

Uruchamianie aplikacji windowsowych w programie Wine

WINE W PRAKTYCE

Warstwa zgodności Wine umożliwia użytkownikom Linuksa uruchamianie programów przeznaczonych dla systemu Windows. Niestety, konfiguracja Wine jest dość skomplikowana; dobrze więc, jeśli lubisz eksperymenty.

JOACHIM VON THADDEN

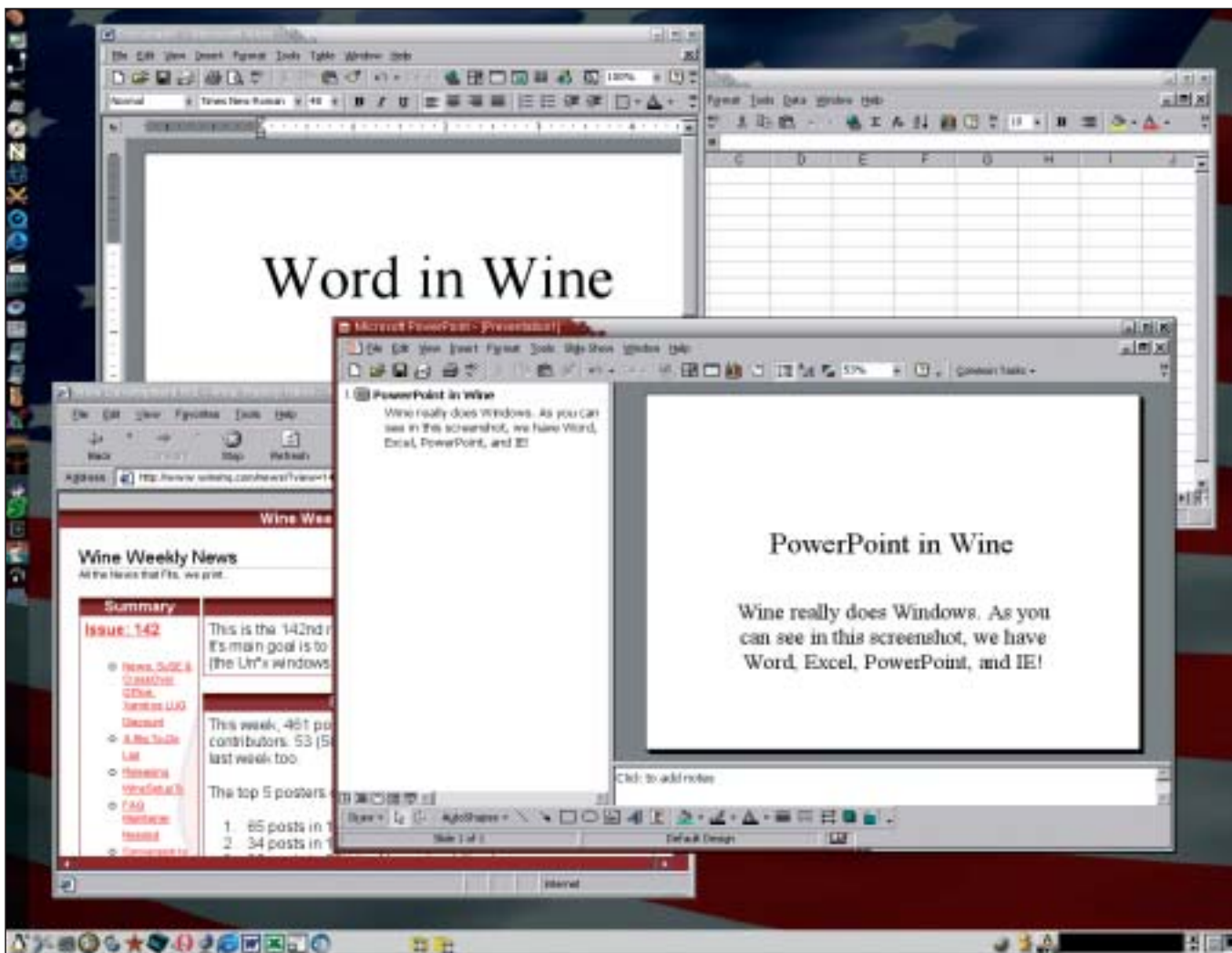
Użytkownicy Linuksa, chcący od czasu do czasu uruchomić program windowsowy, mają całkiem sporo możliwości. Mogą emulować cały system za pomocą aplikacji takich jak VMware czy Qemu, albo określony podsystem, korzystając z programu Win4Lin. Rozwiązania te sprawdzają się całkiem dobrze, mają jednak także parę wad: przede wszystkim wymagają posiadania emulatora i całego systemu operacyjnego – oba zaś zajmują cenne miejsce na dysku. Proces emulacji zmniejsza także wydajność i zajmuje dużo pamięci. Przeciętny program działający w emulatorze osiąga od 50 do 80 procent swojej nominalnej wydajności, nawet przy zwiększonym wykorzystaniu statycznej pamięci RAM przez emulowany system operacyjny.

Alternatywnym rozwiązaniem jest po prostu dostarczenie bibliotek windowsowych dla systemu Linux. Teoretycznie umożliwia to

uruchamianie danego programu w Linuksie; niestety, w praktyce nie jest to takie proste. Po pierwsze, funkcje bibliotek Windows oraz tzw. interfejs programowania aplikacji (Application Programming Interface – API) nie są odpowiednio udokumentowane. W wielu przypadkach programy (zwłaszcza autorstwa Microsoftu) wywołują funkcje, których nie ma w dokumentacji. API cechuje redundancja – można by przypuszczać, że Microsoft utracił kontrolę nad wieloma tysiącami wywołań systemowych i wyjątkowo skomplikowanymi parametrami wywoływania. Niektóre funkcje są ponadto dostosowane specjalnie do wymogów środowiska Windows i bazują na założeniach dotyczących tego właśnie systemu. Na przykład system plików w Linuksie różni się w ogromnym stopniu od systemu plików w Windows. Rozwiązaniem tego problemu byłaby możliwość używania oryginalnych bibliotek systemu Microsoftu.

Oznacza to oczywiście konieczność posiadania jego licencji; wielu użytkowników i tak jednak ją ma – dlaczego by więc nie wykorzystać jej do czegoś pożytecznego?

Rozpowszechniany na zasadzie Open Source projekt Wine oferuje funkcje programistyczne i umożliwia wykorzystanie oryginalnych bibliotek systemu Windows. Skrót „Wine”, oznaczający „Wine is not an emulator” („Wine nie jest emulatorem”) wyraża istotę tego programu: Wine nie jest emulatorem systemu operacyjnego, ale interfejsem umożliwiającym uruchamianie w jednym środowisku aplikacji przeznaczonych dla innego systemu (tzw. warstwą zgodności). Po długim okresie rozwoju i dzięki wsparciu przedsiębiorstw komercyjnych program ten pozwala na uruchamianie zaskakująco dużej liczby programów windowsowych pod Linuksem. W niniejszym artykule zajmiemy się praktyczną stroną korzystania z Wine.



Rysunek 1: Nawet bardzo złożone programy windowsowe, np. Microsoft Word, działają sprawnie w Linuksie.

Instalacja Wine

Ponieważ Wine rozwija się bardzo szybko, istotne znaczenie ma numer używanej wersji. Instalacja najnowszego wydania nie jest najlepszym rozwiązaniem; w zasadzie sugerowałbym przyjęcie konserwatywnego podejścia i trzymanie się działającej wersji. Lepiej nie eksperymentować zbyt wiele z nowymi systemami.

Sam też nie używam wersji najnowszej – mam wydanie 20041019. Numerem wersji pakietów Wine jest zawsze data ich udostępnienia. Wersja wydana w październiku 2004 r. jest wyjątkowo stabilna i ma wszystkie potrzebne mi funkcje. Kolejne wydania mają charakter eksperymentalny i nie nadają się do zastosowań produkcyjnych. Najnowsza wersja, 20050419, wygląda bardziej obiecująco, ale nie współpracuje z programem Microsoft Word.

Instalacja Wine jest na szczęście elementarnie prosta. Rozwlekłe procesy kompilacji i instalacji w przypadku większości dystrybucji należą już do przeszłości – na oficjalnej

stronie Wine znajdują się, stale aktualizowane i działające, pakiety dla Suse, Red Hata, Fedory, Debiana, Mandrake'a, Slackware'a, a nawet FreeBSD [1]. Ze strony [2] ściągnąć można także starsze pakiety. Po pobraniu danego pakietu użytkownicy dystrybucji opartych na RPM mogą po prostu zalogować się jako root i wpisać następujące polecenie

```
rpm -Uvh wine-200xxxxx-*
```

aby zainstalować Wine w swoim systemie. Jeżeli w instalacji przeszkadza poprzednia wersja programu, trzeba najpierw wpisać `rpm -e wine`, aby ją usunąć. W systemie Fedora odpowiedzialne za bezpieczeństwo funkcje mechanizmu Prelink, przechowywanego wstępnie skonsolidowane wersje wszystkich bibliotek systemowych, mogą spowodować parę problemów. Jeżeli nie chcesz go całkiem wyłączać, zaloguj się jako root, a potem – po zainstalowaniu Wine –

usuń powiązania z bibliotekami za pomocą następujących poleceń:

```
killall prelink;prelink -uav; $$  
rm /etc/prelink.cache; ldconfig
```

Ponieważ mechanizm odpowiadający za wstępną konsolidację sprawdza biblioteki raz na dobę, system powróci do stabilnego stanu najpóźniej po 24 godzinach. Jeżeli zależy Ci na czasie, możesz wywołać skrypt programu cron, *prelink*, aby uruchomić ten mechanizm ponownie:

```
killall prelink;$$  
/etc/cron.daily/prelink
```

Pakiety Wine dostarczane wraz z Debianem znane są z tego, że są bardzo nieprzewidywalne, a opieka nad nimi jest kiepska. W przypadku tej dystrybucji warto więc odinstalować wszystkie należące do niej pakie-

ty Wine, następnie zmienić repozytorium i zainstalować oryginalne pakiety pochodzące ze strony głównej programu, tak jak to opisano na stronie [3].

W przedśionku Wine – system katalogów

W Wine pliki konfiguracyjne i robocze oraz aplikacje Windows należące do danego użytkownika przechowywane są w katalogu użytkownika w podkatalogu *.wine*. Uważaj, jeżeli logujesz się jako *root*. Użytkownicy systemu Linux powinni wykorzystywać prawa *roota* tylko do wykonywania zadań administracyjnych; użytkownicy Wine także nie powinni używać uprzywilejowanych kont. Po pierwsze, bezpieczeństwo nie jest najsilniejszą stroną programów windowsowych, mogą one więc zagrozić całemu systemowi; po drugie, wywołania systemowe, których normalnie nie dałoby się uruchomić, mogą zadziałać dla użytkownika z prawami *roota* – zachowanie programów staje się wówczas nieprzewidywalne.

Do zakończenia instalacji niezbędny jest nowy katalog Wine; jeżeli takowy już istnieje, należy – ze względów bezpieczeństwa – zmienić jego nazwę. Utworzenie nowego katalogu jest bardzo proste – wystarczy wpisać *wine* w wierszu poleceń. Plik konfiguracyjny *config* i baza rejestru Windows, w której znajdują się pliki z rozszerzeniem *reg*, zostaną wówczas zapisane bezpośrednio w katalogu *~/wine*. Nowsze wersje Wine nie tworzą pliku *config*. Polecenie

```
cp /usr/share/doc/wine-20041019$$
/samples/config ~/.wine
```

odpowiada za skopiowanie pliku zawierającego domyślną konfigurację. Nawiasem mówiąc, Wine nie obsługuje oryginalnych plików rejestru Windows. Korzysta natomiast z plików w czystym formacie tekstowym, zapisując dane rejestru w plikach *system.reg* (rejestr systemowy) i *user.reg* (rejestr użytkownika). Ponieważ programy najczęściej nie korzystają z rejestru bezpośrednio, ale – tak jest to zalecane – poprzez wywołania systemowe, ułatwia to (zarówno użytkownikom, jak i systemowi) odczytywanie i modyfikowanie bazy rejestru.

Dowiązania znajdujące się w katalogu *~/wine/dosdevices* wykorzystywane są przez Wine do podawania liter napędów, których oczekuje system Windows. Dowiązania te noszą dokładnie takie nazwy, jakich używają programy windowsowe: *c:*, *d:* itd. Podka-

Listing 1: Plik config, część 1

```
[wine]
"GraphicsDriver" = "x11drv"; (x11drv, ttydrv)
"ShowDotFiles" = "1"
"ShowDirSymlinks" = "1"
"Path" = "c:\\windows;c:\\windows\\system"
"Windows" = "c:\\windows"
"System" = "c:\\windows\\system"
"Temp" = "t:\\\"
"Profile" = "c:\\windows\\Profiles\\Administrator"
```

talog *c* jest zatem odwzorowaniem napędu *C:* w drzewie linuksowego systemu plików – tu właśnie, po pierwszym uruchomieniu Wine, pojawia się typowy zbiór plików i podkatalogów systemu Windows. W katalogu *c/windows* znajduje się parę dowiązań symbolicznych takich jak *regedit.exe* czy *notepad.exe*; programy te są częścią pakietu Wine i powinny działać bez wprowadzania dodatkowych ustawień. Aby to sprawdzić, wpiszmy polecenie

```
wine notepad
```

które powinno uruchomić nasz pierwszy program w Wine - notatnik. Niektóre pakiety Wine instalują te programy jako dowiązania twarde w drzewie systemu plików Linuksa; aby uruchomić notatnik, wystarczy wówczas wywołać polecenie *notepad*.

Uproszczona konfiguracja

Za konfigurację Wine odpowiada plik *config*. Jego struktura jest całkiem prosta: podobnie jak windowsowy plik *.ini*, plik *config* zawiera nazwy grup w nawiasach kwadratowych; do każdej z nich przypisana jest para „*Klucz*” = „*Wartość*”. W odróżnieniu od Windows, zarówno klucze, jak i wartości należy zapisać w cudzysłowach; komentarze natomiast oznaczone są średnikami. Kolejność grup nie ma znaczenia.

Grupa *[wine]* (Listing 1) zawiera podstawowe parametry, odpowiedzialne za strukturę systemu plików i dostęp do zasobów systemowych Linuksa. Parametrom „*ShowDirSymlinks*” oraz „*ShowDotFiles*” można tu przypisać wartość 1 – umożliwi to wyświetlanie wszystkich plików należących do systemu Linux w menedżerze plików środowiska Windows. Ukrycie dowiązań symbolicznych może spowodować, że spędzisz wiele czasu na szukaniu plików. Wartości pozostałych parametrów można zostawić bez zmian.



Rysunek 2: Wine działa z wieloma różnymi aplikacjami dla Windows.

Część *[Version]* (Listing 2) pozwala modyfikować zachowanie bibliotek systemu Wine. Klucz „*Windows*” może przyjmować niemal dowolną wartość, oznaczającą wersję Windows: od Windows 3.0 do najnowszego Windows 2003. Ma to znaczenie do wielu aplikacji. Większość aplikacji działa jednak najlepiej, jeśli nadamy temu parametrowi wartość „*wine98*” – w przypadku tej wersji Windows API Wine funkcjonuje najsprawniej.

Listing 2: Plik config, część 2

```
[Version]
;Emulowana wersja Windows (win95,win98,winme,nt351,nt40,win2k,winxp,win2k3,win20,win30,win31)
"Windows" = "win98"
;"DOS" = "6.22"

[DllOverrides]
; Niektóre oryginalne biblioteki nie działają, należy więc użyć wbudowanych
; Nie należy modyfikować poniższych wierszy
"advapi32" = "builtin";Oryginalna biblioteka nie działa
"avicap32" = "builtin";Biblioteka sprzętowa
"capic2032" = "builtin";W pełni zaimplementowana
"comctl32" = "builtin";Wersja oryginalna powoduje błędy
"comdlg32" = "builtin";Konwersja przestrzeni adresowych (thunk)
"crtdll" = "builtin";W pełni zaimplementowana
"ctl3d32" = "builtin";Konwersja przestrzeni adresowych (thunk)
"d3d8" = "builtin";Biblioteka sprzętowa
"d3d9" = "builtin";Biblioteka sprzętowa
[...]
"msi" = "native"
"ole32" = "native"
"odbc32" = "native, builtin"

; Niektóre szpiegujące lub niedziałające programy, których nie chcemy uruchamiać
"*autorun.exe" = "native,builtin"
"*ctfmon.exe" = "builtin"
"*ddhelp.exe" = "builtin"
"eMusicClient.exe" = "builtin"
"*findfast.exe" = "builtin"
"icwconn1.exe" = "builtin"
;Zabezpieczenie przed ładowaniem ICW, nawet jeśli klucz rejestru się zmienił

; Wartości domyślne dla wszystkich innych bibliotek i plików wykonywalnych
"*" = "native, builtin"
;"*" = "builtin, native"
```

Oferujący najwięcej możliwości przełącznik znajduje się w części *[DllOverrides]* (wiersz 5 w Listingu 2). Tu określa się, które wbudowane biblioteki Wine będą używane, które zaś zostaną zastąpione ich oryginalnymi, windowsowymi wersjami. Ustawienia te można uporządkować według ważności: na przykład „*native, builtin*” oznacza, że Wine szukać będzie najpierw biblioteki oryginalnej. Jeżeli okaże się ona niedostępna, wykorzystana zostanie jej kopia należąca do Wine. Aby używać oryginalnych plików DLL, trzeba zapisać je w symulowanym drzewie windowsowego systemu plików w katalogu *windows/system* lub w katalogu zawierającym uruchamianą aplikację. Nie wszystkie pliki DLL są jednak naprawdę przydatne. Na przykład jądro systemu

Windows, system użytkowników i podsystem graficzny znajdują się w *kernel.dll*, *user.dll* i *gdi.dll* i nie da się ich używać w Wine. To samo dotyczy ich 32-bitowych odpowiedników: *kernel32.dll*, *user32.dll* i *gdi32.dll*. Także biblioteki niskopoziomowe, większość sterowników i pliki VXD są bezużyteczne.

Najlepszym wyborem są biblioteki należące do systemu Windows 98; niektóre z nich dostępne są w Internecie [4] lub w firmie Microsoft. Należy unikać bibliotek Windows XP, w zasadzie nigdy nie działają one bowiem bezbłędnie. Nawet w przypadku plików DLL z Windows 98 zalecana jest ostrożność. Niektóre z nich działają dobrze z wybranymi programami, w innych przypadkach lepiej sprawdzają się wersje nale-

żące do Wine. Wine może często wykorzystywać tak zwane formanty (pliki DLL odpowiedzialne za wyświetlanie elementów formularzy, takich jak pola wejściowe lub okna dialogowe), *commctrl.dll* i *commdlg.dll* oraz ich 32-bitowe odpowiedniki; to samo dotyczy bibliotek OLE, zapewniających funkcjonowanie portu COM bez zawieszania procesu instalacji oprogramowania. Ponieważ jest to tak bardzo skomplikowane, użytkownicy Wine mają możliwość określania ustawień dotyczących plików DLL w zależności od aplikacji; tym jednak zajmiemy się później.

WineHQ [5] zawiera uwagi dotyczące problemów związanych z bibliotekami i informacje dotyczące wielu popularnych bibliotek windowsowych. Stopień zaimplementowania bibliotek Wine można sprawdzić na stronie [6].

Niektóre elementy instalowane przez pakiet Office lub inne programy są naprawdę uciążliwe. Okienka programów debugujących, wyskakujące w momencie zawieszania się aplikacji, mogą zamienić życie w koszmar. Typowym przykładem jest program *mdm.exe* w katalogu *windows/system*. Aplikacje tego typu można na szczęście wyłączyć w części *DllOverrides*: na przykład wpis:

```
"*mdm.exe" = "builtin"
```

oznacza, że dany element wyszukiwany będzie w zbiorze bibliotek Wine. Ponieważ zaś program taki nie istnieje, nie da się go uruchomić – to proste. Grupa *[x11drv]* (Listing 3) opisuje własności podsystemu graficznego Wine, konwertującego API Windows na interfejs systemu X11 w Uniksie. Interesują nas ustawienia „*Managed*” i „*Desktop*”. Jeżeli pierwszemu z nich przypiszemy wartość „*Y*”, okna programów windowsowych obsługiwać będzie menedżer plików systemu X11; jeżeli zaś ustawienie to przyjmie wartość „*N*”, obsługą okien zajmie się Wine. Wykorzystanie menedżera okien prowadzi do lepszej integracji z okienkowym środowiskiem graficznym. Jeżeli za wygląd okien odpowiadać będzie Wine, będą one wyglądać dokładnie tak, jak w systemie Windows. Wiele programów działa wyłącznie w tym właśnie trybie. Ustawienie wartości domyślnej „*Y*” jest w większości przypadków najlepszym rozwiązaniem.

Parametr „*Desktop*” umożliwi uruchamianie danego programu w jego własnym oknie; można określić jego rozmiar, np. *800x600*.

Ma to sens, jeżeli dany program zawiesza środowisko graficzne albo chce koniecznie działać w tle. Tryb ten zaleca się także w przypadku gier, często przełączających tryby graficzne. Jeżeli gra się zawiesi, komputer może pozostać w wybranym trybie. Różne programy należące do rodziny Install Shield także często zajmują cały pulpit i przeszkadzają w uruchamianiu na nim jakichkolwiek innych programów. Niektóre części pliku konfiguracyjnego pozwalają na indywidualne konfigurowanie sterownika systemu X11 – zajmijmy się tym nieco później.

Inne ustawienia

Inne ustawienia w tej części sterują akceleracją grafiki, wartości domyślne powinny

więc tu wystarczyć. To samo dotyczy ustawień czcionek, portów we/wy i innych części pliku konfiguracyjnego. Pakiety instalacyjne same wprowadzają ustawienia typowe dla danej dystrybucji. Nieco więcej dzieje się w części „WinMM”, odpowiadającej za odtwarzanie dźwięku. Należy tu włączyć właściwy sterownik; w przypadku KDE jest to „winearts.drv”. Nie wszystkie sterowniki są dobrze zaimplementowane. Przypuszczalnie trzeba będzie uciec się do wykorzystania sterownika „wineoss.drv”; przed wprowadzeniem tego ustawienia należy wyłączyć możliwość odtwarzania dźwięku przez inne programy, aby uniknąć konfliktów. Dotyczy to zwłaszcza serwerów dźwięku w KDE i Gnome, *artsd* i *esd*.

Kolejne części odpowiadają za obsługę domyślnych konfiguracji dla poszczególnych aplikacji.

W sekcji `[AppDefaults\<program>\x11drv]` zdefiniowane są ustawienia wyświetlania dla konkretnego programu.

Opcjonalna część `[AppDefaults\<program>\DllOverrides]` umożliwia określenie domyślnych bibliotek; ustawienia te mają pierwszeństwo przed ustawieniami ogólnymi. Można dzięki temu uruchamiać Install Shield w osobnym oknie albo uniemożliwić programowi Quicktime Player 5 korzystanie z DirectDraw.

W dokumentacji zamieszczonej na stronie [7] znaleźć można znacznie więcej wskazówek na temat parametrów, konfiguracji i dostosowywania Wine do własnych potrzeb.

Zaawansowani użytkownicy wprowadzają najczęściej odpowiednie ustawienia ręcznie, edytując plik konfiguracyjny Wine. Początkujący wolą korzystać z wygodnego narzędzia z interfejsem graficznym. Program *winesetupk* umożliwia prostą edycję i tworzenie `.wine/config` oraz ustalanie najważniejszych wartości jego parametrów. Niestety, autorzy programu nie zajmują się nim od jakiegoś czasu, wykorzystuje więc ono parę przestarzałych opcji konfiguracyjnych. Trwają prace nad nowym narzędziem, *winecfg*, w czasie pisania tego tekstu nie było ono jednak ukończone. Do wyboru są więc następujące możliwości: konfiguracja ręczna, jedno z narzędzi bazujących na Wine lub Crossover Office.

Do startu, gotowi, hop!

Jednym z najistotniejszych poleceń przydatnych przy codziennym korzystaniu z Wine jest ponowne uruchamianie systemu Windows. Na szczęście dzięki Wine można uniknąć resetowania całego komputera. Wpisanie

```
wineboot
```

rozpocznie proces ponownego uruchamiania systemu na sposób windowsowy.

Wiele aplikacji korzysta z instalatora Install Shield. Niestety, Microsoft nie uznał dotychczas za stosowne ustalenia jednego schematu procesu instalacji – w Linuksie zaś nastąpiło to już jakiś czas temu, wraz z wprowadzeniem standardów takich jak DPKG i RPM. Podczas korzystania z Wine największe trudności powodują więc programy instalacyjne; mogą one także sprawiać problemy w przypadku bardziej złożonych aplikacji.

Listing 3: Plik config, część 3

```
[x11drv]
; Liczba kolorów przydzielanych z palety systemowej
"AllocSystemColors" = "100"
; Wykorzystanie własnej palety kolorów
"PrivateColorMap" = "N"
; Przy operacjach graficznych ważniejsza jest jakość wyświetlania niż szybkość
"PerfectGraphics" = "N"
; Używana głębokość kolorów
;;"ScreenDepth" = "16"
; Menedżer okien może zarządzać utworzonymi oknami
"Managed" = "Y"
; Uruchom pulpit Wine w oknie o rozmiarach 1024x768
;"Desktop" = "1024x768"
; Użyj rozszerzenia XFree86 DGA, jeśli jest
; (upewnij się, że masz prawa dostępu do /dev/mem!)
"UseDGA" = "N"
; Użyj rozszerzenia XVIDMode, jeśli jest
"UseXVIDMode" = "Y"
; Użyj rozszerzenia XRandR, jeśli jest
"UseXRandR" = "N"
...

[WinMM]
"Drivers" = "wineoss.drv" ; domyślne dla najczęściej spotykanych konfiguracji
;"Drivers" = "winearts.drv" ; dla KDE
;"Drivers" = "winealsa.drv" ; dla użytkowników ALSA
;"Drivers" = "winejack.drv" ; dla serwera dźwięku Jack
;"Drivers" = "winenas.drv" ; dla systemu dźwięku NAS
;"Drivers" = "wineaudioio.drv" ; dla komputerów z systemem Solaris
;"Drivers" = "" ;

[AppDefaults\_INS0432.\_MP\x11drv]
"Desktop" = "1024x768"

[AppDefaults\QuickTimePlayer.exe\DLLOverrides]
"ddraw" = ""
```

Jeżeli trafisz na program, którego za nic w świecie nie da się zainstalować, spróbuj zainstalować go najpierw w Windows 98. Wcześniej zrób koniecznie kopię zapasową rejestru, albo, jeszcze lepiej, wyeksportuj go. Po zainstalowaniu danego programu skopiuj jego folder do odpowiedniego podkatalogu w katalogu `~/wine`,

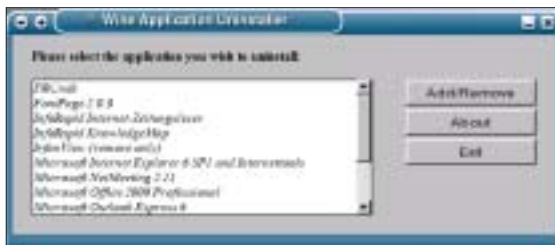
ponownie wyeksportuj rejestr Windows i porównaj wyniki. Następnie uruchom narzędzie `regedit` w Wine, aby zmodyfikować zmienione klucze. W większości przypadków w wyniku instalacji pojawiają się również nowe biblioteki – je także należy skopiować do katalogu Wine. Przy odrobinie szczęścia instalowany program powinien działać. Jeszcze użyteczniejsze są programy niekorzystające z kluczy rejestru – na przykład Lotus Notes. Instalatora tego programu nie da się wprawdzie uruchomić w Wine, można jednak zainstalować Lotus Notes pod Windows i po prostu skopiować wersję 6.51 tej aplikacji w odpowiednie miejsce w drzewie systemu plików Wine.

Programy, które do instalacji lub późniejszego działania wymagają włożenia do napędu płyty CD lub dyskietki, mogą przysporzyć trudności niedoświadczonym użytkownikom. Oczywiście, należy wówczas podłączyć napęd CD w Linuksie; trzeba jednak zabezpieczyć katalog montowania przed dostępem ze strony programu instalacyjnego – inaczej napęd zostałby zablokowany, co uniemożliwiłoby odmontowanie go przy zmianie dysku. Jeżeli masz tego świadomość, zareagowanie na żądanie zmiany dysku nie sprawi Ci kłopotu: wystarczy wówczas odmontować napęd CD z wiersza poleceń, zmienić płytę i podmontować napęd ponownie.

Praktyka wykazuje, że niektóre programy działają bez problemów tylko wówczas, gdy uruchamia się je w ich katalogach. Zdarza się również, że w funkcjonowaniu programów przeszkadza zmiana wirtualnych pulpitów; mogą się wówczas zawiesić i wymagać długiej reanimacji. Oznacza to często konieczność uruchomienia danej aplikacji ponownie. Niektóre programy nie zamykają się tak jak powinny i pozostawiają w pamięci działające składniki; aby uzyskać więcej szczegółów, należy posłużyć się poleceniem `ps -ax |grep wine`.

```
wineserver -kill
```

kończy wszystkie bieżące instancje Wine.



Rysunek 3: Moduł Uninstaller wyświetla zbiór narzędzi do deinstalacji.

Utrudnieniem są także aplikacje okupujące cały ekran i robiące na nim bałagan. Najlepiej, jak już wspomniano, uruchamiać je w osobnym oknie: wystarczy w tym celu usunąć znacznik komentarza w wierszu „Desktop” = „1024x768”. Jeszcze lepszym rozwiązaniem jest utworzenie w pliku konfiguracyjnym specjalnej, poświęconej danej aplikacji sekcji, odpowiadającej za uruchamianie sprawiającego trudności programu w oknie.

Control Center ułatwia zarządzanie programami skonfigurowanymi za pomocą ustawień systemowych Windows. Wpisanie

```
wine control
```

uruchamia to narzędzie i udostępnia zainstalowane w systemie składniki konfiguracyjne. Programy odinstalować można również korzystając z Control Center: polecenie

```
wine uninstaller
```

uruchamia aplet, który wyświetla zbiór zarejestrowanych programów służących do deinstalacji. Obsługa programów DOS-owych jest w Wine ograniczona. Instrukcja

```
wine wcmd
```

wywołuje interpreter poleceń, w którym można uruchamiać programy przeznaczone

Co dalej

Przed wykorzystywaniem Wine w praktyce należy zatroszczyć się o parę kwestii. Po pierwsze, warto mieć świadomość, że w programie tym da się uruchomić bez problemów tylko 30% wszystkich aplikacji windowsowych. Od systemu nie można wymagać zbyt wiele – nie ma on być zamiennikiem Windows; jeżeli szukasz czegoś takiego, oczywistą odpowiedzią jest Linux, pozwalający na korzystanie z zaskakująco wielu użytecznych i bezpłatnych programów. Wine daje szansę uruchamiania ważnych aplikacji windowsowych, nie mających zamienników w Linuksie – umożliwia w ten sposób stopniową przesiadkę z jednego systemu na drugi. Jeżeli intensywnie używasz programów należących do pakietu Office, docenisz niewątpliwie możliwość używania ich nadal i jednocześnie testowania nowych rozwiązań. Godnymi uwagi rozwiązaniami są komercyjne emulatory, Cedega (przeznaczona dla gier) i CrossOver Office (przeznaczony dla programów z rodziny Office) – programy te rozwijają się szybko, a ich użytkownicy mogą korzystać z pomocy technicznej. Trudno znaleźć godnych konkurentów tych systemów.

dla systemu DOS; nie należy jednak liczyć na zbyt wiele. Do uruchamiania programów DOS-owych lepiej użyć `dosemu` czy `dosbox` [8]. Dotyczy to zwłaszcza gier.

Co będzie działać, a co nie

Wine udostępnia warstwę emulacji dla plików binarnych należących do innego systemu operacyjnego. Wiąże się to z paroma ograniczeniami, sprawiającymi, że niektóre

Listing 4: Alternatywne ścieżki uruchamiania programu

```
wine "c:\program files\microsoft office\office\winword.exe"
wine "~/wine/c/Program Files/Microsoft Office/Office/WINWORD.EXE"
cd ~/.wine/c/Program Files/Microsoft Office/Office; wine WINWORD.EXE
cd ~/.wine/c/Program Files/Microsoft Office/Office; wine "c:\program files\microsoft office\office\winword.exe"
```

Inne możliwości: CrossOver Office i Cedega

System Wine dał początek wielu najpopularniejszym produktom komercyjnym, bazującym na kodzie Wine uzupełnionym o wyspecjalizowane funkcje. Specjalnością pakietu CrossOver Office firmy CodeWeavers [13] jest obsługa najważniejszych aplikacji Microsoft Office – trzeba przyznać, że autorzy tego programu wykonali kawał dobrej roboty. Dzięki temu, że udostępniłi napisane przez siebie poprawki i łaty autorom Wine, projekt ten bardzo się rozwinął w ciągu ostatniego roku. CrossOver Office zawiera najnowsze biblioteki Wine, ma bardzo łatwy w obsłudze program instalacyjny dla obsługiwanych aplikacji, jest w pełni zintegrowany ze środowiskami graficznymi, a firma CodeWeavers zapewnia pomoc techniczną dla programu. Ponieważ program nie jest drogi (kosztuje 40 dolarów lub 40 euro), niewygodny i problemy związane z testowaniem nowych wersji Wine, długie maratony instalacyjne i godziny poświęcone na czytanie dokumentacji w Sieci mogą stać się przeszłością.

Program Cedega firmy Transgaming [14] wykorzystuje Wine jako platformę do uruchamiania gier i rozwinął znacznie moduły odpowiedzialne za emulację DirectX. Specjalny kod umożliwia uruchamianie gier zabezpieczonych przed kopiowaniem. Program dostępny jest w wersji subskrypcyjnej w cenie 5 dolarów (lub 5 euro) miesięcznie. Minimalny okres subskrypcji to trzy miesiące. Kod źródłowy programu (bez modułów odpowiedzialnych za uruchamianie programów zabezpieczonych przed kopiowaniem) jest publicznie dostępny – oznacza to, że użytkownicy mogą sami skompilować. Gotowe pakiety instalacyjne przeznaczone dla różnych dystrybucji dostępne są na stronie [15].

aplikacje stają się bezużyteczne lub nieprzydatne w praktyce. Nie da się na przykład uruchomić programów wymagających dedykowanych sterowników sprzętowych. W rzeczywistości nawet próba zainstalowania sterownika sprzętowego przeznaczonego dla Windows jest skazana na porażkę, gdyż oba systemy zupełnie inaczej obsługują sprzęt. Niektóre programy wymagające bezpośredniego dostępu do dysku twardego lub napędu CD-ROM będą mimo to funkcjonować poprawnie. Wine oferuje w tym celu tryb bezpośredniego dostępu – działa dzięki niemu nawet tak nieprawdopodobny kandydat, jak Nero Burning Rom. CloneCD ma własne sterowniki umożliwiające dostęp do napędu – niestety, nie działają one w Wine. Niezależnie od tego, jaki program chcesz uruchomić, warto przejrzeć bazy aplikacji na stronach [9] i [10], aby sprawdzić, czy ktoś już tego wcześniej nie próbował i czy nie udostępnił raportu z instalacji albo odpowiednich wskazówek.

Gry bazujące na DirectX mogą działać lub nie. Więcej informacji na ten temat znaleźć można w Internecie. Wine emuluje zachowanie DirectX 8 – daje to więc szansę uruchamiania gier wykorzystujących tę technologię. Minusem Wine jest to, że nie da się w nim uruchomić żadnego programu zabezpieczonego przed kopiowaniem. Mechanizmy zabezpieczeń są tak niskopoziomowe, że próby ich obejścia z definicji ska-

zane są na porażkę. Niezależnie od tego, na co się zdecydujesz, nie instaluj oryginalnego DirectX. Nie dość, że nie zadziała, ale może również całkowicie zniszczyć aktualną instalację Wine.

Każdy użytkownik Wine zapozna się prędzej czy później z debuggerem *winedbg*. Uruchamia się on za każdym razem, kiedy działająca w Wine aplikacja zawiesi się. Jeżeli nie interesuje Cię, co i dlaczego się stało, wystarczy wyjść z programu za pomocą polecenia *quit*. Jeżeli wolisz przeanalizować dany problem, wpisz *bt*, aby wyświetlić dziennik śledzenia (backtrace).

W przypadku programów sprawiających trudności najlepszym rozwiązaniem jest wypróbowanie najpierw różnych ustawień wersji windowsowej. Niektóre programy są bardzo wybredne, jeśli chodzi o ścieżkę, z której są uruchamiane. Można więc spróbować uruchomić daną aplikację w jej własnym katalogu, podać pełną ścieżkę windowsową albo spróbować wpisać jej ścieżkę w systemie Linux (Listing 4). Przykładowe ścieżki są teoretycznie równoważne, program przeznaczony dla Windows może jednak przestrzegać to inaczej.

Wybór menedżera okien także może spowodować parę problemów. Zmiana ustawień *Managed* i *Desktop* może w tym przypadku uczynić cuda, zwłaszcza jeśli program działa, ale nie udaje się poprawnie wyświetlić jego okna.

Warto także podać parametry debuggowania podczas uruchamiania programu: `WINEDEBUG=„+loaddll” wine program_path` uruchomi program i określi, czy używa on oryginalnych czy wbudowanych bibliotek. Umożliwi to zidentyfikowanie plików DLL, które są niewłaściwie zaimplementowane w Wine i zastąpienie ich oryginalnymi odpowiednikami windowsowymi, co z kolei – miejmy nadzieję – pozwoli uruchomić dany program. Dokumentacja systemu śledzenia (debuggingu) w Wine [11] podaje wiele dodatkowych parametrów, umożliwiających wpisywanie informacji na temat określonych elementów środowiska. Archiwa listy dyskusyjnej Wine-Users [12] zawierają wiele pytań i odpowiedzi poświęconych problemom z Wine. Nowe pytania są oczywiście także mile widziane. ■

INFO

- [1] „Wine Headquarters” – strona programu Wine: <http://www.winehq.org/site/download>
- [2] Wine do pobrania: <http://prdownloads.sourceforge.net/wine>
- [3] Pobieranie pakietów dla Debiana: <http://www.winehq.org/site/download-deb>
- [4] Pliki DLL dla Windows: <http://www.dll-files.com>
- [5] Informacje o plikach DLL: <http://www.winehq.com/site/docs/wine-devel/arch-dlls>
- [6] Stan implementacji bibliotek: http://www.winehq.com/site/status_dlls
- [7] Dokumentacja Wine: <http://www.winehq.org/site/documentation>
- [8] Dosbox: <http://dosbox.sourceforge.net>
- [9] Aplikacje działające w Wine: <http://ap-pdb.winehq.org>
- [10] Frank's Corner, inna strona o Wine: <http://frankscorner.org>
- [11] Dokumentacja systemu śledzenia w Wine: <http://winehq.org/site/docs/wine-user/x1824#AEN1826>
- [12] Fora poświęcone Wine: <http://www.winehq.org/site/forums>
- [13] Crossover Office: <http://www.codeweavers.com>
- [14] Cedega firmy Transgaming: <http://www.transgaming.com>
- [15] Pakiety instalacyjne firmy Transgaming: <http://www003.portalis.it/115/>