Marcin Gogolewski, Michał Ren

Znaczenie wolnych i otwartych standardów dla bezpieczeństwa i rozwoju gospodarki elektronicznej

Ekonomiczne Problemy Usług nr 87, 243-251

2012

ZESZYTY NAUKOWE UNIWERSYTETU SZCZECIŃSKIEGO

NR 702 EKONOMICZNE PROBLEMY USŁUG NR 87 2012

MARCIN GOGOLEWSKI, MICHAŁ REN
Uniwersytet im. Adama Mickiewicza w Poznaniu

ZNACZENIE WOLNYCH I OTWARTYCH STANDARDÓW DLA BEZPIECZEŃSTWA I ROZWOJU GOSPODARKI ELEKTRONICZNEJ

Wprowadzenie

W niniejszym artykule chcielibyśmy przeanalizować wpływ wolnych i otwartych standardów na gospodarkę elektroniczną. Nasze rozważania przeprowadzimy, przedstawiając różne technologie, których stosowanie, choć czasami wydaje się głęboko uzasadnione, powoduje znaczące ograniczenie działania zasad wolnego rynku czy konkurencji, nie niosiąc z sobą jednocześnie istotnych zalet dla ogółu gospodarki. W trakcie tego rozważania podamy też konkretne przykłady, w których zastosowanie otwartych standardów okazało się bardziej korzystne lub wykorzystanie zamkniętych rozwiązań miało negatywne skutki dla rozwoju ekonomicznego. Przedstawiony problem jest bardzo podobny do sytuacji opisanych przez profesora L. Lessiga[1], przy czym w tym przypadku jego wpływ na gospodarkę ma dużo większy zakres.

1. Rozwiązania technologiczne

Zaprezentowany opis nie stanowi, choćby ze względu na ograniczone miejsce, wyczerpującego opracowania stosowanych technologii, jednak powinien być dość reprezentatywny ze względu na stosowane obecnie rozwiązania techniczne oraz prawne.

---

[1] L. Lessig, *Free Culture – How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*, 2004.

2. Cyfrowe zarządzanie prawami

Technologie znane jako DRM (*Digital rights management*) to właściwie spory zestaw różnych rozwiązań, które w pierwotnym swym założeniu miały służyć do ochrony praw autorskich[2]. Nie będziemy zajmować się w tym miejscu sensownością takiego rozwiązania ani rozważać, czyje prawa ono chroni. Opiszemy jedynie krótko samą technologię, podając problemy związane z jej stosowaniem (w większości zupełnie niezwiązane z prawem autorskim).

Ogólna idea wspomnianych technologii jest następująca: publikowana treść jest najczęściej szyfrowana z wykorzystaniem kryptografii asymetrycznej i w celu zwiększenia wydajności przetwarzania – symetrycznej[3] (zazwyczaj są to algorytmy RSA i AES[4]), tak by mogło ją odczytać jedynie odpowiednie licencjonowane urządzenie lub oprogramowanie posiadające klucz deszyfrujący. Już na pierwszy rzut oka widać następujące problemy:

- do odtworzenia danej treści potrzebna jest nie tylko licencja na nią, ale także odpowiednie urządzenie lub oprogramowanie (często ograniczone do jednego systemu operacyjnego!), wymaganie to nie jest uzasadnione technologicznie;
- producent nowego sprzętu lub oprogramowania musi uzyskać odpowiednią licencję, działania takie mogą służyć w przyszłości do ograniczenia konkurencyjności rynku;
- producent oprogramowania musi często wnieść stosowną opłatę, zatem nie będzie w stanie udostępnić swojego produktu za darmo (chyba że wykorzystuje ono do działania inne, licencjonowane oprogramowanie, co jednak w żadnym stopniu nie rozwiązuje pierwotnego problemu).

Poza wyżej wymienionymi problemami istnieje wiele innych, już nie tak oczywistych ani widocznych bez dokładnej analizy:

- treść może być traktowana jako usługa (tak w Polsce są klasyfikowane książki w formie elektronicznej – kupujemy *licencję* na dostęp do książki, a nie samą książkę, co pociąga za sobą m.in. obłożenie podatkiem VAT w wysokości 23%)[5];
- jeżeli producent licencjonowanego sprzętu lub oprogramowania złamie zasady ustanowione przez organizację zarządzającą certyfikacją, na przykład poprzez publikację własnego klucza, to przyszłe utwory publikowane za

---

[2] Ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych, DzU nr. 24, poz. 83, z pózn. zm.

[3] M. Kutyłowski, W. Strothmann, *Kryptografia. Teoria i praktyka zabezpieczania systemów komputerowych*, 1999.

[4] W. Stallings, *Network security essentials, applications and standards*, 2007.

[5] Ustawa z dnia 11 marca 2004 r. o podatku od towarów i usług, DzU nr. 54, poz. 535, z pózn. zm., art. 8 ust. 1.

pomocą danej technologii mogą nie działać na jego sprzęcie, także na tym wyprodukowanym wcześniej i będącym już w posiadaniu użytkownika końcowego;
- w każdym utworze (np. zabezpieczonym filmie na DVD), sprzęcie czy oprogramowaniu uiszczamy opłatę za opisywaną technologię.

Zdawać by się mogło, że pomimo powyższych problemów korzyści płynące ze stosowania DRM przewyższają niedogodności tą technologią wywołane. Jednak po chwili zastanowienia dochodzimy do wniosku, że jedynymi beneficjentami są w tym przypadku firmy oferujące rozwiązania DRM. Żadne z proponowanych zabezpieczeń (przynajmniej tych szeroko stosowanych) nie przetrwało próby czasu, a często zostało złamane jeszcze przed oficjalną premierą[6]. I choć często powoduje ono niepotrzebne problemy legalnych użytkowników końcowych (np. ustawienie regionu w czytniku DVD), to nie stanowi żadnej ochrony przed osobami o większym doświadczeniu, w szczególności przed zorganizowanymi grupami „piratów". Obchodzenie tego typu zabezpieczeń jest oczywiście w wielu krajach nielegalne, ale samo nieautoryzowane kopiowanie niezabezpieczonej treści także jest nielegalne, więc nie widać tu poprawy.

Nie uważamy (cytując B. Schneiera „Posiadając cyfrowy plik [...] można zrobić tyle jego kopii, ile się tylko chce [...]. To jest naturalne prawo cyfrowego świata [...]. Przemysł rozrywkowy usiłuje użyć technologii przeciw temu prawu. Chcą praktycznego sposobu na wystarczające utrudnienie kopiowania, by zachować istniejący model biznesowy. Ale są skazani na porażkę"[7]), że implementacja DRM jest absolutnie niemożliwa, jednak do pełnego działania tego typu technologii należałoby zrezygnować z takich „przywilejów", jak wolny rynek, czy nawet wolność słowa, bo choć obecnie nie jest możliwa na dużą skalę powtórka z historii powstania Hollywood (warto w tym miejscu wspomnieć, że rozkwit Hollywood jako centrum filmowego był w dużej mierze spowodowany chęcią ucieczki przed słonymi opłatami licencyjnymi za patenty wykorzystywane w produkcji filmów), to takie przyzwolenie na utratę wolności wydaje się bardzo mało prawdopodobne.

Nie oznacza to jednak wcale, że takie próby nie są podejmowane, jako przykład można tu podać DMCA – Digital Millennium Copyright Act[8], które odbiera obywatelom właśnie część ich wolności (można na przykład złamać prawo poprzez skopiowanie utworu, do którego mamy pełne prawa autorskie). Nie oznacza to

---

[6] A. Patrizio, *Why the DVD Hack Was a Cinch,* Wired, 1999.02.11, www.wired.com/science/discoveries/news/1999/11/32263.

[7] B. Schneier, *The Futility of Digital Copy Prevention,* CRYPTO-GRAM, May 15, 2001 http://cryptome.org/futile-cp.htm

[8] *Digital Millennium Copyright Act*, H.R.2281, 1997.

także braku działań mających na celu uniemożliwienie obchodzenia zabezpieczeń DRM[9].

Na koniec warto wspomnieć, że istnieją alternatywne technologie, takie jak *watermarking*[10], które choć nie zabezpieczają przed samym kopiowaniem, to umożliwiają często ustalenie sprawcy, nie powodując dodatkowych problemów charakterystycznych dla DRM. Nie umożliwiają one jednak dodatkowej kontroli, takiej jak np. ograniczenie dozwolonej liczby wyświetleń książki w formacie elektronicznym, czyli w praktyce kontroli istotnie pełniejszej, niż miało to miejsce przed powstaniem sieci Internet.

3. Wpływ otwartości kodu na bezpieczeństwo kryptografii

Przy projektowaniu bezpiecznych systemów jedną z podstawowych maksym jest zasada Kerckhoffsa[11] – system kryptograficzny musi być bezpieczny przy założeniu, że wszystkie szczegóły jego budowy są powszechnie znane, a jedynym sekretem jest klucz. Mogłoby się wydawać, że zachowanie opisu implementacji w tajemnicy zwiększa bezpieczeństwo, bo przecież żeby złamać zabezpieczenie, trzeba wiedzieć, jak ono działa. W niektórych przypadkach korzyści mogą przeważyć nad wadami, jednak należy pamiętać, że udostępnienie szczegółów budowy pozwala większej liczbie zainteresowanych na szukanie luk w bezpieczeństwie, również tym osobom, które nie mają środków na przeprowadzenie kosztownego procesu *reverse engineering*. Takie osoby często nie mają motywacji do skrytego wykorzystywania odkrytych luk, a wręcz przeciwnie – zgłaszają ich odkrycie, przyczyniając się do zwiększenia poziomu bezpieczeństwa. Konieczność zapewnienia pełnej dokumentacji budowy systemu utrudnia kiepskim rozwiązaniom przetrwanie na rynku.

Z drugiej strony, jeżeli ktoś odkryje lukę w zamkniętym produkcie, to może jej nie ujawnić choćby ze strachu przed procesem sądowym (były już takie przypadki). Wcale nie musi go to jednak powstrzymać przed ujawnieniem takiej informacji osobom zainteresowanym, np. włamaniem do systemu – można odnieść wymierne korzyści, a fakt zaistnienia takiej współpracy jest zwykle trudny do udowodnienia.

---

[9] J. Billington, *Statement of the Librarian of Congress Relating to Section 1201 Rulemaking*, 2010.

[10] I. Cox, M. Miller, J. Bloom, J. Fridrich, T. Kalker, *Digital Watermarking and Steganography*, 2007.

[11] B. Schneier, *Applied Cryptography*, 1996.

4. Otwartość kodu

Kolejną kwestią jest zatem dostępność kodu źródłowego. Sprawa ta często wywołuje wiele emocji, głównie wśród przeciwników takiego rozwiązania. Głównym argumentem jest najczęściej brak zabezpieczenia przed nieuprawnionym kopiowaniem. Wbrew jednak opinii takich osób problem piractwa bardzo rzadko dotyczy kodu źródłowego, a dużo częściej gotowych aplikacji i często właśnie brak otwartości kodu, w przypadku nieuprawnionego jego wykorzystania, uniemożliwia sprawdzenie, czy czyjeś prawa nie zostały naruszone.

Inną niewątpliwą przewagą posiadania kodu źródłowego nad wersją skompilowaną są większe możliwości testowania. Oczywiście większość użytkowników nigdy nie zajrzy do udostępnionego kodu, ale znajdą się tacy, którzy zgłoszą błędy lub prześlą ulepszenia aplikacji autorowi, by mógł on je włączyć do następnej wersji. Z kolei brak otwartości kodu powoduje, że kolejne pokolenia programistów, zamiast ulepszać dotychczasowy dorobek, cały czas na nowo wynajdują koło.

Brak dostępu do kodu źródłowego oznacza także istotne ograniczenie zasady wolnej konkurencji[12], gdyż podmiot zamawiający po okresie określonym w pierwotnej umowie jest praktycznie skazany na wybór tego samego dostawcy w celu dalszego utrzymania systemu pomimo tego, że poniósł wcześniej wszystkie koszty związane z jego powstaniem. W nowych zamówieniach coraz częściej umieszcza się wymóg przekazania całości kodu źródłowego zamawiającemu (nie potrzebuje on przejmować praw autorskich[13], wystarczy mu zwykle licencja na ulepszanie i modyfikację – w takim przypadku kod nie może być przekazany dalej inaczej niż podwykonawcy, w celu dalszego świadczenia wsparcia). Problem ten występuje także w przypadku ogólnodostępnych systemów, czyli takich, które nie powstały na zamówienie, a producent zadecyduje, że przestaje wspierać dany produkt. Jest to szczególnie dotkliwe dla systemów zależnych, pracujących na przykład pod kontrolą danego systemu operacyjnego, w których trzeba czasami wymieniać niektóre elementy sprzętowe. W wielu krajach nie jest zatem możliwe, w zgodzie z prawem, oferowanie wsparcia dla systemów, dla których nie oferuje go już producent.

Sama otwartość kodu nie jest gwarancją, że nie czyhają w nim niepożądane, trudne do spostrzeżenia pułapki. Od lat organizowane są konkursy tworzenia takiego kodu, który nie dość że przejawia niepożądane, niebezpieczne dla użytkownika działanie, to jeszcze jest lepiej oceniany, jeśli[14]:

– jest zwięzły i wydaje się łatwy do przeczytania;

---

[12] Ustawa z dnia 16 lutego 2007 r. o ochronie konkurencji i konsumentów, DzU nr 50, poz. 331.

[13] Ustawa z dnia 4 lutego 1994 r. o prawie autorskim…

[14] underhanded.xcott.com

- nie wzbudza dodatkowych podejrzeń przy czytaniu w środowisku oznaczającym elementy składni różnymi kolorami (*syntax coloring*);
- jego niepożądane działanie jest aktywne w każdym środowisku (a nie np. tylko na konkretnym rodzaju architektury procesora);
- niepożądane działanie można, po jego odkryciu, wyjaśnić jako niewinny błąd początkującego programisty.

Wprawdzie ostatnie zadanie było dość niewinne (napisać program zarządzania bagażem dla linii lotniczej, który może wysyłać bagaż na niewłaściwe lotnisko, jeśli komentarz wpisany przez pracownika będzie spełniał szczególne warunki), ale w przeszłości pojawiały się też np. implementacje algorytmów szyfrujących, które na różne sposoby dopuszczały do ujawnienia klucza bądź zawierały trudne do spostrzeżenia „usterki" umożliwiające rozszyfrowanie przesyłanego tekstu.

Przykładem systemu DRM, uważanego przez producenta za „niemożliwy do złamania", a którego złamanie okazało się trywialne po odkryciu sposobu działania, było szyfrowanie czcionek PostScript Type 1 firmy Adobe. Zostało ono wprowadzone w 1984 roku i prawie natychmiast złamane – Adobe wykorzystywało jedynie środki prawne, by wymusić licencjonowanie technologii. W końcu, po wprowadzeniu formatu TrueType w 1991 roku, sama firma Adobe opublikowała informację, jak złamać swoje szyfrowanie (wystarczy rozkaz: „1183615869 internaldict begin")[15].

Warto zauważyć, że problemy związane z utrzymywaniem pewnych informacji w tajemnicy są szkodliwe nie tylko w przypadku przestrzeni wirtualnej. Przykładem luki w bezpieczeństwie, z którą mamy do czynienia na co dzień, często nie zdając sobie z tego sprawy, jest brak odporności powszechnie używanych zamków do drzwi na tzw. *key bumping*. Do otwarcia nawet dobrej jakości zamka wystarczy czasami specjalnie przygotowany klucz (możliwy do wytworzenia za pomocą dowolnego klucza pasującego do danego typu zamka i zwykłego pilnika) oraz gumowy młotek. Samo otwarcie zamka jest tak szybkie jak otwieranie go właściwym kluczem i jeśli nawet zostawia na zamku ślady, to są one bardzo trudne do wykrycia[16]. Co gorsza, im lepszy zamek i im lepiej są do siebie dopasowane jego mechaniczne elementy, tym łatwiej atak przeprowadzić. Oczywiście istnieją konstrukcje zamka całkowicie niewrażliwe na próby otwarcia w taki sposób[17]; opisywany atak jest znany od ponad osiemdziesięciu lat[18]. Jednak dopiero spopularyzowanie problemu rozpoczęło proces upowszechnienia takich zamków na rynku.

---

[15] *Adobe Type 1 Font Format* partners.adobe.com/public/developer/en/font/T1_SPEC.PDF.

[16] B. Wels, R. Gonggrijp, *Bumping locks*, 2005.

[17] G. Pulford, *High Security Mechanical Locks – An Encyclopaedic Reference,* 2007.

[18] H.R. Simpson, *Lock Device*, US Patent 1667223, 1928.

5. Inne problemy związane z zamkniętymi standardami

Problem otwartości dotyczy oczywiście nie tylko kodu, ale także formatu zapisu plików. Niektóre z tych formatów, bądź ich istotne części, zostały opatentowane, jak na przykład GIF[19]. Inne przykłady to niektóre formaty wideo, które są albo chronione patentami, albo ograniczone licencją np. do konkretnego systemu (np. ASF). Ograniczenia takie są często wyrokiem śmierci dla aplikacji, które autor chciałby udostępnić za darmo.

Warto podać przykład innego zamkniętego protokołu, który przez wiele lat zmuszał prawie wszystkie firmy działające na terytorium Rzeczypospolitej Polskiej do posiadania licencji na system operacyjny konkretnego producenta. Chodzi oczywiście o Zakład Ubezpieczeń Społecznych i oferowany przez niego „za darmo" program do rozliczeń. Jako powód takiego rozwiązania podawano bezpieczeństwo (rozumiane przez tajność) protokołu, który jak się okazało, można w bardzo prosty sposób zaimplementować, wykorzystując otwarte i wolne oprogramowanie[20]. Co prawda po pięciu latach procesu (24 czerwca 2007 roku) sąd nakazał oficjalne odtajnienie specyfikacji, jednak nadal większość firm jest w praktyce zmuszona do używania pierwotnej wersji, gdyż ZUS nie wspiera alternatywnego oprogramowania, całą odpowiedzialność przenosząc na użytkownika[21].

Na zakończenie warto jeszcze wspomnieć o próbach wymuszenia uznawania patentów na „idee" typu „podwójne kliknięcie" czy proste formaty danych[22]. Działanie takie może spowodować, że prawie każdy program naruszałby wiele różnych patentów i nawet gdyby twórca poświęcił całe swoje życie na przeglądanie baz patentowych, to zarówno ze względu na liczbę już wydanych patentów, jak i na dużą swobodę w języku opisu nie byłby w stanie sprawdzić, jakie patenty narusza jego produkt.

Podsumowanie

Właściwie w każdym analizowanym przypadku widać pewną przewagę rozwiązań otwartych i wolnych nad pozostałymi. Warto zauważyć, że wolność i otwartość sama w sobie nie narusza i nie skłania do naruszania prawa autorskiego, a je-

---

[19] T. Welch, *High speed data compression and decompression apparatus and method*, US Patent 4558302, 1985.

[20] www.openssl.org

[21] P. Przybylski, *Stanowisko ZUS na temat możliwości przekazywania dokumentów ubezpieczeniowych z wykorzystaniem alternatywnych do programu Płatnik programów interfejsowych wytworzonych przez dowolnego producenta*, Baza wiedzy Zakładu Ubezpieczeń Społecznych, 2007.

[22] www.nosoftwarepatents.org

dynym bezpośrednim skutkiem jest promowanie lepszych, bardziej efektywnych rozwiązań i wzrost konkurencyjności, co powinno bezpośrednio przełożyć się na wzrost gospodarczy. W sytuacji takiej produkt gorszej jakości ma mniejsze szanse na przetrwanie, bo można ocenić nie tylko wygląd zewnętrzny, ale także sposób działania. Oczywiście otwartość skutkowałaby zmniejszeniem dochodów nielicznych firm, czerpiących bezpośrednie korzyści z dostarczania pewnych technologii (m.in. DRM), jednak stanowią one jedynie niewielki margines rynku.

Jako przykład na zakończenie można podać tendencję widoczną coraz bardziej na polskim rynku książek w formie elektronicznej, gdzie kolejne sklepy i wydawnictwa odchodzą od zabezpieczeń typu DRM na rzecz *watermarkingu*, notując jednocześnie wzrost sprzedaży. Co ciekawe, podobna zasada dotyczy także książek dostępnych zupełnie za darmo w Internecie, takich jak np. *Handbook of Applied Cryptography* A.J. Menezesa, P.C. Oorschota i S.A. Vanstone'a czy *Thinking in Java* B. Eckela.

Literatura

1. *Adobe Type 1 Font Format*, partners.adobe.com/public/developer/en/font/T1_SPEC.PDF
2. Billington J., *Statement of the Librarian of Congress Relating to Section 1201 Rulemaking*, 2010
3. Cox I., Miller M., Bloom J., Fridrich J., Kalker T., *Digital Watermarking and Steganography*, 2007.
4. *Digital Millennium Copyright Act,* H.R.2281, 1997.
5. Kutyłowski M., Strothmann W., *Kryptografia. Teoria i praktyka zabezpieczania systemów komputerowych*, 1999.
6. Lessig L., *Free Culture – How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*, 2004.
7. Patrizio A., *Why the DVD Hack Was a Cinch,* Wired, 1999.02.11, www.wired.com/science/discoveries/news/1999/11/32263.
8. Przybylski P., *Stanowisko ZUS na temat możliwości przekazywania dokumentów ubezpieczeniowych z wykorzystaniem alternatywnych do programu Płatnik programów interfejsowych wytworzonych przez dowolnego producenta*, Baza wiedzy Zakładu Ubezpieczeń Społecznych, 2007.
9. Pulford G., *High Security Mechanical Locks – An Encyclopaedic Reference*, 2007.
10. Schneier B., *Applied Cryptography*, 1996.
11. Schneier B., *The Futility of Digital Copy Prevention,* CRYPTO-GRAM, May 15, 2001 http://cryptome.org/futile-cp.htm
12. Simpson H.R., *Lock Device*, US Patent 1667223, 1928.
13. Stallings W. *Network security essentials, applications and standards*, 2007.

14. underhanded.xcott.com
15. Ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych, DzU nr 24, poz. 83, z późn. zm.
16. Ustawa z dnia 11 marca 2004 r. o podatku od towarów i usług, DzU nr 54, poz. 535, z późn. zm., art. 8 ust. 1.
17. Ustawa z dnia 16 lutego 2007 r. o ochronie konkurencji i konsumentów, DzU nr 50, poz. 331.
18. Welch T., *High speed data compression and decompression apparatus and method*, US Patent 4558302, 1985.
19. Wels B., Gonggrijp R., *Bumping locks*, 2005.
20. www.nosoftwarepatents.org
21. www.openssl.org

THE IMPORTANCE OF OPEN STANDARDS FOR SECURITY AND DEVELOPMENT OF DIGITAL ECONOMY

Summary

This article analyses the impact of free and open standards on digital economy. We present several technologies, and specific examples of cases where the application of open standards has proven more advantageous or where using closed standards had negative consequences for economic development. It is worth noting that freedom and openness do not by themselves lead to violations of any copyright laws, and their direct consequence is the promotion of better, more efficient, more competitive solutions, which in turn should positively influence economic growth.

Translated by Michał Ren