

von Rechtsanwältin **Elisabeth Keller-Stoltenhoff**

Nutzungsrechte an Freier Software (Open-Source)

Open-Source-Software (OSS) oder Freie Software sind einzelne Anwendungsprogramme oder ganze Betriebssysteme, deren Quellcode veröffentlicht ist und die frei genutzt, vervielfältigt und verändert werden können. Dies eröffnet vielen Nutzern die hochwillkommene Möglichkeit, komplexe Software lizenzfrei zu nutzen und zu verändern. Das Problem entsteht aber bei der Weitergabe, insbesondere dann, wenn der Quellcode der OSS durch den Nutzer verändert worden ist.

Denn freie Software darf nicht mit Freeware verwechselt werden. Diese ist im allgemeinen Sprachgebrauch die übliche Bezeichnung für Software, die von jedermann unentgeltlich im Objectcode genutzt und - weitgehend - ohne Einschränkungen weiterverbreitet werden darf. Der Quellcode ist jedoch (grundsätzlich) nicht frei zugänglich.

(Auszüge des Textes wurden auch veröffentlicht im IT-Rechts-Lexikon 2010)

Der folgende Beitrag will darlegen, wann Software als OSS zu qualifizieren ist und welche Chancen und Risiken die Nutzung von OSS mit sich bringt.

I. Opensource Definition

Gemäß der Opensource Definition der „Debian Free Software Guidelines“ (<http://debiananwenderhandbuch.de/dfsg.html>) wird Software als OSS angesehen, wenn ihre Lizenzbedingungen folgende Kriterien erfüllen:

1. Freie Weitergabe

Die Lizenz darf niemanden in seinem Recht einschränken, die Software als Teil eines Software-Paketes, das Programme unterschiedlichen Ursprungs enthält, zu verschenken oder zu verkaufen. Die Lizenz darf für den Fall eines solchen Verkaufs keine Lizenz- oder sonstigen Gebühren festschreiben.

2. Quellcode

Das Programm muss den Quellcode beinhalten. Die Weitergabe muss sowohl für den Quellcode, als auch für die kompilierte Form zulässig sein. Wenn das Programm in irgendeiner Form ohne Quellcode weitergegeben wird, so muss es eine allgemein bekannte Möglichkeit geben, den Quellcode zum Selbstkostenpreis zu bekommen, vorzugsweise als gebührenfreien Download aus dem Internet. Der Quellcode soll die Form eines Programms haben, das ein Programmierer vorzugsweise bearbeitet. Ein absichtlich unverständlich geschriebener Quellcode ist daher nicht zulässig. Zwischenformen des Codes, so wie sie etwa ein Präprozessor oder ein Konverter („Translator“) erzeugt, sind unzulässig.

3. Abgeleitete Software

Die Lizenz muss Veränderungen und Derivate zulassen. Außerdem muss sie es zulassen, dass die solcher Art entstandenen Programme unter denselben Lizenzbestimmungen weiter vertrieben werden können wie die Ausgangssoftware.

4. Unversehrtheit des Quellcodes des Autors

Die Lizenz darf die Möglichkeit, den Quellcode in veränderter Form weiterzugeben, nur dann einschränken, wenn sie vorsieht, dass zusammen mit dem Quellcode so genannte „Patch files“ weitergegeben werden dürfen, die den Programmcode bei der Kompilierung verändern. Die Lizenz muss die Weitergabe von Software, die aus einem veränderten Quellcode entstanden ist, ausdrücklich erlauben. Die Lizenz kann verlangen, dass die abgeleiteten Programme einen anderen Namen oder eine andere Versionsnummer als die Ausgangssoftware tragen.

5. Keine Diskriminierung von Personen oder Gruppen

Die Lizenz darf niemanden benachteiligen.

6. Keine Einschränkungen bezüglich des Einsatzfeldes

Die Lizenz darf niemanden daran hindern, das Programm in einem bestimmten Bereich einzusetzen. Beispielsweise darf sie den Einsatz des Programms in einem Geschäft oder in der Genforschung nicht ausschließen.

7. Weitergabe der Lizenz

Die Rechte an einem Programm müssen auf alle Personen übergehen, die diese Software erhalten, ohne dass für diese die Notwendigkeit besteht, eine eigene, zusätzliche Lizenz zu erwerben.

8. Die Lizenz darf nicht auf ein bestimmtes Produktpaket beschränkt sein

Die Rechte an dem Programm dürfen nicht davon abhängig sein, ob das Programm Teil eines bestimmten Software-Paketes ist. Wenn das Programm aus dem Paket herausgenommen und im Rahmen der zu diesem Programm gehörenden Lizenz benutzt oder weitergegeben wird, so sollen alle Personen, die dieses Programm dann erhalten, alle Rechte daran haben, die auch in Verbindung mit dem ursprünglichen Software-Paket gewährt wurden.

9. Die Lizenz darf die Weitergabe zusammen mit anderer Software nicht einschränken

Die Lizenz darf keine Einschränkungen enthalten bezüglich anderer Software, die zusammen mit der lizenzierten Software weitergegeben wird. So darf die Lizenz z. B. nicht verlangen, dass alle anderen Programme, die auf dem gleichen Medium weitergegeben werden, auch quelloffen sein müssen.

II. Geschichte der OSS

In den 60er und 70er Jahren des letzten Jahrhunderts war der Quellcode der von Programmierern geschriebenen Computerprogramme sehr maschinennah. Die Software stellte noch keinen wesentlichen wirtschaftlichen Wert dar, diente sie doch alleine dazu, die aufwändigen Geräte bedienbar zu machen. Sie war stets offen und: war stets austauschbar. Erst durch die Entwicklung einer eigenen Softwareindustrie, die sich von der Hardwareindustrie erst emanzipierte und sie dann majorisierte, entstand die Vorstellung von Software als eigenem wirtschaftlichem Wert. Sie wurde daher auch entsprechend geschützt und zunehmend „proprietär“ vermarktet. Das heißt, der Quellcode wurde als Betriebsgeheimnis geschützt und jeder Nutzer musste Lizenzen erwerben, um die Software nutzen zu können. Diese Entwicklung stieß bei vielen Programmierern auf Widerstand, da sie das Bedürfnis hatten, die auf dem Markt vorhandene Software an ihre eigenen Bedürfnisse anpassen zu können. Hierzu bestand aber keine (lizenz-) rechtliche Befugnis. In den 80er Jahren des letzten Jahrhunderts wurde daher der Ruf nach "freier Software" laut. 1983 gründete dann der US-Amerikaner Richard Stallmann die "Free Software Foundation" (FSF), mit dem Ziel, ein vollkommen freies, uneingeschränkt nutzbares System von UNIX-kompatibler Software zu schaffen, das sog. GNU-System. Wirtschaftlich bedeutsam und für die gesamte Computerindustrie relevant wurde diese Bewegung aber erst durch den LINUX- Erfinder Linus Torvalds. Torvalds machte 1991 den Linux-Quellcode über das Internet mit dem Ziel zugänglich, dass andere Programmierer LINUX verändern und verbessern und ebenfalls die veränderten und verbesserten Versionen frei zugänglich veröffentlichen. Seitdem hat die Verbreitung von OSS erheblich zugenommen und immer mehr Computer werden mit OSS verkauft.

III. Nutzungsrechte an OSS

Software ist in der Regel urheberrechtlich geschützt. Dies gilt auch für Freie Software oder Open-Source (OSS). Die für proprietäre Software eingeräumten Softwarelizenzen sind dahingehend ausgerichtet, die Freiheit der Nutzung, Verbreitung und Veränderung der Software einzuschränken. Der Softwarehersteller kann so durchsetzen, eine angemessene Vergütung für die Entwicklungsleistungen zu erhalten, Art und Umfang der Werknutzung zu bestimmen und die Software vor ungewollter Veränderung zu schützen. (www.it-recht-kanzlei.de/Verwertungsrechte-computerprogramme.html)

Den Entwicklern der freien Software geht es nicht um solche Einschränkungen, sondern darum, die Freiheiten der OSS-Nutzung gemäß der Open-Source-Definition (Freie Software oder Open-Source-Software) dauerhaft zu gewährleisten. Die für freie Software entwickelten lizenzrechtlichen Regelungen

dienen diesem Zweck.

IV. Lizenzmodelle für Freie Software

Richard Stallmann, der 1983 die "Free Software Foundation" (FSF) gegründet hatte, entwickelte daher bereits 1989 die GNU-General Public License (GPL), die grundlegende Lizenz für die meisten OSS-Programme.

Anknüpfend an die GPL sind entsprechend den Bedürfnissen der Lizenzgeber eine Unzahl von unterschiedlichen OSS-Lizenzen entwickelt worden, die die Freiheit der Nutzung unterschiedlich einschränken. Dabei ist in erster Linie die Strenge des so genannten Copyleft-Effektes für die Unterscheidung der einzelnen Lizenztypen maßgeblich.

Der Copyleft - Effekt (Ziffer 2 b GPL) besagt, dass Bearbeitungen der Software (derivative Work) nur unter der Ursprungslizenz weitergegeben oder veröffentlicht (distribute or publish) werden dürfen. Der Copyleft-Effekt kann unterschiedlich streng ausgestaltet sein. Es können bei einer Lizenz mit einer beschränkten Copyleft-Klausel bestimmte Ausnahmen zugelassen werden, so dass der Nutzer je nach der genauen Regelung der Ursprungslizenz bestimmte Weiterentwicklungen einer eigenen Lizenz unterstellen und so zumindest Teile eigenständig vermarkten kann.

Im Folgenden sollen die wichtigsten OSS-Lizenztypen nach Strenge des Copyleft-Effekts dargestellt werden. Ein guter Überblick über die unterschiedlichen Lizenztypen und ihrer Regelungen findet sich unter <http://www.ifross.de/>.

1. Lizenz mit strenger Copyleft-Klausel

Bei einer Lizenz mit einer strengen Copyleft-Klausel muss der Nutzer alle Weiterentwicklungen und Änderungen ausnahmslos der Ursprungslizenz unterstellen. Bekannte Beispiele für eine solche Lizenz sind die bereits angesprochene GNU General Public License (GPL) und die Common Public License (CPL).

Die wichtigste Lizenzregelung bei der GPL betrifft die Pflichten des Nutzers beim Vertrieb von veränderter Software. So muss der Nutzer neben einem Änderungsvermerk vor allem die bereits angesprochene Copyleft-Klausel befolgen. Dies hat auf Dauer zur Folge, dass die Software trotz individueller Veränderungen frei nutzbar bleibt und aufgrund der Offenlegung des Quellcodes stets von externen Nutzern weiterentwickelt werden kann.

2. Lizenz mit beschränkter Copyleft-Klausel

Bei den Lizenzen mit einer beschränkten Copyleft-Klausel kann der Nutzer – abhängig von der jeweiligen individuellen Lizenzvereinbarung – einfache OpenSource Software und Softwareteile, die unter einer anderen, eigenen Lizenz stehen, miteinander kombinieren. Dies ist insbesondere dann relevant, wenn der Nutzer eigene, „proprietäre“ Komponenten entwickelt, die er aus Gründen der besseren Vermarktbarkeit nicht vollständig der Ursprungslizenz unterstellen will. Die bekannteste existierende Lizenz dieses Lizenztyps ist die Mozilla Public License (MPL) und die GNU Lesser General Public License (LGPL). Die LGPL unterscheidet sich von der GPL dadurch, dass Software, die nicht mit GPL oder LGPL kompatibel ist, weil sie beispielsweise anderen Lizenzen unterfallen, mit LGPL-lizenzierter Software kombiniert werden kann, ohne dass sie ihre Eigenständigkeit bzw. die eigenständige Lizenzierung verliert, wenn sie nur auf vorhandenen Bibliotheken der LGPL-lizenzierten Software zugreift. Dies bedeutet, dass diese eigene Software eigenständig bleiben kann, wenn sie die bereits vorhandenen Softwareelemente nur nutzt und nicht verändert. Wenn allerdings die LGPL-lizenzierte Software selbst im Sourcecode verändert wird, so muss die veränderte Software wiederum der LGPL unterstellt werden. Somit entspricht die LGPL in diesem Punkt dem strengen Copyleft-Erfordernis der GPL.

3. Lizenz ohne Copyleft-Klausel

Bei der Lizenz ohne Copyleft-Klausel ist der Nutzer frei, von ihm veränderte Teile der Software oder hinzugefügte Softwaremodule eigenständig zu lizenzieren, ohne Rücksicht auf die bisherige Lizenzierung nehmen zu müssen. Dies bedeutet beispielsweise, dass der Nutzer „proprietäre“ Software vollkommen eigenständig lizenzieren kann oder auch unter eine andere OpenSource-Lizenz stellen kann. Der unveränderte Teil der Software verbleibt dabei unter der ursprünglichen Lizenz. Für den veränderten, „proprietären“ Teil kann der jeweilige Nutzer tatsächlich vollständig eigene Lizenzbedingungen aufstellen. So muss er beispielsweise auch nicht den Quellcode für den veränderten Teil der Software offen legen. Eine bekannte Lizenz dieses Lizenztyps ist die sog. Berkeley Software Distribution-Lizenz (BSD), die nach der US-Universität Berkeley benannt ist. Ein weiterer bekannter Lizenztyp ohne Copyleft-Klausel ist die Apache Software License. Die Apache-Lizenz ist die Freie-Software-Lizenz der Apache Software Foundation. Die aktuelle Version 2.0 wurde im Januar 2004 veröffentlicht. Sie ist gegenüber der vorherigen Version 1.1 stark erweitert worden. Auf Grund ihres Umfangs wird in den Quelltexten der einzelnen Apache-Projekte nicht mehr der komplette Text, sondern lediglich ein Verweis auf die Originallizenz eingefügt. Die Apache-Lizenz wird von der Free Software Foundation als Lizenz für freie Software anerkannt. Sie ist allerdings nicht GPL-kompatibel.

4. Lizenz mit Wahlmöglichkeiten

Neben den bisher vorgestellten Lizenztypen gibt es auch sog. Lizenzen mit Wahlmöglichkeiten. Bei diesen Lizenzen wird danach unterschieden, welche Veränderungen an der Software vorgenommen werden. Je nachdem können dann unterschiedliche Rechtsfolgen oder spezielle Wahlmöglichkeiten für die Nutzung der veränderten Software vorgesehen werden. Bekannte Lizenzen dieses Typs sind die Perl Artistic License und die Clarified Artistic License.

5. Lizenz mit Sonderrechten

Ende der 1990er Jahre haben sich verstärkt Softwareunternehmen im OpenSource-Bereich engagiert, indem sie den Quellcode ihrer bislang nur kommerziell entwickelten Software offen gelegt haben. Dabei haben diese Unternehmen teilweise eigene Lizenzen entwickelt, innerhalb derer sie sich Sonderrechte zugewiesen haben. Diese Sonderrechte bestanden z.B. darin, dass die Unternehmen als Inhaber der Verwertungsrechte an der ursprünglichen Software zwar den Quellcode offen legten und insbesondere auf eine qualitative Weiterentwicklung durch externe, freiwillige Programmierer hofften, aber sich gleichzeitig vorbehielten, die externen Weiterentwicklungen „proprietär“ zu verwenden, d.h. auch selbstständig weiterzuentwickeln und vor allem eigenständig kommerziell zu vermarkten. Dieser Lizenztyp mit Sonderrechten ist von der OpenSource-Bewegung zwar oft misstrauisch gesehen worden, jedoch sind auch diese Lizenzen als OpenSource Lizenzen anerkannt worden. Mittlerweile haben diese Lizenzen mit Sonderrechten allerdings ihre Bedeutung verloren, da die Unternehmen immer mehr auf andere der hier dargestellten Lizenztypen zurückgreifen.

V. Inhalt der GNU General Public License

1. Übersicht

Die GNU-GPL ist die wichtigste Lizenzart, die die Nutzung von Freier Software regelt. Sie wird daher im Folgenden näher dargestellt.

Die GNU GPL regelt drei Nutzungsformen der Software: das Vervielfältigen (copying), das Vertreiben (distribution) und das Verändern (modifying). Das Ausführen des Programms als Nutzung von Software im eigentlichen Sinne ist nicht eingeschränkt. Es ist nicht von der Lizenz erfasst (outside the scope). Im deutschen Urheberrecht ist dagegen bereits das Zwischenspeichern eines Programms in den Arbeitsspeicher urheberrechtlich relevant und bedarf der Zustimmung des Urhebers.

Anwendbar ist die GNU GPL auf jede Software, die mit einem Hinweis des Urhebers versehen ist, dass das nachfolgende Werk gemäß der GPL vertrieben werden darf (Abschnitt O der Lizenzbestimmungen).

2. Kopier- und Verbreitungsrecht

Im ersten Abschnitt der GPL-Bestimmungen wird die Erlaubnis erteilt, den Quelltext des Programms unverändert zu kopieren und zu verbreiten. Voraussetzung ist ein ausdrücklicher Copyright-Vermerk und ein Haftungsausschluss. Nach US-amerikanischem Recht entsteht das zu schützende Urheberrecht nur dann, wenn das Werk einen ausdrücklichen Copyright-Vermerk enthält. Weiterhin ist immer eine Kopie der Lizenz an den Empfänger der Programme mit zu senden. Meist sind die Lizenzbestimmungen im Anhang der Programmdatei enthalten oder es findet sich ein Verweis auf die Homepage der FSF. Die Lizenzen werden dezentral weitergegeben, d.h. nicht vom Urheber selbst, sondern von Nutzer zu Nutzer.

3. Lizenzgebührenverbot

Für die Vervielfältigung und Verbreitung der GPL-Software darf keine Lizenzgebühr verlangt werden. Dies betrifft nicht das Recht, ein Entgelt für die Kopierkosten und die Kosten für die Erstellung eines Handbuchs und für sonstige mit dem Erwerb der Software verbundenen Dienstleistungen zu verlangen. Dies bedeutet aber, dass die GPL-Software auch dann unbegrenzt vervielfältigt und weiterverbreitet werden darf, wenn sie von einem Distributor als kommerzielle Distribution verkauft wurde.

4. Copyleft-Effekt

Der Copyleft Effekt (Ziffer 2 b GPL) besagt, wie oben aufgeführt, dass Bearbeitungen der Software (derivative Work) nur unter der Ursprungslizenz weitergegeben oder veröffentlicht werden (distribute or publish) werden dürfen.

"2b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License."

4.1 Weitergabe

Die Frage, was unter Weitergabe im Sinne der GPL zu verstehen ist, wirft schwierige Abgrenzungsfragen auf.

Tipp: Es ist gesichert, dass eine Weitergabe innerhalb eines Rechtsträgers, also innerhalb einer Firma nicht unter den Begriff „distribute“ im Sinne der GPL fällt, nicht aber die Weitergabe innerhalb eines Konzerns im Sinne von §§ 16 ff. AktG.

4.2 „derivative work“

Sehr umstritten ist, was unter einem „derivative work“ anzusehen ist. Es genügt hier nicht, dass Softwarebestandteile unabhängig sind, auch deren Verbreitung muss als eigenständiges Werk erfolgen. Auch ist es gerade bei Software schwierig zu entscheiden, was als Teil eines Ganzen (in whole or in part) anzusehen ist. Anhaltspunkte ergeben sich aus der folgenden Aufstellung.

- Ergänzung einer bestehenden Datei: Wird eine unter der GPL lizenzierte Software so geändert, dass bestehende Dateien mit eigenem Code erweitert werden oder ein bestehender Code ersetzt wird, darf dieser Code stets nur unter der GPL vertrieben werden.
- Gemeinsamer Vertrieb auf einem Datenträger: Wird ein eigenes Programm mit einer GPL-Software auf einer CD-ROM oder einem anderen Datenträger zusammen vertrieben, kann das eigene Programm einer beliebigen Lizenz unterstellt werden, wenn die Programme eigenständig sind.
- Anwendungsprogramme für Linux: Sofern ein für Linux geschriebenes Anwendungsprogramm nur über normale Systemaufrufe mit dem Linux-Kernel abläuft und im so genannten »Userland« verbleibt, muss es nicht der GPL unterstellt werden.
- Verwendung von Softwaretools, Editoren, Kompilern: Sofern GPL-Softwaretools als Hilfsmittel zur Erstellung eines Programms verwendet werden und kein Code dieser Werkzeuge in die so erzeugte Software eingefügt wird, müssen diese auch nicht der GPL unterstellt werden.
- Kernelmodule: Umstritten ist die Lage bei Kernelmodulen. In Linux wurden »Binary Only Modules« geduldet; dies sagt allerdings nichts über die Zulässigkeit aus. Man wird keine einheitliche Aussage dazu geben können, sondern darauf abstellen müssen, ob im Einzelfall ein getrennter Vertrieb vorliegt.
- Programmbibliotheken: Für Programmbibliotheken, die der GPL unterstellt sind, gilt Ähnliches wie für Kernelmodule: Es hängt von der Art der Bibliothek und der Form des Vertriebs ab, ob das auf die Bibliothek zugreifende Programm der GPL unterstellt werden muss. Entsprechend ist die Lage bei der umgekehrten Situation, wenn das zugreifende Programm der GPL untersteht.
- Kommunikation über Schnittstellen: Über Schnittstellen können sowohl eigenständige Programme als auch »abhängige« Bestandteile angebunden werden. Daher kann nur anhand des Umstands, dass eine Schnittstelle verwendet wird, noch keine Aussage darüber getroffen werden, ob der so eingebundene Code unter der GPL lizenziert werden muss. Findet nur ein Informationsaustausch anhand von Kommunikationsmitteln statt, die üblicherweise für das Zusammenwirken eigenständiger Programme verwendet werden, verlangt die GPL nicht, dass ihr der fremde Code unterstellt werden muss. Dies ist im Regelfall bei der Verwendung von Pipes, Sockets und Kommandozeilenargumenten der Fall. Bei einer engeren Verbindung über die Kommunikation hinaus, insbesondere wenn durch die Kombination die Struktur der Softwarebestandteile betroffen ist, kann im Einzelfall ein »abgeleitetes Werk« vorliegen.

Der Copyleft-Effekt schränkt den Nutzer also stark ein. Er darf eigene Weiterentwicklungen nicht eigenständig kommerziell vermarkten, sondern muss sich an die vereinbarten Lizenzbedingungen halten und auch die Weiterentwicklungen den Ursprungslizenzen unterstellen.

5. Regelung zum Lizenzverstoß

Werden die Lizenzbedingungen nicht eingehalten, liegt ein Lizenzverstoß vor. Der Lizenzgeber kann gegen den Verletzer zivil- oder strafrechtlich vorgehen. Unklar ist allerdings sowohl nach deutschem als auch nach US-amerikanischem Recht, ob die FSF Inhaberin der ursprünglichen Urheberrechte an der OSS ist. Denkbar ist auch, dass derjenige Inhaber der Urheberrechte ist, der jeweils die Lizenz weitergibt. Zur Zeit ist nicht eindeutig geklärt, wer befugt ist, Lizenzverstöße gerichtlich zu verfolgen. Nach deutschem Recht ist Torvalds in Bezug auf Linux Inhaber der ursprünglichen Urheberrechte. Die nachfolgend an der Weiterentwicklung beteiligten Programmierer können je nach Umständen Miturheber (§ 8 UrhG), Urheber eines verbundenen Werkes (§ 9 UrhG) oder lediglich Bearbeiter sein. Dies führt in der Praxis dazu, dass nur der ursprüngliche Urheber faktisch die Möglichkeit hat, Lizenzverstöße zu verfolgen.

Aber die GPL hat sich dennoch in der Gerichtspraxis bewährt. So entschied das Tribunal de grande instance de Paris, das vergleichbar ist mit einem deutschen Landgericht, das in erster Instanz entscheidet, mit Urteil vom 28. März 2007, dass Bestimmungen der GPL wirksam vereinbart und anzuwenden sind. Es schaffte damit einen wichtigen Präzedenzfall für die gerichtliche Durchsetzung der GPL.

Auch das Landgericht München I erkannte mit Urteil vom 24.7.2007 erneut die Wirksamkeit der GPL (Az.: 7 O 5245/07) an.

6. Änderungsrecht

Der zweite Abschnitt der Lizenz regelt, dass eine Veränderung am Programm vorgenommen und das abgeänderte Werk wiederum frei verbreitet und vervielfältigt werden kann. Voraussetzung dafür ist, dass die weitergegebene Version einen Copyright-Vermerk und einen Haftungsausschluss enthält. Außerdem muss die weitergegebene Version den deutlich sichtbaren Hinweis enthalten, welche Veränderungen vorgenommen wurden. Das Datum der Veränderung ist einzufügen.

7. Haftung

Abschnitt 11 der GPL schließt jegliche Gewährleistung für das Programm "soweit gesetzlich zulässig" aus, "da die Software kostenlos zur Verfügung gestellt worden ist." Findet hingegen deutsches Recht Anwendung, so halten die Bestimmungen einer AGB-rechtlichen Kontrolle nicht stand. Die Haftung darf nach den §§ 309 ff. BGB allenfalls für leichte Fahrlässigkeit ausgeschlossen werden. Auch ein kompletter Ausschluss der Gewährleistungsrechte ist nicht zulässig, da die Software i.d.R einer neu hergestellten Sache gleichsteht. Nachdem auch eine geltungserhaltende Reduktion der Klauseln auf das gesetzlich zulässige Maß unzulässig ist, hätte dies eine komplette Nichtigkeit der Haftungs- und Gewährleistungsausschlüsse der GPL zur Folge. Daran ändert auch die in der GPL V3 (siehe unten) verwendete salvatorische Klausel („soweit gesetzlich zulässig“) nichts, da sie einen Verstoß gegen das gesetzliche Transparenzgebot bedeutet. Dem Nutzer muss allein aufgrund des Lizenztextes bewusst sein, welche Rechte ihm bei einer Beschränkung der Haftung und Gewährleistung verbleiben. Die

Einholung von Rechtsrat zur Bestimmung des „gesetzlich Zulässigen“ wird dem Nutzer von der Rechtsprechung nicht zugemutet.

Daraus folgt, dass die beschränkenden Bestimmungen der GPL als Ganzes unwirksam sind. An deren Stelle treten die gesetzlichen Regeln: Wegen der Unentgeltlichkeit ist die reine Überlassung von OSS regelmäßig als gemischter Schenkungsvertrag einzuordnen. Der Entwickler haftet damit nur für Vorsatz und grobe Fahrlässigkeit. Gewährleistungspflichten treffen ihn nur, wenn er Mängel der Software arglistig verschwiegen hat. Eine Arglist ist in der Praxis allerdings nur schwer zu beweisen. Somit sind auch unter der Geltung deutschen Rechts Ansprüche gegen den Entwickler aus einem Schenkungsvertrag kaum realisierbar. Andere vertragliche Konstellationen können im Einzelfall zu Gewährleistungsrechten führen.

Etwas anders stellt sich die Situation in Bezug auf den Distributor dar. Soweit er Eigenleistungen, insbesondere im Bereich der Zusammenstellung, der Installationsroutinen oder der Dokumentation erbringt, verlangt er im Gegenzug regelmäßig eine Vergütung. In der Folge richten sich die Rechte der Beteiligten nach Kaufrecht oder Werkvertragsrecht. Die Verwaltung kann dann die bekannten Gewährleistungsansprüche nach dem Bürgerlichen Gesetzbuch (BGB) wie Nacherfüllung, Minderung, Rücktritt und/oder Schadenersatz geltend machen. Voraussetzung ist das Vorliegen eines Mangels, beispielsweise wenn die ausgewählten Komponenten inkompatibel sind oder Installationsroutinen nicht ordnungsgemäß funktionieren.

Zur Lösung des Problems bieten sich verschiedene Ansätze an. Zum einen könnte ein externes Unternehmen mit der Auswahl, Beschaffung und Installation der OSS beauftragt werden. Dann haftet das Unternehmen entsprechend den vertraglichen Regelungen als Generalunternehmer. Eine andere Möglichkeit eröffnen einige Distributoren selbst: Gegen eine entsprechende Vergütung übernehmen sie die Gewährleistung und bieten Ihren Kunden so ein gewisses Maß an rechtlicher Sicherheit.

VI. Überblick über die wichtigsten Änderungen der GPL Version 3

Am 29. Juni veröffentlichte die FSF die GPL Version 3 (<http://www.gnu.org/licenses/gpl-3.0.html> wurde). Sie trat am 1. Juli 2007 in Kraft. Grund für die Revision war, dass die 16 Jahre alte Vorgängerversion keine passenden Regelungen für technische und rechtliche Neuerungen hatte. Auch sollte eine bessere Internationalisierung erreicht werden und die starke Prägung der Lizenz durch das US-Rechts gelockert werden. Die Version 3 bemüht sich daher um eine neutralere Sprache und berücksichtigt auch andere Urheberrechtssysteme. Die wichtigsten Änderungen der GPL V3 zur GPL V 2 stellen sich wie folgt dar:

1. Einführung einer Patentregelung

Die GPL 3 enthält eine Patent-Lizenz, die bisher nicht ausdrücklich geregelt war. Die neuen Regelungen postulieren, dass jeder, der GPL-Software vertreibt, für die er Patente benutzt hat, auch anderen ermöglichen muss, diese Patente voll zu nutzen. Entweder darf also jeder diese Patente nutzen oder keiner. Das ist jedenfalls die Intention. Ob das alles auch in der Praxis so funktioniert, wird sich erweisen.

2. Neue Kompatibilitätsregelungen

Gemäß der GPL 2 fallen bei einer Lizenzverletzung die Rechte des Verletzers weg. Er darf die Software nicht mehr nutzen. Dies wurde in der GPL 3 beibehalten, allerdings mit einer besonderen Regelung, die eine Wiederherstellung von Rechten vorsieht, wenn sich der Lizenznehmer rechtskonform verhält. Man hat damit eine Möglichkeit gefunden, die sowohl unter europäischem, als auch unter US-Recht wirksam sein soll.

3. Wahl der Lizenzart

Ziffer 14 regelt ein Wahlrecht des Lizenzgebers, seine Software unter einer bestimmten Version der GPL anzubieten und zusätzlich im Wege einer dynamischen Verweisung auch die Freigabe hinsichtlich jeder späteren neuen Version zu erteilen. Macht der Lizenzgeber keine Angaben, kann der Nutzer entscheiden, welcher Version der GPL er sich unterwerfen will. Für welche Anwender der Umstieg auf die GPL V3 vorteilhaft ist, lässt sich pauschal nicht sagen. Insoweit ist insbesondere aufgrund der geplanten Nutzungs- und Anwendungszwecke der Software zu entscheiden, ob die entsprechenden Einschränkungen und Verbote für technische Schutzmaßnahmen und Software-Patente akzeptabel sind.

4. Einführung einer DRM-Regelung

Durch das Digital-Rights-Management (DRM) werden die durch die GPL eingeräumten Nutzungsrechte faktisch genommen, Solche Nutzungseinschränkungen widersprechen nach Ansicht der FSF dem Grundsatz der freien Verwendbarkeit von GPL-Software. In Ziffer 3 ist nun geregelt, dass keine GPL-Software für DRM eingesetzt werden darf.

5. Kompatibilität mit anderen Lizenzen, Auflockerung des strengen Copyleft-Effektes

In Ziffer 7 soll der strenge Copyleft-Effekt abgemildert werden. So werden auf der einen Seite nunmehr Lizenzen mit fehlender oder abgeschwächter Copyleft-Klausel anerkannt, die dem Nutzer mehr Freiheiten als nach der GPL zugestehen. Solche „additional permissions“ werden künftig einfach als Bestandteil der Lizenz angesehen. Andererseits werden auch Lizenzen akzeptiert, die den Anwendern zusätzliche Pflichten auferlegen, sofern sich diese in dem aufgeführten Pflichtenkatalog zu Ziffer 7

wiederfinden. Solche zulässigen weiteren Pflichten sind etwa von der GPL-Lizenz abweichende Garantie- oder Gewährleistungsregelungen oder eine mit der Lizenzierung verbundene Einschränkung der Nutzung von Markenrechten. In Ziffer 13 ist schließlich die Kompatibilität mit der Affero General Public License (AGPL) geregelt, wonach die Lizenzbedingungen der AGPL im Falle der Verbindung von GPL-Software mit AGPL-Software unverändert fortbestehen.

6. Anpassung der Haftungsregelungen an nationales Recht

Der Unwirksamkeit der Haftungsklauseln der GPL V2 nach anderen Rechtsordnungen wird nunmehr in Ziffer 7 der GPS V3 Rechnung getragen. Unternehmern wird beim Vertrieb eigener Software neben der GPL-Software die Möglichkeit eröffnet, strengere Haftungs- und Gewährleistungsbestimmungen aufzunehmen. Da die strengeren Haftungs- und Gewährleistungsbestimmungen jedoch nur bezüglich der neu hinzugefügten Softwareteile vereinbart werden dürfen und daher die alten Programmbestandteile weiterhin den unveränderten Bestimmungen der GPL V2 unterliegen, verbleibt es bei der Unwirksamkeit der Haftungsausschlüsse für die alten Programmbestandteile wegen Verstoßes gegen das AGB-Recht.

Fazit

Freie Software ist, wie dieser Artikel zeigt, nicht wirklich frei. Sie unterliegt komplexen Lizenzbedingungen, die unbedingt zu beachten sind, will der Nutzer unangenehme rechtliche Konsequenzen und zum Teil sehr hohe Schadensersatzansprüche vermeiden. Die OSS-Gemeinde ist großzügig bei der Weitergabe ihrer Arbeitsergebnisse, reagiert aber empfindlich, wenn die Freie Software proprietär verwertet wird. Die Nutzung und Weitergabe von Freier Software sollte daher mit großer Sorgfalt erfolgen. Insbesondere sollte Wert darauf gelegt werden, dass bei der Gestaltung von Lizenzverträgen, die sich auch auf freie Software beziehen, den Lizenzbedingungen, denen die jeweilige Freie Software unterliegt, Rechnung getragen wird.

(Auszüge des Textes wurden auch veröffentlicht im IT-Rechts-Lexikon 2010)

Autor:

RAin Elisabeth Keller-Stoltenhoff

Rechtsanwältin