabe
  - Letztlich nur geknüpft an bestimmte Pflichten wie…
    - Beibehaltung etwaiger Copyright-Hinweise,
    - Weitergabe Lizenztexte,
    - Weitergabe Quelltext.

Gemeinsamkeiten aller Lizenzen

- Umfassende Haftungsfreistellungen in jedwede Richtung
  - Also sowohl für Sach- wie auch Rechtsmängel
    - Sachmangel: „Bugs“
    - Rechtsmangel: „Ich habe eine Bibliothek lizenzwidrig verwendet“
- Üblicherweise auch Wiedergabe in Lizenzbedingungen der diese einbindende Software, bspw. BSD: „Dieses Produkt enthält Software, die von der University of California, Berkeley und Beitragsleistenden entwickelt wurde.“

Unterscheidung somit durch Copyleft

Strenges Copyleft

- GPL
- AGPL
- IBM Public License

Eingeschränktes Copyleft

- LGPL
- CDDL
- MPL
- EPL

Kein Copyleft

- Apache License v2.0
- MIT
- BSD

Was ist dieses ominöse Copyleft?

- Veränderungen sind nur erlaubt, wenn sie unter den gleichen Lizenzbedingungen weitergegeben werden
- Sorgt somit dafür, dass Fortentwicklungen des freien Ur-Programms frei bleiben
- Copyleft-Lizenz fordert, dass auch die abgeleiteten Werke unter einer solchen Lizenz zur Verfügung gestellt werden

Drehen wir den Spieß um: Fallstricke für Unternehmen als Einsetzende von Open Source

- **Haftung und Haftungsfreistellung**
  - Damit haftet (für die eingesetzte OSS) letztlich jedes einsetzende Unternehmen, ggf. also der Kunde, für den man als Entwickler arbeitet
  - Ausgleich allenfalls im Innenverhältnis mit dem Entwickler
  - Sichert man als Entwickler seinem Kunden Rechtsmangelfreiheit zu? Das kann in die Hose gehen…
- **Unbekannte Lizenzen und deren Detailregelungen** (bspw. Copyleft in IBM Public License)

Drehen wir den Spieß um: Fallstricke für Unternehmen als Einsetzende von Open Source

- **Infizierung: Problem des sog. „Viralen Effekts“**
  - Werden Lizenzbedingungen nicht eingehalten, kann durch Copyleft-Effekt die gesamte proprietäre Software infiziert werden.
  - Schlimmstenfalls steht dann gesamte Software unter Open Source Lizenz.
- **Kompatibilität von Lizenzen mit strengem Copyleft**
  - Kann schwierig sein, Softwarekomponenten mit verschiedenen strengen Copylefts in einer Software zu verwenden, da jede Lizenz verlangt, den neu entstandenen Code unter die ursprüngliche Lizenz zu stellen.
- **Kompatibilität von Lizenzen mit eingeschränktem Copyleft**
  - Lizenzen mit eingeschränktem Copyleft sind grundsätzlich mit anderen eingeschränkten Copyleft- und Non-Copyleft-Lizenzen verträglich. Bei Verwendung dieser Lizenzen sind jedoch Lizenzbedingungen zur Dokumentation einzuhalten.

Die rechtssichere Einbindung

Ganz grundsätzlich bei eigentlich allen Open Source-Lizenzen

- Open Source Software darf üblicherweise frei in jedem Umfeld verwendet, modifiziert und verteilt werden
- Kopie der Lizenz muss dem Quelltext(!)-Paket beiliegen
  - GPL: Strittig, ob JavaScript-Verteilung bereits Quelltext-Paket ist.
  - Daher im Frontend Minification am besten so einstellen, dass sie Copyrights weiter beinhaltet und einen Link zum Lizenztext zur Verfügung stellt

Ganz grundsätzlich bei eigentlich allen Open Source-Lizenzen

- Manchmal müssen Änderungen am Quellcode dem Lizenzgeber zurückgeschickt werden (Lizenztext beachten! - https://tldrlegal.com/)
- Meistens bei modifizierten Dateien an auffälliger Stelle angeben, dass sie modifiziert sind (GPL, Apache bspw.)
- in der Quellform alle Original-Urheberrechtsvermerke beibehalten

Wie und wo gebe ich verwendete Lizenzen an?

- Im Quelltext
- Wenn UI, dann in der Lizenzübersicht
- Getrennt für jede Lizenz die Pakete auflisten
- Wenn auf einem Gerät ohne UI: Bspw. mit einem Label auf dem Gerät à la „Lizenzübersicht unter http://www.xyz.de/licenses.html“

Wie und wo stelle ich Sourcecode zur Verfügung, wenn laut Lizenz erforderlich?

- Je nach Lizenz ist es erforderlich, den Sourcecode der Open Source-Bibliotheken zur Verfügung zu stellen
- GPL erlaubt bspw. entweder einen Download oder einen Datenträger – oder das sog. „Written Offer“, also das schriftliche Angebot, auf Anfrage den Sourcecode zur Verfügung zu stellen.
  - Für ersteres ist eine Variante, im Internet für die Open Source-Bibliothek ein Sourcecode-Paket vorzuhalten.
  - Denkbar ist eine Unterteilung nach Softwareprodukt das man anbietet und dort Sourcecode-Pakete
  - Alternativ, wenn sich Sourcecode oft ändert oder man keine rechte Verwaltung dafür hat: Written Offer und dann Sourcecode auf Anfrage rausschicken

(L)GPL und das abgeleitete Werk („derivative work“)

- Kompliziert(er) wird es bei (L)GPL:
  - Wann wird mein Code von der (L)GPL infiziert?
  - → Wenn das neue Werk ein davon abgeleitetes Werk ist
  - Und das ist immer dann der Fall, wenn die ursprüngliche Bibliothek so eingebunden ist, dass der Lizenzgeber = Entwickler sie nicht mehr selbst ohne Weiteres tauschen kann
    - (L)GPL: Bei Einbindung des Codes in das eigene Werk [Copyleft +]
    - GPL: Immer der Fall bei statischem und (je nach Sichtweise auch:) dynamischem Linking – FSF sieht es als derivative work, OSI nicht [Copyleft +/-]
    - LGPL: Nur bei statischem Linking [Copyleft +/-]
    - Bei Frontend-Code normalerweise kein Problem, da andere Werke an sich unabhängig [Copyleft -]

Wie gehe ich mit dem „derivative work“ um?

- Auf Linking soweit möglich verzichten.
- Wenn Linking erforderlich: Code so abspalten, dass daraus ein eigenes Programm entsteht, das man dann einfach über API-Calls aufruft
  - Denn das geht immer: Reine Verwendung ist immer ohne Infizierung möglich

Was ist mit Code aus dem Internet, bspw. Stackoverflow?

- Häufig unter keiner Lizenz, daher gewertet als Schenkung
  - Anders bspw. bei Stackoverflow: CC-BY-SA („ShareAlike“ = Copyleft)
- Urheberrecht vom Entwickler bleibt aber natürlich bestehen
- Meist Einbindung problemlos möglich, aber Urheber zu nennen, bspw. auch mit Link auf den Artikel

§ 106 UrhG: Unerlaubte Verwertung urheberrechtlich geschützter Werke

- *„Wer in anderen als den gesetzlich zugelassenen Fällen ohne Einwilligung des Berechtigten ein Werk oder eine Bearbeitung oder Umgestaltung eines Werkes vervielfältigt, verbreitet oder öffentlich wiedergibt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.“*
- Vorsatztat - wann Eventualvorsatz, wann Fahrlässigkeit?
- → Scantiefe hoch
- → Lizenztexte / Urheber müssen mit vertretbarem Aufwand gefunden werden
  - Homepage,
  - GitLab, …

Was tue ich als Unternehmen, um rechtssicher einzubinden?

- Prozesse einführen
  - Bspw. mit Lizenzscannern
- Entwickler aufmerksam machen
  - Schulungen
  - Dokumentation von verwendetem Code erstellen
- Verträge mit Kunden prüfen: Sichere ich „Rechtsmangelfreiheit“ zu? Wenn möglich, nicht tun.
- Von Dienstleistern, die man selbst beauftragt, Lizenzübersichten verlangen

Was tue ich als Unternehmen, um rechtssicher einzubinden?

Lizenzscanner

Überblick

- Schwierig, alle Lizenzen herauszufinden
- Daher Lizenzscanner, bspw.:
  - Blackduck
  - WhiteSource
  - Sonarqube (mit Plugin)
  - Flexera, Gitlab (Enterprise)
  - Fossology
  - als bekannteste Produkte

Überblick

- Blackduck und WhiteSource im Wesentlichen feature- und auch preislich ähnlich
  - Blackduck mit gefälligem UI
  - WhiteSource eher wenig intuitiv in der Bedienung, aber inzwischen mit Mehr an Funktionalität
- Flexera, Gitlab (noch) deutlich hintendran
- Fossology und ScanCode Toolkit (auch in ORT eingebunden) als Freeware guter Einstieg für einen ersten Überblick

OSS Review Toolkit (ORT)

OSS
Review Toolkit

Join us on Slack! ort-talk

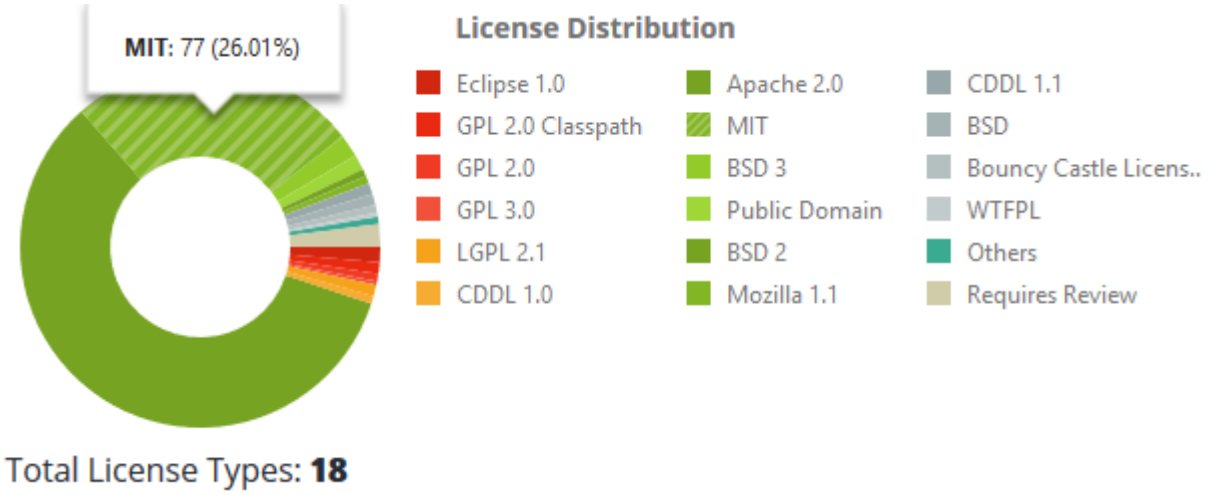
Wrapper Validation passing | Static Analysis passing

Build and Test passing | JitPack 13bc59458a | codecov 58%

TODOs 131 | lgtm alerts 5 | REUSE compliant | openssf best practices in progress 73%

WhiteSource

- Übersicht über verwendete Lizenzen aller Pakete, gegliedert im Wesentlichen nach Copyleft:
  - Rot bis orange für strenges Copyleft
  - Gelb für eingeschränktes Copyleft
  - Grün ohne Copyleft



Links

- https://opensource.org/licenses/category - Open Source Licenses by Category
- https://opensource.org/osd-annotated - The Open Source Definition (Annotated)
- https://choosealicense.com/licenses/ - OSS-Lizenzen verstehen und wählen nach einfachem Muster
- https://tldrlegal.com/ - Software Licenses in Plain English
- https://resources.whitesourcesoftware.com/research-reports/the-forrester-wave-software-composition-analysis-2019 - Übersicht über Lizenzscanner, PDF kostenlos bestellbar

Herzlichen Dank fürs Zuhören.

Dr. Falk W. Müller
falk@surrounded.de
0160 / 44 77 493