

Piotr Gawrysiak

Licencje wolnego oprogramowania

Zeszyty Prawnicze 8/1, 193-203

2008

Artykuł został opracowany do udostępnienia w internecie przez Muzeum Historii Polski w ramach prac podejmowanych na rzecz zapewnienia otwartego, powszechnego i trwałego dostępu do polskiego dorobku naukowego i kulturalnego. Artykuł jest umieszczony w kolekcji cyfrowej bazhum.muzhp.pl, gromadzącej zawartość polskich czasopism humanistycznych i społecznych.

Tekst jest udostępniony do wykorzystania w ramach dozwolonego użytku.

PIOTR GAWRYSIAK

Politechnika Warszawska

LICENCJE WOLNEGO OPROGRAMOWANIA

I. WPROWADZENIE

Przeciętnemu współczesnemu konsumentowi oprogramowanie komputerowe jawi się jako jeden z wielu towarów, dostępnych w obrocie handlowym. Jest ono przedmiotem – na pozór – umów sprzedaży, jest dostępne w ofercie sklepów detalicznych takich jak sklepy wielkopowierzchniowe. W odróżnieniu od klasycznych wytworów kultury niematerialnej, takiej jak dzieła literackie, muzyka czy też filmy, jest ono także wykorzystywane jako narzędzie pracy – choć oczywiście nie oznacza to, iż nie tworzone jest także takie oprogramowanie, którego jedynym celem istnienia ma być dostarczenie konsumentowi rozrywki.

W obowiązującym polskim porządku prawnym, oprogramowanie traktowane jest jednak podobnie do innych wytworów kultury niematerialnej¹. Jest to zrozumiałe, jako że to właśnie wirtualność oprogramowania – i związany z tym zerowy krańcowy koszt jego produkcji – stanowi podstawową cechę, odróżniającą je od tych narzędzi, które istnieją w przestrzeni realnej – takich jak pióro, młotek, czy choćby komputer. Tym samym oprogramowanie nie jest zwykle sprzedawane użytkownikom końcowym, ale udzielane są im licencje pozwalają-

¹ Por. art. 74 Ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (Dz.U. Nr 24 poz. 83).

ce uruchomić *kopię* programu na komputerze użytkownika. Dowolne zwielokrotnianie przedmiotowego oprogramowania przez użytkowników, pomimo iż technicznie możliwe i nie przynoszące bezpośrednich strat wytwórcy, czy też dystrybutorowi oprogramowania, traktowane jest jako przestępstwo², które w języku potocznym określane jest mianem *piractwa komputerowego*.

II. KONCEPCJA WOLNEGO OPROGRAMOWANIA

Warto zauważyć, iż powyższe ograniczenie prawa użytkownika oprogramowania do jego zwielokrotniania i ewentualnej późniejszej redystrybucji, jest wyłącznie ograniczeniem natury prawnej nie zaś technicznej czy ekonomicznej. Utworzenie kolejnej kopii programu jest bowiem czynnością niedestrukcyjną w stosunku do oryginału, zaś z racji jego cyfrowej natury także nie wiąże się z ponoszeniem jakichkolwiek kosztów, niezbędnych w przypadku wykonywania choćby tradycyjnych kopii treści drukowanych (np. kserowanie książek). Ograniczenie to zatem służy wyłącznie ochronie interesów twórcy oprogramowania, zaś w istotny sposób ogranicza możliwości jego wykorzystania przez użytkowników.

Choć może wydawać się to zaskakujące, to jednak nie każdy twórca oprogramowania komputerowego będzie zainteresowany tego rodzaju ochroną, oferowaną domyślnie przez istniejący system prawny. Już bowiem we wczesnych latach siedemdziesiątych XX wieku wskazywać zaczęto na to, iż osiąganie zysków z owych, wyłącznie prawnych, ograniczeń redystrybucji oprogramowania nie zawsze będzie rzeczą moralną³. Zwolennicy tego podejścia, z których wymienić należy przede wszystkim Richarda Stallmana, byłego pracownika Massachusetts Institute of Technology, postulują, iż oprogramowanie komputerowe, ze względu na swą istotność dla funkcjonowania społeczeństwa informacyjnego, nie powinno podlegać restrykcjom takim, jak stosowane

² Por. art. 116 i n. o prawie autorskim i prawach pokrewnych.

³ Por. J. BARTA, J. MARKIEWICZ, *Oprogramowanie open source w świetle prawa. Między własnością a wolnością*, Zakamycze, Kraków 2005.

w przypadku innych treści określanych mianem „własności intelektualnej” (ang. *intellectual property*). Zamiast tego, powinno być przedmiotem wolnej wymiany – co zakłada możliwość nieodpłatnego jego kopiowania, oraz powinno być, podobnie jak to ma miejsce w przypadku wytworów kultury materialnej, dostępne do dowolnej modyfikacji dla każdego swojego użytkownika⁴.

Oprogramowanie komputerowe traktowane w powyższy sposób określane bywa mianem *wolnego oprogramowania* – od angielskiego terminu „free software”. Określenie to wymaga komentarza, ponieważ angielskie słowo *free* może być rozumiane rozmaicie. Oznaczać może bowiem nieodpłatność pewnego dobra materialnego, czyli jego zerową cenę, w tym znaczeniu odpowiadając polskiemu słowu „darmowy”. Jednocześnie jednak znaczyć może wolny, nie zniewolony. Zwolennicy koncepcji wolnego oprogramowania, w tym w szczególności Richard Stallman, podkreślają, iż istotne jest dla nich właśnie to drugie znaczenie. Koszt egzemplarza oprogramowania nie jest czynnikiem decydującym o tym, czy dany program jest „wolnym oprogramowaniem” czy też nie. Samo zaś rozróżnienie pomiędzy oboma powyższymi homonimami bywa często – w nieco humorystyczny sposób – przedstawiane następująco: *free as in freedom* (wolny – jak w wolności), co dotyczy wolnego oprogramowania, oraz *free as in beer* (bezpłatny – jak w darmowym piwie).

Jakie zatem cechy powinien spełniać program komputerowy, aby można było określić go mianem „wolnego oprogramowania”? Można się tu posłużyć definicją⁵ opracowaną przez założoną przez Stallmana fundację *Free Software Foundation*, określającą zbiór praw użytkownika oprogramowania, które spełnione być winny przez warunki licencji programu, jaki jest wolnym oprogramowaniem. Prawa te są następujące:

⁴ Por. R.M. STALLMAN, *Free Software, Free Society*, Free Software Foundation, Boston, Massachusetts 2002.

⁵ Por. FREE SOFTWARE FOUNDATION, *The Free Software Definition*, dostępne w Internecie <http://www.gnu.org/philosophy/free-sw.html> dostęp 29 kwietnia 2008.

Wolność zerowa (freedom 0): prawo do uruchomienia i wykorzystywania programu (jako narzędzia) w dowolnym celu (ang. *The freedom to run a program, for any purpose*);

Wolność pierwsza (freedom 1): prawo do analizy sposobu działania programu i jego modyfikacji dla własnych potrzeb (ang. *The freedom to study how a program works, and adapt it to your needs*);

Wolność druga (freedom 2): prawo do kopiowania i redystrybucji programu, tak by inni mogli go wykorzystać (ang. *The freedom to redistribute copies of a program so you can help your neighbor*);

Wolność trzecia (freedom 3): prawo do poprawiania programu i do upubliczniania poprawek, tak aby korzyść mogła odnieść cała społeczność użytkowników programu (ang. *The freedom to improve the program, and release your improvements to the public, so that the whole community benefits*).

Traktowane jako całość, powyższe prawa są *de facto* przeniesieniem praw naturalnych, dotyczących przedmiotów kultury materialnej, na obiekty wirtualne, jakimi są programy komputerowe. W przypadku przedmiotów materialnych wszystkie owe wolności są niejako „wbudowane” w cechy samych przedmiotów i uważane są – a przynajmniej ma miejsce to w kulturze europejskiej – za naturalne prawa posiadacza przedmiotu. Oczywiście nie dotyczy to wolności drugiej, tj. prawa do kopiowania i redystrybucji oprogramowania. Tego rodzaju działanie – czyli natychmiastowa i nie pociągająca za sobą kosztów duplikacja pewnego przedmiotu – jest bowiem fizycznie nie do zrealizowania w odniesieniu do przedmiotów niewirtualnych.

Powyższe stwierdzenie jest pewnego rodzaju uproszczeniem – współczesne metody produkcji przemysłowej pozwalają bowiem na wytwarzanie dóbr materialnych przy ponoszeniu często minimalnych kosztów. A zatem i w przypadku tych przedmiotów możemy mieć

do czynienia z próbami ich kopiowania – i rzeczywiście ma to miejsce. Produkty markowe są często przedmiotem takiego proceduru kopiowania, w którym wytwarzane są egzemplarze opatrzone logo innego, niż faktyczny wytwórca producenta. Proces ten jednak, pomimo iż na pozór podobny do kopiowania oprogramowania, odróżnia jednak przede wszystkim niższa jakość wytwarzanych kopii. Kopiowane towary przedstawiają dla konsumenta zdecydowanie odmienną, niż „oryginalne” egzemplarze wartość – w przypadku oprogramowania zaś, nie ma żadnej różnicy pomiędzy kopią, a oryginałem.

III. LICENCJA GNU PUBLIC LICENSE

Wolne oprogramowanie, w tym w szczególności koncepcja wymienionych powyżej „wolności” zdefiniowanych przez Richarda Stallmana nie byłaby niczym innym jak tylko ciekawostką filozoficzną, gdyby nie to, iż podjęto próby zastosowania tej koncepcji w praktyce. Jak bowiem okazało się, liczba tych twórców oprogramowania którzy *pragną* by efekt ich pracy – oprogramowanie – było powszechnie i bezpłatnie dostępne dla innych osób, zgodnie z zasadami określonymi przez Richarda Stallmana, jest dość pokaźna. Mogłoby się wydawać, iż dla tychże twórców oprogramowania, takie „upublicznienie” swoich dzieł nie powinno być zadaniem trudnym. Wystarczy tu bowiem oświadczenie woli, w wyniku którego twórca programu zrzeknie się swoich praw majątkowych, czyniąc tym samym ów program produktem publicznym – *public domain* – przy jednoczesnym zapewnieniu, iż wraz z kodem wykonywalnym programu komputerowego (tj. jego formą nadającą się do bezpośredniego wykonania przez komputer) będzie dystrybuowany jego *kod źródłowy* (ang. *source code*) – tj. jego postać czytelna dla człowieka, którą można dowolnie modyfikować oraz przetłumaczyć w sposób automatyczny do postaci kodu wykonywalnego.

W praktyce jednak tego rodzaju działanie nie zawsze będzie skuteczne, w szczególności dotyczyć to może formy tego oświadczenia – która na przykład, z racji specyfiki dystrybucji treści cyfrowych takich jak oprogramowanie, zwykle nie będzie mieć choćby formy pisemnej.

Ponadto, jeśli intencją twórcy jest zapewnienie „otwartości” swojego dzieła, w myśl wolności sformułowanych przez Stallmana, to zrzeczenie się praw majątkowych nie będzie tu działaniem wystarczająco skutecznym. Nieetyczne jednostki mogą bowiem redystrybuować oprogramowanie już tylko wyłącznie w postaci kodu wykonywalnego, bez kodu źródłowego – tym samym pozbawiając kolejnych odbiorców możliwości jego modyfikacji. Co więcej – redystrybucja ta może być prowadzona nawet na zasadach ściśle komercyjnych (pamiętajmy bowiem, iż twórca programu zrzekł się praw do niego).

Z jednej strony, mamy tu zatem do czynienia ze zbyt restrykcyjną w stosunku do potrzeb twórców oprogramowania ochroną, jaką zapewnia prawo autorskie poprzez mechanizm copyright. Z drugiej zaś strony grozi niebezpieczeństwo wypaczenia idei wolnego oprogramowania (jako dostępnego dla wszystkich do wykorzystania i modyfikacji) w przypadku całkowitej rezygnacji z tego mechanizmu. Oczywiście staje się więc konieczność dystrybucji takiego oprogramowania w ramach licencji, skonstruowanych tak, by zezwalać licencjobiorcom (czyli potocznie mówiąc – nabywcom oprogramowania) na redystrybucję kopii programu i dowolną jego modyfikację, ale obligować ich do przeniesienia pełni tych praw na kolejnych właścicieli.

Opracowanie wzoru takiej umowy, z którego mógłby skorzystać każdy twórca wolnego oprogramowania, stało się jednym z priorytetów wspomnianej już fundacji Free Software Foundation. Licencja ta, opracowana w 1989 roku, nazwana została GNU General Public Licence, w skrócie GPL⁶. Nie jest to obecnie jedyna licencja stosowana przez twórców wolnego oprogramowania (z innych warto tu zapewne wymienić licencję BSD oraz licencję Apache Foundation), jednakże jest z pewnością najpopularniejsza. Szacuje się, iż obecnie ponad połowa ogólnej liczby pakietów spełniających warunki wolnego oprogramowania została upubliczniona przy użyciu tej licencji.

⁶ Prof. FREE SOFTWARE FOUNDATION, *The GNU General Public License*, dostępne w Internecie <http://www.gnu.org/licenses/gpl-3.0.html>, dostęp 26 kwietnia 2008.

Z czysto prawnego punktu widzenia GPL jest umową o przekazaniu części praw własności intelektualnej⁷, podobną do umów zawieranych choćby pomiędzy wydawcą książki, a jej autorem. O ile jednak w tym ostatnim przypadku celem umowy jest ochrona praw autorskich – zwanych copyright (dosłownie – „prawo kopii”, co można interpretować jako przekazanie prawa do tworzenia kolejnych kopii dzieła), to celem GPL jest ochrona dowolności kopiowania programu. W szczególności licencja pozwala na dalszą redystrybucję oprogramowania tylko i wyłącznie wtedy, gdy kolejne kopie programu także będą objęte licencją GPL. Innymi słowy, jeśli jakieś oprogramowanie zostanie „upublicznione” poprzez zastosowanie licencjonowania zgodnego z GPL, to jest to proces nieodwracalny. Wykorzystanie tego programu w produktach strictly komercyjnych, takich w których częścią produktu nie jest kod źródłowy, jest po prostu zakazane. Efekt ten zwany jest często „copyleft” – licencja GPL wykorzystuje tutaj bowiem mechanizmy prawa autorskiego, zaprojektowane do chronienia „copyright”, aby osiągnąć dokładnie przeciwny efekt⁸. Z tego też powodu licencja ta porównywana jest niekiedy do organizmu wirusowego – „zarażony” nim pojedynczy egzemplarz programu staje się nosicielem „wirusa GPL”, jako że posiadać go będą wszystkie jego kopie.

Przyrównanie licencji GPL do wirusa ma jeszcze jedno dodatkowe uzasadnienie. Ma ona bowiem możliwość „zarażania” innych programów, w tym komercyjnych, choć oczywiście terminologia „epidemiologiczna” jest tutaj użyta jedynie obrazowo. Otóż dla programisty, pakiet oprogramowania opublikowany na licencji GPL z dostępnym kodem źródłowym jest czymś więcej niż tylko narzędziem, którym jest dla zwykłego, tj. takiego, który nie tworzy nowego oprogramowania, użytkownika komputera. Może być także komponentem

⁷ Odnotować należy jednak, że kwestia nie jest bezsporna i wyrażane były też poglądy, iż oświadczenie woli autora, który decyduje się udostępnić swój program jako wolne oprogramowanie, traktować należy jako zrzeczenie się przez niego majątkowych praw autorskich. Por. J. BARTA, J. MARKIEWICZ, *op. cit.*, s. 79 i n.

⁸ Por. J. BARTA, J. MARKIEWICZ, *op. cit.*, s. 49 i n.

nowych programów. Części kodu źródłowego oryginalnego programu, realizujące szczególnie wartościowe funkcje, mogą być bowiem ponownie użyte w nowych programach, poprzez ich włączenie do nowo opracowywanego kodu. Korzyść dla programisty tworzącego ów nowy program jest oczywista – nie dość, że wykorzystanie takiego „kawałka” innego programu oszczędza czas, to na dodatek, ponieważ wykorzystywany kod źródłowy pochodzi zwykle z programu, który był używany przez wiele osób na różnych komputerach – to jest on dobrze przetestowany i tym samym pozbawiony wielu błędów, jakie są nie do uniknięcia w przypadku tworzenia aplikacji od podstaw. Powstaje tu jednak pytanie – co z autorem oryginalnego programu? Nie odnosi on przecież żadnych korzyści z faktu wykorzystania jego pracy nie tylko materialnych ale także moralnych. W szczególności może nawet nie zdawać sobie sprawy z tego, iż jego dzieło zostało ponownie użyte.

Powyższa sytuacja została uwzględniona w warunkach licencji opracowanych przez Stallmana. Jeśli bowiem decydujemy się inkorporować jakiś element programu objętego licencją GPL do własnego programu, to ten ostatni także musimy bezwarunkowo upublicznić wykorzystując tę samą licencję! Oznacza to, iż jeśli chcemy czerpać korzyści z zasobów oprogramowania dostępnego na licencji GPL to musimy także wnosić swój własny wkład do tychże zasobów. Rozwiązanie to nie gwarantuje oczywiście zysków autorom oprogramowania – zapewnia im jednak osobistą satysfakcję, oraz gwarancję tego, iż opracowane przez nich rozwiązania zawsze pozostaną wolnym oprogramowaniem. W praktyce wymaga to rzecz prosta proaktywnych działań prawnych, przypadki nawet świadomego „łamania” postanowień licencji GPL nie są odosobnione. Ochroną praw autorów programów, którzy zdecydowali się na ich objęcie licencją GPL zajmuje się m.in. Free Software Foundation oraz, od 2004 roku, organizacja GPL Violations⁹.

⁹ Por. *GPL Violations*, dostępne w Internecie, <http://www.gpl-violations.org>, dostęp 26 kwietnia 2008.

VI. ZAKOŃCZENIE

Jak łatwo zauważyć, opisana powyżej konstrukcja licencji GPL podporządkowana jest celom, określonym w kategoriach wyłącznie moralnych. Motywacją dla opracowania owej licencji, a także motywacją, jaką kierują się (generalizując oczywiście) twórcy oprogramowania je wykorzystujący, jest przekonanie o tym, iż oprogramowanie powinno być dostępne dla każdego na warunkach określonych przez wolności Stallmana. Ma to być oprogramowanie wolne – *free*. Nie są tutaj przyjmowane żadne założenia dotyczące choćby jakości oprogramowania, czy też jego przyjazności dla użytkownika.

Powyższy pogląd nie jest powszechny. Wielu twórców oprogramowania argumentuje, iż dostępność kodu źródłowego programów komputerowych pozwala opracowywać produkty technologicznie doskonalsze, niż te, w których kod wgląd ma jedynie niewielka grupka pracowników producenta. Innymi słowy, program z dostępnym kodem źródłowym – (ang. *open source*) – powinien być lepszy od programu czysto komercyjnego, zamkniętego (ang. *proprietary*), zaś ewentualne kwestie etyczne związane z jego dostępnością traktowane są tu jako drugorzędne. To podejście doprowadziło do powstania licencji, których celem nie jest ochrona przed możliwością ponownego „zamknięcia” kodu źródłowego, lecz raczej zachęta do jak najszerszego wykorzystywania objętego nimi oprogramowania. Z formalnoprawnego punktu widzenia licencje te stanowią w zasadzie usankcjonowanie praktyki rezygnacji z praw własności intelektualnej do programu (tj. publikacji programu jako *public-domain*) z dodatkowymi klauzulami wymuszającymi informowanie nabywców produktów wykorzystujących dany program o jego oryginalnych autorach (co z kolei jest podobne do formuły „*attribution*” występującej w popularnych licencjach opracowanych dla treści nie będących oprogramowaniem komputerowym – Lawrence Lessiga Creative Commons¹⁰). Faktyczny rozwój oprogramowania objętego takimi licencjami zależy już zatem wyłącznie od dobrej woli instytucji,

¹⁰ Por. L. LESSIG, *Free Culture: How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*, Penguin Press, New York 2004.

które wykorzystują jego kod źródłowy – nie ma bowiem żadnego wymagania prawnego, które zmuszałoby je do upubliczniania ewentualnych poprawek. Daje to dobre efekty wtedy, gdy oprogramowanie wykorzystywane jest jako istotny komponent infrastruktury informatycznej przez kilka dużych podmiotów (np. korporacji), które nie konkurują ze sobą bezpośrednio, a na przykład powiązane są wertykalnymi zależnościami biznesowymi (rodzaju odbiorca-dostawca, czy też zleceniodawca-wykonawca). Wtedy wspólne opracowywanie oprogramowania, na jakie pozwala dostępność kodu źródłowego, jest atrakcyjne biznesowo (nie ma tu bowiem zagrożeń związanych z konkurencją), zaś permissywna licencja pozwala wykorzystać tak opracowane technologie we własnych, w pełni komercyjnych produktach. Przykładem takich jest rodzina narzędzi internetowych konsorcjum Apache, czy też system operacyjny OpenBSD, objęte wspomnianą już licencją BSD.

Wolne oprogramowanie jest fenomenem stosunkowo nowym, szczególnie na rodzimym, niedojrzałym jeszcze rynku usług i produktów informatycznych. Rozwiązania prawne licencji wolnego oprogramowania, takich jak wspomniana w niniejszym tekście licencja GPL, nie były jak do tej pory, poza nielicznymi wyjątkami¹¹ przedmiotem orzecznictwa. Dynamika rozwoju pakietów określanych przez swoich twórców jako wolne oprogramowanie, oraz coraz większe zainteresowanie tym modelem tworzenia i dystrybucji oprogramowania, przejawiane zarówno przez konsumentów jak też i korporacje biznesowe, wskazują jednak na to, iż wkrótce sytuacja ta ulegnie zmianie.

OPEN SOURCE SOFTWARE LICENCES

Summary

Computer software is regarded in popular culture as a typical consumer good. Legally, it constitutes however an example of a „content”, which

¹¹ Por. A. METZGER, *Die GPL – kommentiert und erklärt*, O'Reilly Verlag, Köln 2005, s. 23 i n.

is protected – in a way not dissimilar to literary or audio-visual works – by copyright law. An important property of these works, closely related to recent advances in computer science and resulting ease of digitization and duplication, is a possibility of costless duplication. This allows the creator of a digital work – for example a computer program – to distribute it at no cost, hence the cost of duplication itself is either minimal or even non-existent.

In recent years, an emergence of a new content distribution philosophy can be observed. Its proponents postulate, that it should be possible to create such computer software, that would be available for everyone to use, modify and redistribute – all that at no cost.

This paper introduces the concept of free software and discusses some legal aspects of its creation and distribution. An overview of the construction of the most popular free software licence, the GNU General Public License is also included.