

IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK

DODAJ DO KOSZYKA

CENNIK I INFORMACJE

ZAMÓW INFORMACJE
O NOWOŚCIACH

ZAMÓW CENNIK

CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

100 sposobów na Linux

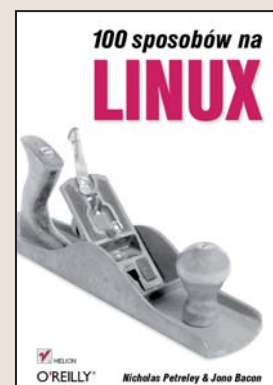
Autorzy: Nicholas Petreley, Jono Bacon

Tłumaczenie: Paulina Krzemińska

ISBN: 83-246-0065-5

Tytuł oryginału: [Linux Desktop Hacks](#)

Format: B5, stron: 316



Linux przeszedł długą drogę — od środowiska tekstowego, którego obsługa była umiejętnością graniczącą z wiedzą magiczną, aż do wygodnego i stabilnego systemu operacyjnego wyposażonego w doskonały interfejs graficzny. Ilość oprogramowania dostępnego dla Linuksa nieodpłatnie ciągle się zwiększa, a narzędzia w niczym nie ustępują swoim komercyjnym odpowiednikom przeznaczonym dla systemu Windows. Ogromną zaletą Linuksa jest jego elastyczność — przy odrobinie wiedzy można skonfigurować go tak, aby dokładnie odpowiadał naszym potrzebom. Wystarczy dodać mechanizmy obsługi odpowiednich urządzeń i usunąć moduły, które są zbędne.

„100 sposobów na Linux” to doskonałe źródło wiedzy dla tych użytkowników Linuksa, którzy chcą uczynić pracę z nim jeszcze bardziej komfortową. Z tej książki dowiesz się, jak w prosty sposób zmodyfikować Linuksa, aby lepiej służył Twoim celom. Znajdziesz tu wskazówki, dzięki którym udoskonalisz i dostosujesz do swoich potrzeb interfejs użytkownika i zwiększysz wydajność systemu. Nauczysz się sprawnie administrować środowiskiem i wydobędziesz z X, KDE, GNOME i konsoli wszystko, co mają do zaoferowania.

- Konfiguracja programu rozruchowego
- Wykonywanie zrzutów ekranu z wiersza poleceń
- Zmiana okien logowania
- Instalacja animowanych kursorów myszy w X Window
- Zdalny dostęp do pulpitu i programów
- Tworzenie skryptów dla KDE
- Przyspieszanie procesu uruchamiania aplikacji w Gnome
- Szyfrowanie poczty
- Podłączenie odbiornika GPS do Linuksa
- Konfigurowanie firewalla
- Rekompilacja jądra
- Podłączanie nowych urządzeń

Chcesz, by Linux pracował dla Ciebie?

Koniecznie przeczytaj tę książkę

Wydawnictwo Helion
ul. Chopina 6
44-100 Gliwice
tel. (32)230-98-63
e-mail: helion@helion.pl



Spis treści

Twórcy	7
Wstęp	11
Rozdział 1. Uruchamianie Linuksa	19
1. Uruchamianie komputera	19
2. Likwidowanie i odzyskiwanie głównego rekordu rozruchowego (MBR)	23
3. Pomińnięcie menedżera rozruchu	25
4. Ustawianie obrazka dla programu LILO	27
5. Tworzenie własnego ekranu powitalnego w LILO	29
6. Wyświetlanie tła w programie rozruchowym GRUB	34
7. Tworzenie ekranu powitalnego dla programu GRUB	34
8. Upiększanie procesu uruchamiania Debiana	36
9. Grafika w konsoli	41
Rozdział 2. Konsola	45
10. Przypisanie akcji do klawiszy	45
11. Pobudzanie konsoli programem Macro Music Magic	51
12. Robienie zrzutów ekranu z wiersza poleceń	54
13. Odchudzanie znaku zachęty wiersza poleceń	56
14. Upraszczanie zmiany katalogów	59
15. Kolorowanie nazw plików w programie stronicującym	61
Rozdział 3. Menedżery logowania	67
16. Szybkie przełączanie między użytkownikami	68
17. Podwojenie ekranów logowania KDM	69
18. Podwojenie ekranów logowania GDM	70
19. Włączanie trybu wieloużytkownikowego z wiersza poleceń	72
20. Logowanie programem Qingy	74
21. Zmienianie motywu programu Qingy	81

Rozdział 4. Sposoby związane z systemem graficznym X	85
22. Zaczernianie ekranu	86
23. Ożywienie pulpitu animowanymi kursorami myszy	87
24. Konwersja kursorów XP do Linuksa	92
25. Użycie czcionek z systemu Windows i Macintosha	94
26. Jak nie przeoczyć przypomnienia	97
27. Wyświetlanie komunikatów w odpowiedzi na zdarzenia	102
28. Włączanie klawiszy szybkiego dostępu za pomocą programu Hotkeys	104
29. Włączanie klawiszy szybkiego dostępu za pomocą programu LinEAK	108
30. Dostęp do systemów Windows i Mac OS X z Linuksa	112
31. Udostępnianie pulpitu przez internet	115
32. Zdalny dostęp do programów	116
33. Dodanie głębi do pulpitu	119
34. Rentgen pulpitu	122
Rozdział 5. Środowisko graficzne KDE	125
35. Konqueror oknem na świat	125
36. Dostęp do zdalnych systemów bez hasła	128
37. Gadżety na pulpit z programu karamba	130
38. Nietypowe sposoby uruchamiania aplikacji	134
39. Sposoby skryptowe z DCOP	136
40. Tworzenie własnych akcji w menu kontekstowym w KDE	138
41. KDE jeszcze łatwiejsze w użyciu	143
42. Dodanie głębi do okien KDE	147
43. Ograniczenie KDE w trybie kiosk	150
44. Uruchomienie najnowszej wersji KDE	153
Rozdział 6. Sposoby na GNOME	155
45. Losowanie tapety	155
46. Upiększanie pulpitu GNOME apletami gDesklets	157
47. Dodawanie własnych pozycji do menu uruchamianego za pomocą prawego klawisza myszy	159
48. Kompilacja najnowszej wersji GNOME	162
Rozdział 7. Możliwości terminala	165
49. Współdzielenie aplikacji i monitorów za pomocą programu screen	166
50. Ustalanie domyślnego wyglądu terminali	169
51. Przezroczystość terminala	170
52. Wyświetlanie dokumentów Worda w terminalu	173
53. Wyświetlanie dokumentów PDF w terminalu	175
54. Wyświetlanie plików Worda i PDF w programie Mutt	176

Rozdział 8. Programy w środowisku graficznym	177
55. Krótszy czas uruchamiania programu OpenOffice.org	177
56. Czytanie wiadomości z Yahoo!Mail w dowolnym programie pocztowym	180
57. Szyfrowanie poczty	182
58. Odbieranie poczty programem procmail	187
59. Konwersja skrzynki pocztowej	190
60. Konfiguracja przeglądarki Firefox za pomocą ukrytych opcji	192
61. Eliminacja opóźnień w przeglądarkach	194
62. Instalacja modułów rozszerzających w przeglądarkach	196
63. Uruchamianie telefonu internetowego	198
64. Wykrywanie ruchu i przysyłanie obrazu przez internet	200
65. Automatyczne publikowanie zrzutów ekranu w internecie	203
66. Wyszukiwanie sieci bezprzewodowych	205
67. Mapa świata rzeczywistego	208
68. Połączenie się z PPTP VPN Microsoftu	210
69. Odtwarzanie zastrzeżonych formatów multimedialnych	213
 Rozdział 9. Administracja i automatyzacja	 215
70. Automatyzacja zadań za pomocą programu cron	215
71. Aktualizowanie zegara przez internet	218
72. Automatyczne uruchamianie aplikacji	220
73. Sztuczne podtrzymywanie pracy programów	223
74. Klonowanie instalacji Linuksa	226
75. Przekierowanie portów przez SSH	229
76. Sterowanie ustawieniami nowego użytkownika	231
77. Powiadamianie użytkownika o zdarzeniach systemowych za pomocą poczty internetowej	233
78. Logowanie bez hasła	235
79. Usprawnienie sieci przewodowej	237
80. Ochrona przed programami systemu Windows	238
81. Konfiguracja firewalla	243
82. Monitorowanie parametrów połączeń za pomocą programu Wavemon	245
83. Tworzenie kopii zapasowych przez sieć	247
84. Odzyskiwanie Debiana	249
85. Poprawienie wydajności systemu programem prelink	251
86. Pobieranie najnowszego kodu źródłowego	252
87. Przyspieszenie kompilacji	255

Rozdział 10. Jądro	259
88. Kompilacja jądra	259
89. Aktualizacja jądra do wersji 2.6	263
90. Użycie jądra CKO w celu zwiększenia wydajności środowiska graficznego	266
91. Poprawienie osiągnięć jądra bez ponownej kompilacji	269
Rozdział 11. Sprzęt	273
92. Połączenie z internetem przez Bluetooth i telefon komórkowy	273
93. Projekt Utopia — wygodne korzystanie z urządzeń USB	275
94. Poprawienie częstotliwości odświeżania	278
95. Drukowanie na nieobsługiwanych	280
96. Zarządzanie energią za pomocą ACPI	284
97. Użycie urządzenia iPod w Linuksie	286
98. Synchronizacja urządzenia iRiver z Linuksem	292
99. Zwiększanie wydajności dysku twardego	294
100. Przyspieszenie działania gier	298
Skorowidz	301

Sposoby związane z systemem graficznym X

Sposoby 22. – 34.

Sposoby przedstawione w tym rozdziale nie są przypisane do konkretnego środowiska graficznego lub menedżera okien. W większości przypadków potrzebny jest po prostu system graficzny X11.

Tematyka tego rozdziału rozciąga się od prostej zmiany koloru tła widocznego podczas uruchamiania środowiska graficznego po zmianę wskaźnika myszy na animowane grafiki. Opisane także będą: sposób wykorzystania dodatkowych klawiszy klawiatur multimedialnych do obsługi internetu i multimediiów w systemie X11, sposób uzyskania zdalnego dostępu do programów (nawet uruchomionych na innych platformach) i sposób na dodanie efektów 3D i przezroczystości do okien.

System graficzny X11, nazywany również X lub X Window, jest silnikiem graficznym typu klient-serwer, który uruchamia większość środowisk graficznych i menedżerów okien w Linuksie. Prawie wszystkie dystrybucje Linuksa zawierały pakiet XFree86, będący darmową implementacją X11. Jednak twórcy XFree86 zmienili jego licencję, co ostatecznie doprowadziło do powstania nowej gałęzi kodu XFree86 i nowej wersji X11, zwanej Xorg lub X.org. Poza licencją nowa wersja ma wiele zalet. Ma wbudowaną obsługę przezroczystości i rzucania cienia, w wielu przypadkach zaproponowano również lepsze możliwości rozwiązywania problemów niż XFree86. W większości dystrybucji zrezygnowano już z XFree na rzecz Xorg lub taka zmiana jest planowana. Jedynym wyjątkiem jest dystrybucja Debian.

Większość sposobów przedstawionych w tym rozdziale nie wymaga środowiska Xorg. Wyjątkiem są sposoby, które wywołują największe wizualne wrażenie, jak przezroczystość i cienie. Sposoby opisujące zmianę wskaźnika myszy nie wymagają Xorg, wystarczy XFree86 w wersji co najmniej 4.3.0.

Używając Xorg, należy pamiętać, że w zależności od środowiska, w którym sposoby są uruchamiane, system może być mniej lub bardziej stabilny. Problemy te mogą wystąpić przede wszystkim w sposobach opisujących przezroczystość i cienie. Jest to pierwsza próba dodania tych funkcji do systemu X i istnieją pewne niedociągnięcia. Programiści tworzący Xorg nadal poprawiają i rozwijają te funkcje, należy jednak być przygotowanym na to, że opisane sposoby zablokują jeden menedżer okien, a w innym będą działać normalnie.



SPOSÓB

22.

Zaczernianie ekranu

Pozbycie się brzydkiej szarej siatki, która wita użytkownika otwierającego sesję X11

Podczas uruchamiania sesji X11 domyślnym tłem jest szara siatka. Niektóre menedżery okien i środowiska graficzne zmieniają tło prawie natychmiast, jednak w innych brzydka siatka towarzyszy procesowi uruchamiania dosyć długo. Przedstawiona poniżej wskazówka pozwala zmienić domyślne tło na jednolity czarny kolor. (Nie ma możliwości ustawienia innego koloru niż czarny).

Aby ustawić czarne tło, należy dodać przełącznik `-br` do odpowiedniego pliku konfiguracyjnego. To, który plik ma zostać zmieniony, zależy od sposobu uruchamiania menedżera okien lub środowiska graficznego.

Polecenie startx

Jeśli menedżer okien jest uruchamiany poleceniem `startx`, należy zmienić plik `/usr/X11R6/bin/startx`. Trzeba się upewnić, że domyślne argumenty uruchamiające serwer X zawierają przełącznik `-br`. Przykładowo — początek pliku `startx` może wyglądać następująco:

```
userclientrc=$HOME/.xinitrc
userserverrc=$HOME/.xserverrc
sysclientrc=/etc/X11/xinit/xinitrc
sysserverrc=/etc/X11/xinit/xserverrc
defaultclientargs=""
defaultserverargs="-nolisten tcp"
clientargs=""
serverargs=""
```

Do definicji `defaultserverargs` lub `serverargs` należy dodać przełącznik `-br`. Wystarczy dopisać przełącznik między znaki cudzysłowu, na przykład:

```
defaultserverargs="-nolisten tcp -br"
```

Od tej pory, po wydaniu polecenia `startx` uruchamiającego menedżera okien, domyślne tło będzie czarne.

Polecenie xinit

Jeśli menedżer okien jest uruchamiany poleceniem `xinit`, należy dodać przełącznik `-- -br` na końcu polecenia. Na przykład, aby uruchomić menedżer okien `blackbox` poleceniem `xinit`, należy wpisać:

```
$ xinit /usr/bin/blackbox -- -br
```

Jeśli polecenie `xinit` jest używane do uruchomienia menedżera okien lub środowiska graficznego dla innego użytkownika [Sposób 19.], należy dodać przełącznik `-- -br` po poleceniu, ale przed określeniem numeru wirtualnego terminala:

```
$ xinit /usr/bin/blackbox -- -br :1
```

Plik Xservers

Menedżery okien lub środowiska graficzne mogą używać pliku *Xservers*, w którym określony jest sposób, w jaki uruchamiany ma być system X11. W tym przypadku należy zalogować się jako użytkownik *root* i znaleźć plik *Xservers* poniższym poleceniem:

```
# locate Xservers
/etc/X11/xdm/Xservers
```

W powyższym przykładzie odnaleziony został plik *Xservers* wykorzystywany przez menedżer XDM. Jeśli jest to aktualnie używany menedżer logowania, należy zmienić ten właśnie plik. Plik *Xservers* można też znaleźć w katalogu graficznego menedżera logowania KDM dla KDE lub w katalogu menedżera GNOME — GDM. Należy zmienić tylko ten plik, który jest związany z wykorzystywanym menedżerem logowania. Plik należy otworzyć w edytorze tekstu i znaleźć w nim poniższy wiersz:

```
:0 local@tty1 /usr/X11R6/bin/X -nolisten tcp
```

Do tego wiersza trzeba dodać przełącznik `-br`:

```
:0 local@tty1 /usr/X11R6/bin/X -br -nolisten tcp
```

Plik należy zapisać, a tło z szarego wzoru powinno zmienić się na jednolity czarny.



SPOSÓB
23.

Ożywienie pulpitu animowanymi kursorami myszy

Zamiana domyślnych kursorów na kolorowe i animowane

W systemie Windows łatwo można zmienić wygląd kursora. W Linuksie nie jest to takie proste, chociaż środowisko KDE jest już temu bliskie. W tym sposobie opisano, jak sprawdzić, czy dystrybucja Linuksa pozwala na zmianę wyglądu kursorów, jak znaleźć, pobrać i zainstalować nowe wzory kursorów i jak włączyć ich wyświetlanie. Dostępnych jest wiele różnych schematów kursorów — od subtelnych dodających czerwoną kropkę do wskaźnika myszy do ekstrawaganckich przedstawiających pingwina Tuxa.

Sprawdzenie czy środowisko obsługuje schematy kursorów

Aby móc zmieniać schematy kursorów, potrzebny jest system XFree86 w wersji co najmniej 4.3 lub Xorg. Wszystkie ostatnio opublikowane dystrybucje zawierają jeden z nich, jest więc mało prawdopodobne, aby dystrybucja używana przez Czytelnika nie spełniała tych wymagań. Warto się jednak upewnić. Aby sprawdzić, jaka wersja systemu X jest zainstalowana, należy wydać polecenie:

```
# X -version
```

Jeśli zainstalowany jest system XFree86, należy upewnić się, czy jest to wersja 4.3 lub nowsza. W wyniku polecenia `X -version` trzeba poszukać ciągu tekstowego podobnego do poniższego:

```
XFree Version 4.3.0 (or higher)
```

Jeśli dystrybucja używa gałęzi Xorg, w wyniku powinien się znaleźć poniższy zapis:

```
X Protocol Version 11, Revision 0, Release 6.7 (or higher)
```

Jeśli w wyniku polecenia znajduje się jeden z powyższych ciągów, oznacza to, że dystrybucja obsługuje schematy kursorów.

Domyślnie dostępne schematy kursorów

Niestety dystrybucje Linuksa znacznie różnią się miejscem przechowywania schematów kursorów. Jeśli w dokumentacji dystrybucji nie jest napisane, w którym miejscu się one znajdują, można spróbować je odszukać. Lokalizacja schematów kursorów jest zdefiniowana w pliku `/usr/X11R6/lib/X11/config/Imake.tmpl`. Najpierw należy sprawdzić, czy plik istnieje. Można to zrobić za pomocą polecenia `ls`:

```
# ls /usr/X11R6/lib/X11/config/Imake.tmpl
```

Jeśli wynikiem polecenia jest: Nie ma takiego pliku ani katalogu (No such file or directory), można spróbować znaleźć plik w następujący sposób:

```
# locate Imake.tmpl
```

Jeśli pliku `Imake.tmpl` nie ma w systemie, należy w inny sposób odszukać, w którym miejscu dystrybucja przechowuje ogólnie dostępne pliki ze schematami kursorów. Dobrym miejscem do rozpoczęcia poszukiwań jest wyszukiwarka Google i fora dyskusyjne odpowiednich dystrybucji.

Jeśli plik jest w systemie, poniższym poleceniem można ustalić domyślne położenie schematów kursorów:

```
$ grep XcursorPath /usr/X11R6/lib/X11/config/Imake.tmpl
# ifndef XcursorPath
#     define XcursorPath "~/.cursors:~/.icons:/usr/local/share/cursors/xorg-
x11:/usr/share/cursors/xorg-x11:/usr/share/pixmaps/xorg-x11"
# ifndef XcursorPath
# define XcursorPath Concat(~/.cursors:~/.icons:/usr/local/share/cursors/xorg-
x11:/usr/share/cursors/xorg-x11:/usr/share/pixmaps/xorg-x11:, IconDir)
        XCursORPATH = XcursorPath                /* Xcursor cursors
path */
```

Pierwszy wiersz `define XcursorPath` zawiera przydatne informacje. W dystrybucji Gentoo, użytej w tym przykładzie, zostaną przeszukane następujące katalogi:

```
~/.cursors
~/.icons
/usr/local/share/cursors/xorg-x11
/usr/share/cursors/xorg-x11
/usr/share/pixmaps/xorg-x11
```

Należy sprawdzić, czy w wymienionych katalogach istnieją pliki ze schematami kursorów. Pierwsze dwa katalogi znajdują się w katalogu domowym użytkownika, jednak niewiele (jeśli jakiegokolwiek) dystrybucji instaluje tam domyślnie alternatywne schematy kursorów.

Bardziej prawdopodobne jest odnalezienie schematów w jednym z trzech pozostałych katalogów z listy. W dystrybucji Gentoo schematy znajdują się w katalogu `/usr/share/cursors/xorg-x11`, więc właśnie ta ścieżka będzie sprawdzana jako pierwsza:

```
# ls /usr/share/cursors/xorg-x11
Blue   Silver   default  gentoo-blue  handhelds  whiteglass
Gold   blueglass gentoo    gentoo-silver redglass
```



Ważna jest kolejność, w jakiej system przeszukuje katalogi. Najpierw szuka w katalogu domowym, potem przeszukuje ogólnie dostępne katalogi.

Aby zainstalować nowy schemat dla jednego użytkownika, należy umieścić go w katalogu domowym tego użytkownika (w `~/.cursors` lub w `~/.icons`).

Aby zainstalować schemat dla wszystkich użytkowników, należy umieścić go w jednym z pozostałych katalogów z listy.

To właśnie schematy kursorów. Można wybrać schemat, zwracając się do katalogu, w którym znajduje się schemat.



W niektórych dystrybucjach zdarza się nieprzemyślane rozwiązanie umieszczania schematów kursorów w tym samym miejscu, co schematów ikon. W takim przypadku nie wszystkie pliki w katalogu będą schematami kursorów. Jeśli po wybraniu schematu nadal widoczny będzie domyślny kursor, oznacza to, że został wybrany schemat ikon.

Ustawianie własnego domyślnego schematu kursorów

Każdy użytkownik może ustawić swój domyślny schemat kursorów, tworząc lub zmieniając plik `~/.icons/default/index.theme` (znak tylda `~` oznacza katalog domowy użytkownika). Plik `index.theme` informuje system, który ze schematów ma być użyty. Potrzebne katalogi należy stworzyć poniższym poleceniem:

```
$ mkdir ~/.icons ~/.icons/default
```

Następnie w katalogu `~/.icons/default` trzeba utworzyć plik `index.theme` i, za pomocą edytora tekstu, wpisać do niego następujące wiersze:

```
[Icon Theme]
Inherits=whiteglass
```

W tym przykładzie został użyty schemat kursorów o nazwie *whiteglass*. Schemat ten jest zainstalowany w systemie, katalog *whiteglass* znajduje się w ogólnodostępnym katalogu `/usr/share/cursors/xorg-x11`. Jeśli katalogu *whiteglass* nie ma w systemie, domyślny kursor się nie zmieni. Należy wtedy użyć takiego schematu, jaki dostępny jest w systemie.

W wyniku zmian, po uruchomieniu menedżera okien, powinny być widoczne nowe kursory.



Najlepszym sposobem na zmianę schematu kursorów jest modyfikacja ustawień w pliku `index.theme`, po której należy ponownie uruchomić menedżer okien lub środowisko graficzne. Jeśli się tego nie wykona, mogą wystąpić nieoczekiwane efekty lub nie będzie żadnych zmian.

Rady dla użytkowników kart graficznych NVIDIA

Posiadacze kart graficznych NVIDIA często używają sterowników NVIDIA pobranych ze strony internetowej producenta, wykorzystujących akcelerację karty graficznej. Jeśli jest używany sterownik z akceleracją, a nie domyślny sterownik z X11, kursory mogą migotać.

Aby naprawić ten błąd, należy najpierw w pliku konfiguracyjnym X11 odnaleźć fragment `Device` opisujący kartę graficzną. Plik ten nazywa się `XF86Config`, `XF86Config-4` lub `xorg.conf`, w zależności od tego, jaka wersja X serwera jest zainstalowana. Zwykle znajduje się on w katalogu `/etc/X11`. Fragment `Device` powinien zawierać poniższe wiersze, jednak nie muszą one wyglądać dokładnie tak, jak to pokazano:

```
Section "Device"
    Driver      "NVIDIA"
    Option      "HWCursor"      "On"
```

Należy wyłączyć opcję `HWCursor` — wiersz odpowiedzialny za tę opcję powinien wyglądać następująco:

```
Option      "HWCursor"      "Off"
```

Po następnym uruchomieniu środowiska graficznego, problem mrugającego kursora powinien zniknąć.

Ustawianie wybranego schematu kursorów. Przykład 1

Popularnym źródłem schematów kursorów i innych gadżetów jest strona KDE-Look (<http://www.kde-look.org>). Dokładny adres do strony zawierającej tylko schematy kursorów to <http://www.kde-look.org/index.php?xcontentmode=36>. Można stamtąd pobrać wybrany schemat. Mimo że strona ta skupia się na KDE, schematy kursorów będą działały na każdym X serwerze.

Niestety nie ma standardów dla pakietów ze schematami. Nie wszystkie pakiety zawierają plik `README`, w którym znajdują się instrukcje dotyczące rozpakowania i instalowania plików. Z tego powodu niemożliwe jest przedstawienie jednej metody postępowania z tymi pakietami.

Zazwyczaj jednak reguły są podobne — trzeba je tylko zaadaptować do pobranego schematu. W tym przykładzie będzie użyty schemat Red Dot ze strony KDE-Look. Bezpośredni adres do tego schematu to <http://www.kde-look.org/content/show.php?content=4805>, a nazwa pliku to `4805-RedDot.tar.gz`. Należy znaleźć lub utworzyć katalog tymczasowy, w którym można będzie rozpakować plik, a następnie przejść do niego, np.:

```
$ mkdir ~/temp
$ cd ~/temp
```

Po zainstalowaniu schematu można będzie skasować zawartość tego katalogu. Teraz trzeba rozpakować pobrany plik poniższym poleceniem:

```
$ tar zxvf 4805-RedDot.tar.gz
```

Zostaną utworzone dwa katalogi: *RedDot* i *RedDotSource*. Opcja `-z` polecenia `tar` informuje, że plik jest spakowany programem *gzip*.

Po rozpakowaniu pliku pojawią się katalogi ze schematem kursorów. Są to te, które zawierają podkatalog *cursors*. W tym przypadku podkatalog ten znajduje się w katalogu *RedDot* i właśnie ten katalog trzeba zainstalować. (Dla ciekawskich — w katalogu *RedDotSource* znajdują się pliki, których autor schematu użył do jego tworzenia).

Najprostszym sposobem instalacji kursorów Red Dot jest po prostu skopiowanie katalogu *RedDot* wraz z jego zawartością do katalogu `~/icons`:

```
$ cp -a RedDot ~/.icons
```

Jeśli schemat ma być dostępny dla wszystkich użytkowników, należy skopiować katalog *RedDot* do jednego z ogólnodostępnych katalogów (zgodnie z jednym z powyższych przykładów katalogi te to `/usr/local/share/cursors/xorg-x11`, `/usr/share/cursors/xorg-x11`, `/usr/share/pixmaps/xorg-x11`, jednak w innych dystrybucjach mogą być inne). Aby możliwe było skopiowanie katalogu, należy zalogować się jako użytkownik *root*, w przeciwnym razie zbyt niskie uprawnienia nie pozwolą na zapisywanie zmian w tych katalogach.

W pliku `~/icons/default/index.theme` należy umieścić poniższe wiersze:

```
[Icon Theme]
Inherits=RedDot
```

Po ponownym uruchomieniu menedżera okien lub środowiska graficznego powinny być widoczne kursory ze schematu Red Dot.

Ustawianie wybranego schematu kursorów. Przykład 2

Jak wcześniej zostało wspomniane, nie ma jednego standardu dla pakietów ze schematami kursorów. Poniższy przykład pokazuje, że reguły instalacji schematu są takie same, nawet jeśli pakiet jest inny.

W tym przypadku należy pobrać schemat Golden XCursors 3D dostępny na stronie <http://www.kde-look.org/content/show.php?content=5507>. Plik ze schematem nazywa się *5507-Golden-XCursors-3D-0.8.tar.bz2*. Należy go rozpakować poniższym poleceniem:

```
$ cd ~/temp
$ tar jxvf 5507-Golden-XCursors-3D-0.8.tar.bz2
```

Zostaje utworzony katalog *Golden-XCursors-3D-0.8*. (Jeśli wersja pliku zmieniła się od czasu wydania tej książki, należy zmienić powyższe polecenie, podając nową nazwę pliku ze schematem). Następnie trzeba przejść do tego katalogu i wyświetlić jego zawartość poleceniem `ls`:

```
$ cd Golden-XCursors-3D-0.8
$ ls
COPYING Gold README default
```

W tym przypadku autor dołączył plik *README* z instrukcjami opisującymi sposób instalacji schematu, ale gdyby go nie było, można użyć metody z poprzedniego przykładu. Najpierw należy odnaleźć katalog zawierający podkatalog *cursors*. Trzeba otworzyć katalog *Gold*, wydając następujące polecenie:

```
$ ls Gold  
cursors
```

Znajduje się tam szukany podkatalog *cursors*. Należy więc skopiować katalog *Gold* do katalogu *~/icons*:

```
$ cp -a Gold ~/.icons
```

Aby schemat Gold był dostępny dla wszystkich użytkowników, należy skopiować katalog *Gold* do ogólnodostępnego katalogu (katalogi te to: */usr/local/share/cursors/xorg-x11*, */usr/share/cursors/xorg-x11*, */usr/share/pixmaps/xorg-x11*, ale w innych dystrybucjach mogą być inne). Należy zalogować się jako użytkownik *root*, zbyt niskie uprawnienia nie pozwolą bowiem na zapisywanie zmian w tych katalogach.

W pliku *~/icons/default/index.theme* trzeba umieścić poniższe wiersze:

```
[Icon Theme]  
Inherits=RedDot
```

Po ponownym uruchomieniu menedżera okien lub środowiska graficznego powinien być widoczny kursor w schemacie Gold.

SPOSÓB
24.

Konwersja kursorów XP do Linuksa

Użycie kursorów zaprojektowanych dla systemu Windows XP w Linuksie

Można użyć jeszcze bardziej ozdobnych kursorów niż te, które są dostępne dla Linuksa. Firma Stardock dla systemu Windows XP stworzyła produkt CursorXP. Pozwala on użytkownikom systemu Windows na wybór z setek ozdobnych animowanych schematów kursorów. W tym sposobie przedstawiony jest skrypt, który konwertuje te schematy tak, aby mogły być użyte z systemem X11 w Linuksie.

Istnieje ogromny zbiór dostępnych schematów kursorów przeznaczonych do użytku z programem CursorXP. Można je znaleźć na stronie <http://www.wincustomize.com>. Bezpośredni adres prowadzący do strony zawierającej schematy kursorów to: <http://www2.wincustomize.com/skins.asp?library=25>.

Aby można było wykorzystać te schematy w systemie Linux, należy skorzystać ze skryptu napisanego przez Erica Windischa i zmodyfikowanego przez Nicholasa Petreleya i Jamesa Barrona. Skrypt ten, napisany w języku Perl, służy do konwersji schematów CursorXP na format obsługiwany przez środowisko X11. W tym sposobie użyta będzie wersja skryptu o nazwie *np-sd2xc.pl*.

Skrypt do poprawnego działania wymaga pakietu ImageMagic w wersji 6 lub nowszej. Będą także potrzebne następujące moduły języka Perl:

- `Image::Magick`,
- `Getopt::Long`,
- `Config::IniFiles`.

Po upewnieniu się, że wszystkie niezbędne pakiety są zainstalowane, należy pobrać schemat kursorów przeznaczony do konwersji. W tym przykładzie będzie użyty plik *Gear.zip*. Aby rozpocząć konwersję, należy rozpakować go do roboczego katalogu. Poniższe polecenia korzystają z katalogu roboczego o nazwie *geartemp* (jeśli został pobrany inny plik niż *Gear.zip*, należy zmienić nazwę pliku w poleceniach):

```
$ mkdir ~/geartemp
$ cd ~/geartemp
$ unzip Gear.zip
```

Polecenie rozpakowuje plik, dzięki temu powstaje nowy o nazwie *Gear.CurXPTheme*. Mimo że plik ma rozszerzenie *.CurXPTheme*, to jest to następny plik spakowany programem *zip*, który można rozpakować poniższym poleceniem:

```
$ unzip Gear.CurXPTheme
```

Po rozpakowaniu należy skorzystać ze skryptu *np-sd2xc.pl*. Przełącznik *-name* pozwala na określenie nowej nazwy schematu kursorów. Schemat użyty w tym przykładzie nazywa się *Gear*, ale można mu nadać inną nazwę. Dla przykładu będzie to *CoolGear*. Należy wydać polecenie:

```
$ np-sd2xc.pl -name CoolGear
```

Podczas konwersji schematów CursorXP program może dodać cienie do wszystkich kursorów. To kwestia gustu, ale ja uważam, że kursory 3D z cieniami ładnie wyglądają. Aby dodać cień do kursorów, należy dopisać przełącznik *-shadow* do polecenia:

```
$ np-sd2xc.pl -shadow -name CoolGear
```

Teraz można zainstalować schemat (w tym przykładzie będzie zainstalowany w katalogu domowym użytkownika). Poniższe polecenia instalują schemat i ustawiają go jako domyślny (przełącznik *-R* wskazuje poleceniu *cp*, żeby skopiować wszystkie podkatalogi wraz z ich zawartością do katalogu *~/icons*):

```
$ cp -R CoolGear ~/.icons
$ cp -R default ~/.icons
```

Drugie polecenie nadpisuje plik *index.theme* w katalogu *~/icons/default*.

Po ponownym uruchomieniu środowiska graficznego lub menedżera okien powinny być widoczne kursory oryginalnie zaprojektowane dla systemu Windows w linuksowym środowisku opartym na X serwerze.



Nie wszystkie schematy z CursorXP udaje się dobrze skonwertować. Źle zachowujący się schemat może uniemożliwić włączenie środowiska graficznego lub menedżera okien. Jeśli wystąpi ten problem, należy przełączyć się kombinacją klawiszy *Ctrl+Alt+F1* do wirtualnej konsoli i zalogować się. Następnie należy otworzyć w edytorze tekstu plik `~/icons/default/index.theme` i zmienić w nim parametr `Inherits` na schemat, który działa.

**SPOSÓB
25.**

Użycie czcionek z systemu Windows i Macintosha

Wykorzystanie ulubionych czcionek z innych systemów operacyjnych

Dużym problemem w linuksowym środowisku jest brak wysokiej jakości czcionek. Tworzenie czcionek to czasochłonny i drogi proces, który wymaga dużo umiejętności. Ci, którzy to potrafią, w większości nie chcą oddawać swojej pracy za darmo (ta postawa wyraźnie kontrastuje z tysiącami programistów *open source*, którzy publikują swój kod za darmo). Można kupić czcionki przeznaczone dla Linuksa, ale taniej jest użyć czcionek z innego systemu operacyjnego.



Kopiując czcionki z innego systemu operacyjnego, trzeba mieć na uwadze prawne konsekwencje swoich czynów. Czcionki, będące częścią prawnie zastrzeżonych systemów operacyjnych, mogą być chronione licencją, która ogranicza ich użycie. Przed ich kopiowaniem należy przeczytać warunki licencji dotyczące czcionek.

Użycie czcionek z systemu Windows

Linux w pełni obsługuje czcionki TrueType używane w systemie Windows. Można je z niego skopiować i zainstalować w katalogach czcionek systemu X albo w katalogu `.fonts` w katalogu domowym użytkownika.

Chociaż powyższa metoda działa, nie zawsze jest konieczne kopiowanie plików — kiedy system Windows znajduje się na tym samym komputerze, co Linux, można użyć bardziej eleganckiej metody.

Po pierwsze, trzeba zamontować partycję z systemem Windows. Aby to zrobić, potrzebne jest jądro Linuksa obsługujące system plików VFAT (lub NTFS w przypadku systemów Windows 2000 i XP). Przy samodzielnej kompilacji jądra należy dodać obsługę tych systemów. W narzędziu do konfiguracji jądra należy zaznaczyć opcję: *File systems / DOS/FAT/NT / File systems / VFAT Support* lub — przy systemie plików NTFS: *File systems / DOS/FAT/NT / File systems / NTFS file system*.

Aby uzyskać dostęp do partycji z systemem Windows, należy ją zamontować. Można to zrobić ręcznie poleceniem:

```
foo@bar:~$ mkdir /mnt/windows
foo@bar:~$ mount -t vfat /dev/hda1 /mnt/windows
```

Partycja z systemem Windows może się znajdować w innym miejscu niż `/dev/hda1` — w takim przypadku należy zmodyfikować powyższe polecenie. Punkt montowania partycji jest dowolny — w tym przykładzie jest to `/mnt/windows`. Gdy nie ma pewności, jakie partycje są dostępne, można to sprawdzić poleceniem:

```
foo@bar:~$ ls -al /dev/hda*
```

Żeby ułatwić montowanie, można dodać ten punkt do listy dostępnych punktów montowania w pliku `/etc/fstab`. Plik ten wskazuje systemowi dostępne dyski oraz sposób, w jaki można się do nich dostać. Trzeba do niego dodać jeden z dwóch poniższych wierszy, w zależności od tego, jaki system plików jest używany na partycji z systemem Windows:

```
/dev/hda1    /mnt/windows    vfat    rw    0    0
/dev/hda1    /mnt/windows    ntfs    ro    0    0
```

Powyższy zapis przekazuje systemowi informację, że partycja `/dev/hda1` jest dostępna w punkcie montowania `/mnt/windows` z możliwością odczytu i zapisu oraz żeby zamontował ją przy uruchomieniu systemu. Ponieważ w Linuksie zapis na partycjach NTFS nie jest w pełni obsługiwany, a jego użycie może spowodować zniszczenie danych, system NTFS będzie zamontowany z prawem *tylko do odczytu* (*read only*).

Aby użyć czcionek, potrzebny jest dostęp do katalogu z czcionkami w systemie Windows i uruchomienie kilku narzędzi, które sprawiają, że czcionki będą użyteczne w systemie X. Chociaż X obsługuje czcionki TrueType, potrzebuje kilku specjalnych plików, które zawierają informacje o czcionkach. W systemie Windows czcionki znajdują się zazwyczaj w katalogu `C:\Windows\Fonts`, należy więc przejść do `/mnt/windows/Windows/Fonts` i wydać opisane niżej polecenia.

Pierwsze polecenie — `ttmkfdir` — tworzy specjalny plik informacyjny o czcionkach o nazwie `fonts.scale`. Plik ten zawiera listę czcionek i ich właściwości w formacie zrozumiałym dla systemu X. Aby utworzyć plik, należy przejść do katalogu z czcionkami i wydać polecenie:

```
foo@bar:~$ ttmkfdir
```

W przypadku, gdy katalog z czcionkami systemu Windows znajduje się na partycji NTFS, nie można wydać tego polecenia, ponieważ partycja ta jest zamontowana w trybie *tylko do odczytu*. Można sobie poradzić z tym problemem poprzez skopiowanie czcionek do katalogu znajdującego się na partycji z Linuksem (np. `/tmp/fonts`), a następnie z poziomu tego katalogu trzeba wydać powyższe polecenie. Następnie należy skopiować utworzony plik `fonts.scale` na nośnik, który będzie dostępny też z systemu Windows, jak np. dyskietka lub pendrive. Stamtąd należy go przenieść do katalogu `C:\Windows\Fonts`. Tę operację trzeba będzie powtórzyć za każdym razem, gdy zostaną dodane nowe czcionki w systemie Windows, które mają być też dostępne z Linuksa.

Po stworzeniu pomocniczego pliku należy powiadomić system XFree86, że istnieje nowy katalog z czcionkami, który ma być przez niego używany. Można to zrobić na dwa sposoby.

Pierwszy to dodanie wiersza `FontPath` do pliku konfiguracyjnego systemu X11, zwykle `/etc/X11/XF86Config-4`:

```
FontPath "/mnt/windows/Windows/Fonts"
```

Po dodaniu tego wiersza należy ponownie uruchomić X serwer.



Trzeba pamiętać, żeby zwracać uwagę na wielkość liter w nazwach katalogów. Na przykład główny katalog systemu Windows pisany jest czasem wielkimi literami: `WINDOWS`. Należy więc odpowiednio dostosować ścieżkę dostępu.

Inną metodą jest użycie polecenia `chkfontpath` do dynamicznego dodania ścieżki (sposób ten działa tylko wtedy, gdy używany jest serwer czcionek XFS):

```
foo@bar:~$ chkfontpath --add /mnt/windows/Windows/Fonts
```

Jeśli katalog został dodany tym poleceniem, trzeba uruchomić ponownie serwer czcionek i system XFree86. Można to zrobić poniższym poleceniem:

```
foo@bar:~$ /etc/init.d/xfs restart
```

Od tej pory czcionki z systemu Windows powinny być dostępne.

Użycie czcionek z systemu Mac OS X

Chociaż system Mac OS X obsługuje czcionki TrueType, firma Apple umieszcza czcionki i wiele dodatkowych informacji systemowych w plikach zwanych *data fork resource files*. Pliki te mają rozszerzenie `.dfont` i zawierają dużo więcej informacji niż przeciętny plik czcionki TrueType. Informacje te przeznaczone są dla systemu Mac OS X, więc zwykłe skopiowanie tych plików do Linuksa nie wystarczy — system X nie będzie wiedział, co zrobić z dodatkowymi informacjami. Trzeba więc przekonwertować te pliki za pomocą programu Fondu.

Program Fondu (<http://fondu.sourceforge.net/>) wydostaje dane o czcionkach z plików `.dfont` i tworzy użyteczne pliki TrueType. Fondu zawiera nie tylko konwerter, ale także wiele innych użytecznych narzędzi, które radzą sobie z różnicami między definicjami czcionek z systemów uniksowych i z systemu Mac OS X. Z podanej wyżej strony należy pobrać archiwum Mac OS X Stufflt zawierające program Fondu.

Po pobraniu archiwum należy kliknąć dwa razy ikonę programu, aby rozpakować zawartość. Na pulpicie pojawi się ikona `.pkg`. Po kliknięciu tej ikony rozpocznie się proces instalacji. Chociaż instalator wymaga jedynie klikania klawisza *Next* i podania miejsca instalacji plików, trzeba pamiętać o ustawieniu zmiennych `PATH` i `MANPATH` na `/usr/local/bin` i `/usr/local/man`, aby pliki były dostępne w całym systemie Mac OS X. Można to zrobić w pliku `.profile` w katalogu użytkownika (plik ten jest odczytywany przy każdym logowaniu — ustawia on wszystkie zmienne środowiskowe).

```
PATH = /usr/local/bin:$PATH
MANPATH = /usr/local/man:$MANPATH
export PATH MANPATH
```

Aby sprawdzić, czy zmienne są prawidłowo ustawione, należy wylogować się i zalogować ponownie poniższymi poleceniami:

```
foo@bar:~$ echo $PATH
foo@bar:~$ echo $MANPATH
```

Aby skonwertować czcionki, należy utworzyć tymczasowy katalog, w którym będzie odbywał się proces konwersji:

```
foo@bar:~$ mkdir ~/fontconv
```

Stworzenie tego katalogu jest konieczne, ponieważ katalogi systemowe są dostępne w trybie *tylko do odczytu*. Poniższe polecenie kopiuje pliki z */System/Library/Fonts* do nowego katalogu:

```
foo@bar:~$ cp /System/Library/Fonts/* ~/fontconv
```

Należy przejść do katalogu *fontconv* i przeprowadzić konwersję:

```
foo@bar:~$ fondu *
```

Po ukończeniu procesu w katalogu pojawiają się pliki z rozszerzeniem *.ttf*. Należy utworzyć katalog o nazwie *macfonts*. Następnie trzeba skopiować do niego pliki TrueType i utworzyć z nich archiwum *tar*:

```
foo@bar:~$ mkdir ~/macfonts
foo@bar:~$ cp *.ttf ../macfonts
foo@bar:~$ tar zcvf ../macfonts/*.ttf macfonts
```

Stworzone w ten sposób archiwum należy skopiować do Linuksa i rozpakować je do systemowego katalogu z czcionkami lub do katalogu *.fonts* w katalogu domowym użytkownika. Jak to zostało zaznaczone w poprzednim fragmencie, należy wydać polecenie *ttmkfdir* tworzące plik informacyjny o czcionkach. Należy też dodać ścieżkę do czcionek w pliku konfiguracyjnym systemu X11.



SPOSÓB

26.

Jak nie przeoczyć przypomnienia

Przypomnienia mogą pojawiać się nad wszystkimi innymi aplikacjami na pulpicie

Możliwe jest wyświetlanie wiadomości na wierzchu X Window, nawet w trakcie używania wielu pulpitów wirtualnych. Może być to przydatne, gdy zwykle wyskakujące okienko lub wiadomość pojawiająca się w terminalu niedostatecznie przyciąga uwagę użytkownika. W tym sposobie wykorzystane zostaną możliwości funkcji programu X11 On-Screen Display (XOSD), który zostanie zastosowany przy wyświetlaniu przypomnienia na ekranie.

XOSD pozwala wyświetlać wiadomości, symbole lub paski postępu bezpośrednio na ekranie nad wszystkimi innymi aplikacjami. Przypomina to wskaźnik głośności w wielu telewizorach.

Instalacja programu XOSD jest zależna od dystrybucji, która jest używana przez Czytelnika. Nazwa pakietu, który trzeba zainstalować, zawiera zwykle wyrażenie XOSD. Użytkownicy korzystający z zarządcy pakietów yum (np. w dystrybucji Fedora Core i innych opartych na pakietach RPM), powinni zalogować się jako użytkownik *root* i zainstalować program XOSD poniższym poleceniem:

```
# yum install xosd
```

W przypadku Debiana i innych dystrybucji bazujących na pakietach RPM, które obsługiwane są przez program *apt-get*, XOSD instaluje się poniższym poleceniem:

```
# apt-get install xosd-bin
```

Aby zainstalować XOSD w dystrybucji Gentoo, należy wydać polecenie:

```
# emerge xosd
```

Dużą zaletą używania XOSD jest fakt, że przyciąga on uwagę skuteczniej niż jakakolwiek inna technika komunikatów. Wiadomość pojawia się na każdym pulpicie wirtualnym i przesłania każdą używaną aplikację. Można wypróbować sposób działania XOSD poniższym poleceniem:

```
$ echo "Cześć, tu program osd_cat." | osd_cat
```

Wiadomość powinna się pokazać w górnym lewym narożniku ekranu. Jest ona wyświetlona małą czerwoną czcionką. Jest tak mała, że prawie nieprzydatna. Istnieją proste sposoby powiększenia jej rozmiaru. Jeśli tekst się pokazał, oznacza to, że program *osd_cat* działa.

Upiększanie napisu

Aby ustalić wygląd napisu wyświetlanego przez program *osd_cat*, można dodać wiele przełączników. Po pierwsze, wypadaloby zwiększyć rozmiar czcionki. Aby użyć wybranej czcionki, należy dodać przełącznik *-f* z definicją czcionki.

Definicje czcionek takich jak te, które używane są przez program *osd_cat*, wyglądają nieprzystępnie — zawierają informacje opisane mnóstwem gwiazdek i kresek. Każda część definicji przekazuje informacje o typie czcionki, wielkości, pogrubieniu itd. Bezpiecznie jest przepisać definicje z tej książki, ale jeśli Czytelnik ma ochotę na eksperymenty, może skorzystać z programu *xfontsel*, w którym można dowolnie zmieniać ustawienia czcionek. Program pozwala ustawiać różne parametry, które tworzą definicję czcionki wyświetlanej na ekranie. Po utworzeniu całej definicji należy kliknąć przycisk *Select*, wówczas definicja zostanie skopiowana do schowka. Można ją wtedy wkleić do skryptu lub programu.

Poniższa definicja opisuje dużą, przyciągającą wzrok, czcionkę dostępną w prawie każdej dystrybucji Linuksa:

```
-adobe-helvetica-bold-r-normal-**-240-***-p-***-
```

Jeśli zainstalowane są czcionki TrueType z systemu Windows [Sposób 25.], wybór rozmiarów czcionek jest znacznie większy. Im większa czcionka, tym dłużej będzie renderowana przez program *osd_cat*. Oto przykład:

```
"-microsoft-comic sans ms--*-**-48--*-**-*-**"
```

Można wyświetlić tekst napisany czcionką większą niż 48 punktów, ale może minąć kilka sekund, zanim pojawi się on na ekranie.



Definicje czcionek zawierające spacje muszą być umieszczone w cudzysłowie. W przeciwnym wypadku program *osd_cat* może nie zrozumieć definicji.

Poniżej znajduje się polecenie *osd_cat* z definicją czcionki:

```
$ echo "Cześć, tu program osd_cat." | osd_cat -f -adobe-helvetica-bold-r-normal--*-240--*-p--*-**
```

Po wydaniu tego polecenia na ekranie powinna pojawić się wiadomość napisana dużo większą czcionką.

Aby zmienić kolor tekstu, należy użyć przełącznika *-c*. Kolory, których można użyć, są zdefiniowane w pliku */usr/X11R6/lib/X11/rgb.txt*. Najbezpieczniej jest wybierać kolory standardowe: *blue*, *green*, *yellow*, *cyan*, *white*, *black* i *magenta*. Wybredni użytkownicy mogą wybrać też inne kolory, np. *PapayaWhip*.

Kolor zielony jest typowy dla informacji wyświetlanych na ekranie telewizora, więc sprawdzi się też w XOSD:

```
$ echo "Cześć, tu program osd_cat." | osd_cat -c green \
-f -adobe-helvetica-bold-r-normal--*-240--*-p--*-**
```

Wyświetlany tekst może rzucać cień. Oczywiście, jeśli tekst pojawia się na ciemnym tle, cienia nie będzie widać. Aby dodać cień o grubości dwóch pikseli, należy dodać przełącznik *-s 2*:

```
$ echo "Cześć, tu program osd_cat." | osd_cat -c green -s 2 \
-f -adobe-helvetica-bold-r-normal--*-240--*-p--*-**
```

Pozycję wyświetlanego tekstu można ustalić przełącznikiem *-p*. Przełącznik *-p middle* ustawia tekst na środku ekranu (można wybrać *top* i *bottom* — odpowiednio na górze i na dole ekranu). Można zauważyć, że przy ustawieniu *middle* tekst na moment pojawia się na górze ekranu, dopiero potem na środku. Ta wada programu *osd_cat* będzie prawdopodobnie naprawiona w jego następnych wersjach. Można ustawić też czas wyświetlania wiadomości na okres dłuższy niż domyślne 5 sekund. Aby napis pozostał na ekranie przez 60 sekund, należy dodać przełącznik *-d 60*:

```
$ echo "Cześć, tu program osd_cat." | osd_cat -c green -s 2 -p middle -d 60 \
-f -adobe-helvetica-bold-r-normal--*-240--*-p--*-**
```

Dostępne są też inne przełączniki służące na przykład do robienia wcięć w tekście. Więcej informacji o przełącznikach można znaleźć na stronie `man` polecenia `osd_cat`.

Skrypt wyświetlający przypomnienia

Kiedy wiadomo już, co potrafi XOSD, można napisać skrypt wyświetlający przypomnienia o określonej godzinie. Aby skrypt działał, oprócz programu `osd_cat`, potrzebny będzie program `at`. Musi być również uruchomiony demon `atd`. Wiele dystrybucji Linuksa instaluje `at` i `atd` oraz uruchamia demona domyślnie. Jeśli programy te nie są zainstalowane, należy je zainstalować, postępując zgodnie z procedurą dla konkretnej dystrybucji i upewnić się, że program `atd` uruchamia się podczas startu komputera.

Aby stworzyć skrypt, należy zalogować się jako użytkownik `root` lub użyć polecenia `sudo`, uruchomić ulubiony edytor tekstu i utworzyć plik `/usr/local/bin/remindme`. Do niego należy wpisać skrypt z przykładu 4.1.

Przykład 4.1. Skrypt bash do pliku remindme

```
#!/bin/bash

# sprawdzenie, który ekran jest obecnie używany
HOST="$(xrdp -symbols | grep SERVERHOST | cut -d= -f2)"
DISPLAYNUM="$(xrdp -symbols | grep DISPLAY_NUM | cut -d= -f2)"
THISDISPLAY=$HOST:$DISPLAYNUM.0

# sprawdzenie, czy katalog (reminders) skryptu przypominającego jest w katalogu domowym
# jeśli nie ma, tworzenie katalogu

if [ ! -e ~/reminders ] ; then
    mkdir ~/reminders
fi

unique=`date +%F-%H-%M-%S`
reminder="reminders/reminder-"$unique
# tworzenie wynikowego skryptu reminder
echo '#!/bin/bash' > ~/$reminder
echo $THISDISPLAY >> ~/$reminder
echo "echo \"$2 dzisiaj\" | osd_cat -s 2 -c green -p middle \
-f -adobe-helvetica-bold-r-normal-***-240-***-p-***- -d 60" >> ~/$reminder
# nadanie mu atrybutu wykonywania
chmod +x ~/$reminder
# powiadomienie demona atd o konieczności uruchomienia skryptu o zadanej porze
at $1 -f ~/$reminder
```

Należy zapisać plik i nadać mu atrybut wykonywania:

```
$ sudo chmod +x /usr/local/bin/remindme
```

Aby ustawić przypomnienie o telefonie do szefa o godzinie 14:00, należy wydać polecenie:

```
$ remindme 14:00 "Zadzwoń teraz do szefa."
```

Powyższy skrypt przekształca to polecenie na osobny skrypt *reminder* i ustawia czas jego uruchomienia na 14.00. Nowy skrypt, korzystając z XOSD, umieści wiadomość na ekranie. Oznacza to, że nie będzie można jej przeoczyć, ponieważ będzie umieszczona na wszystkich pulpitych.

Działanie skryptu przebiega w kilku etapach. Pierwsza część sprawdza, który ekran X Window jest aktualnie używany. Potem do pliku *reminder* zostaje dodany wiersz ustawiający numer ekranu, na którym ma pracować roboczy skrypt. Program *osd_cat* nie pobiera tej informacji od środowiska, więc musi być ona umieszczona w skrypcie. Nie można zakładać, że użytkownik będzie zawsze używał domyślnego ekranu: 0.0. Skrypt musi więc określić aktualny ekran, wykonując polecenie *xrdb*.



Niektóre wersje programu *osd_cat* zwracają szczególną uwagę na sposób zapisu zmiennej *DISPLAY*. Nie działają, gdy przed wyrażeniem `<:0.0>` podana zostanie nazwa komputera. Należy o tym pamiętać podczas uruchamiania programu.

Następna część skryptu tworzy unikalną nazwę pliku bazując na aktualnym czasie. Potem tworzy pierwszy wiersz skryptu *bash* (`#!/bin/bash`) i zapisuje go do pliku *reminder*, który uruchomi się w podanym czasie.

Następnie dodaje polecenie wyświetlenia powiadomienia na ekranie, w którym `$2` zostanie zastąpione wartością drugiego parametru wywołania skryptu, czyli treścią powiadomienia (dla powyższego przykładu będzie to „zadzwoń do szefa”). Kolejna część skryptu nadaje stworzonemu plikowi atrybut wykonywalności. Ostatnia część skryptu informuje demona *atd*, aby w zadanym czasie uruchomił stworzony wcześniej skrypt. Godzinę wywołania przypomnienia można podać na kilka różnych sposobów, np. 14:00, 2:00PM lub 2pm. Wszystkie te formaty będą poprawnie odczytane przez program. Skrypt domyślnie przyjmuje, że skrypt przypominający ma zostać wywołany tego samego dnia, w którym jest tworzony. Ambitni użytkownicy mogą zmienić skrypt tak, aby pozwalał również określać datę wyświetlania powiadomienia.

Można zauważyć, że skrypt, na podstawie aktualnej daty i czasu, nadaje tworzonemu plikom unikalne nazwy i zapisuje je w katalogu `~/reminders`, np. `~/reminders/reminder-2004-10-04-05-07-39`. Symbol tylda `~` oznacza katalog domowy użytkownika. Wszystkie pliki *reminder* są więc przechowywane w podkatalogu *reminders* w katalogu domowym. Jeśli skrypt będzie często używany, katalog ten zapełni się plikami tymczasowymi. Można skasować niepotrzebne pliki, oceniając ich przydatność na podstawie daty zawartej w nazwie pliku. Każdy plik *reminder* zawiera w nazwie dokładną datę i godzinę, dzięki czemu łatwo można rozpoznać przedawnione pliki i skasować je.



Należy pamiętać, że data z nazwy pliku to data jego utworzenia, a nie wyświetlenia na ekranie. Jeśli plik został utworzony o 13.00, a ma być wyświetlony o 15.00, to nie należy go kasować przed 15.00.



SPOSÓB

27.

Wyświetlanie komunikatów w odpowiedzi na zdarzenia

Wiele aplikacji może uruchomić zewnętrzny program w wyniku zajścia jakiegoś zdarzenia — użycie skryptów wykorzystujących XOSD spowoduje wyświetlenie komunikatów przyciągających uwagę

Wiele aplikacji używanych na co dzień ma możliwość uruchamiania zewnętrznych programów w wyniku jakiegoś zdarzenia. Przykładami mogą być: klient poczty KMail, menedżer osobisty Jpilot czy monitor plików log Swatch. Każdy z opisanych w tym rozdziale sposobów korzysta z XOSD [Sposób 26.].

Program KMail

Jeśli używany jest klient poczty KDE KMail, można skonfigurować go tak, aby uruchamiał zewnętrzny program za każdym razem, gdy przychodzi nowa wiadomość. Przykład ten pokazuje, jak za pomocą XOSD wyświetlić na ekranie komunikat „Nadeszła nowa wiadomość pocztowa” za każdym razem, gdy KMail odbierze nową pocztę.

Aby to osiągnąć, należy napisać skrypt, który wyświetli na ekranie napis „Nadeszła nowa wiadomość pocztowa”. Program KMail nie pozwala na podanie całego polecenia wyświetlającego komunikat jako programu, który zostanie uruchomiony po nadejściu poczty. Zamiast tego należy podać nazwę skryptu, który zostanie w takiej sytuacji wykonany. Zadaniem tego skryptu jest wyświetlenie odpowiedniego napisu. Za pomocą ulubionego edytora tekstu należy utworzyć plik `~/nowapoczta` i wpisać do niego poniższy skrypt. Uruchamia się on z katalogu domowego użytkownika, więc aby go stworzyć, nie są potrzebne uprawnienia użytkownika `root`.

```
#!/bin/bash

# sprawdzenie, który ekran jest obecnie używany
# potem wyeksportowanie zmiennej środowiskowej DISPLAY

HOST="$(xrdp -symbols | grep SERVERHOST | cut -d= -f2)"
DISPLAYNUM="$(xrdp -symbols | grep DISPLAY_NUM | cut -d= -f2)"
THISDISPLAY=$HOST:$DISPLAYNUM.0

export DISPLAY=$THISDISPLAY

echo "Nadeszła nowa wiadomość pocztowa"'\'' | osd_cat -s 2 -c yellow -p middle \
-f -adobe-helvetica-bold-r-normal-***-240-***-p-*** -d 30
```



W powyższym zapisie dziwić może użycie różnych rodzajów cudzysłowów. Zastosowanie takiej składni jest konieczne ze względu na interpreter języka bash, dla którego wykrzyknik ma specjalne znaczenie — powoduje odwołanie się do historii poleceń. Aby wykrzyknik mógł być wyświetlony, musi być zapisany w pojedynczym cudzysłowie.

Należy zapisać plik i nadać mu atrybut wykonywania:

```
$ chmod +x ~/nowapoczta
```

Aby program KMail wykonywał skrypt, gdy nadejdzie nowa wiadomość, należy wykonać poniższe kroki:

1. Włączyć program KMail i wybrać z menu *Ustawienia/Konfiguruj powiadomienia....*
2. Kliknąć przycisk *Więcej opcji* umieszczony pod polem *Działania*.
3. Zaznaczyć opcję *Wykonaj program*.
4. Wpisać **~/nowapoczta** w polu edycji tej sekcji.
5. Kliknąć przyciski *Zastosuj* i *OK*, co spowoduje powrót do głównego ekranu programu KMail.

Program Jpilot

Menedżer osobisty Jpilot można skonfigurować tak, aby uruchamiał skrypt wyświetlający na ekranie wiadomości w określonym czasie. Za pomocą edytora tekstu należy stworzyć poniższy skrypt i zapisać plik jako *~/jpilotalert*. Skrypt uruchamia się z katalogu domowego użytkownika, więc uprawnienia użytkownika *root* nie są potrzebne.

```
#!/bin/bash

# sprawdzenie, który ekran jest obecnie używany
# potem wyeksportowanie zmiennej środowiskowej DISPLAY

HOST="$(xrdp -symbols | grep SERVERHOST | cut -d= -f2)"
DISPLAYNUM="$(xrdp -symbols | grep DISPLAY_NUM | cut -d= -f2)"
THISDISPLAY=$HOST:$DISPLAYNUM.0

export DISPLAY=$THISDISPLAY

echo "Umówione spotkanie $1 $2"'"!' | osd_cat -s 2 -c yellow -p middle \
-f -adobe-helvetica-bold-r-normal-**-240-**-p-***- -d 30
```

Poniżej przedstawione są kroki, które trzeba wykonać, aby program Jpilot wykonywał zadany skrypt. Pokazany jest również sposób, w jaki należy przekazywać datę i godzinę przypomnienia do zewnętrznego programu:

1. otworzyć okno *Preferences*, wybierając z menu *File/Preferences* lub wciskając kombinację klawiszy *Ctrl+E*;
2. kliknąć zakładkę *Alarms*;
3. zaznaczyć opcję *Execute this command*;
4. wpisać następujący ciąg tekstowy do pola tekstowego *Alarm Command*:
~/jpilotalert "%d" "%t".

Program Swatch

Swatch to program, który monitoruje dzienniki systemowe w poszukiwaniu słów kluczowych. Może być skonfigurowany w taki sposób, aby przeszukiwał jeden lub więcej plików z dziennikami. Gdy program znajdzie zadane słowo lub napotka wyrażenie pasujące

Ostrzeżenie programu Swatch jest standardowo wypisywane w oknie, w którym uruchomiony jest program. Może być także wysłane jako wiadomość za pomocą poczty email, jednak do czasu jej odczytania włamanie do sieci może być już faktem. Ten sposób wykorzystuje możliwości XOSD do zwrócenia uwagi użytkownika na ostrzeżenia programu Swatch.

Poniżej przedstawiono przykład ustawienia programu Swatch tak, aby przekazywał wiadomość o nieudanej próbie zalogowania się (prawdopodobnie ktoś próbuje odgadnąć hasło). W tym przykładzie plik konfiguracyjny programu Swatch to `/root/.swatchrc`, a obserwowany plik dziennika to `/var/log/auth.log` — w kilku dystrybucjach jest to plik rejestrujący wszystkie próby zalogowania. Poniżej znajduje się fragment pliku `/root/.swatchrc`, który szuka słowa kluczowego *failure* w pliku `/var/log/auth.log`:

```
# Bad login attempts
watch for /failure/
    pipe "osd_cat -c magenta -p middle \
-f \"*-arial black-**-***-48-**-***-**-**\" -d 60"
```

Kiedy program Swatch znajdzie nowy wpis w dzienniku zawierający słowo *failure*, prześle go do programu *osd_cat*, który wyświetli go na ekranie. Prawie niemożliwe jest niezauważenie tego napisu podczas pracy przy komputerze. Wielkość czcionki tego napisu będzie mniejsza niż w poprzednich sposobach, ponieważ wiadomości programu są zwykle długie, a dobrze by było, gdyby całe ostrzeżenie zmieściło się na ekranie.



Włączanie klawiszy szybkiego dostępu za pomocą programu Hotkeys

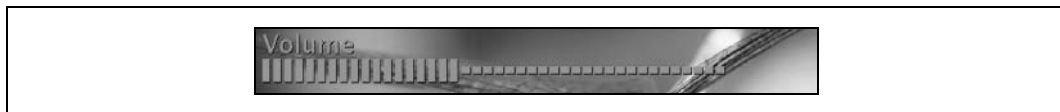
Aktywacja klawiszy do obsługi internetu i multimedialnych na klawiaturze

Program Hotkeys jest już przestarzały i od dawna nie rozwijany, nie obsługuje więc wszystkich typów nowoczesnych klawiatur. Jeśli jednak włoży się odrobinę wysiłku w konfigurację programu Hotkeys, można osiągnąć zamierzony efekt — wykorzystanie dodatkowych klawiszy klawiatury. Jest to prostsze, jeśli program obsługuje klawiaturę użytkownika. W przeciwnym razie konfiguracja jest trudniejsza i w takim przypadku bardziej opłaca się przyrzeć nowocześniejszemu programowi LinEAK [Sposób 29.].

Najpierw należy zainstalować program Hotkeys. Można do tego użyć menedżera pakietów dystrybucji lub skompilować go samodzielnie. Pakiety RPM dla różnych dystrybucji można znaleźć na stronie <http://rpm.pbone.net/index.php3?stat=3&search=hotkeys&srodzaj=3>. Warto również zajrzeć na stronę: <http://rpmfind.net>. Dla przykładu — w dystrybucji Debian pakiety programów Hotkeys i XOSD instaluje się następującym poleceniem:

```
# apt-get install hotkeys xosd-bin
```

Instalując jednocześnie *xosd-bin*, można uzyskać jedną z lepszych funkcji programu Hotkeys — wyświetlanie informacji na ekranie (rysunek 4.1). Aby uzyskać tę funkcję przy samodzielnej kompilacji programu, należy pamiętać o zainstalowaniu XOSD [Sposób 26.].



Rysunek 4.1. Wskaźnik głośności programu Hotkeys

Proces konfiguracji programu Hotkeys należy rozpocząć od skopiowania domyślnej konfiguracji klawiatury do własnego pliku konfiguracyjnego. Domyślny plik konfiguracyjny to zwykle */etc/hotkeys.conf*. Należy utworzyć kopię tego pliku w katalogu domowym użytkownika i dostosować go do używanej klawiatury:

```
$ mkdir ~/.hotkeys
$ cp /etc/hotkeys.conf ~/.hotkeys
```

Poniżej przedstawiona jest lista klawiatur obsługiwanych domyślnie przez program:

```
$ hotkeys -L
Supported keyboards: (with corresponding options to --kbd-list or -l)
mx2500      - Memorex MX2500 Keyboard
kbp8993     - Chicony KBP-8993 Keyboard
sk2500      - Fujitsu/Logitech/Trust SK2500 Keyboard / Liteon-ak2500
sk2505      - SK-2505 Keyboard
sk2800c     - SK-2800C
ibook       - iBook Internal Keyboard
mx1998      - Memorex MX1998 Keyboard
sk2501a     - Silitek SK5210A Keyboard
msnatpro    - Microsoft Natural Keyboard Pro
msnet       - Microsoft Internet Keyboard
logitech-cfo - Logitech Cordless Freedom Optical Keyboard
polypix     - Plypix Keyboard
sk7100      - Silitek SK7100 Keyboard
itouch      - Logitech Cordless iTouch/Internet/Cordless Desktop
hp5181      - HP 5181 Internet Keyboard
msnetpro    - Microsoft Internet Pro Keyboard
acerwl      - Acer Wireless Keyboard
btc9000     - BTC 9000
orktekusb   - ORKTEK USB Hub/keyboard
kb9930      - IBM Rapid Access II Keyboard
mck800      - Process MCK-800
kb9963      - Compaq KB-9963 keyboard
pb5140w     - Packard Bell Model 5140W
mx3000      - Memorex MX3000 Keyboard
sk9925      - Silitek SK-9925 USB Keyboard
```

Lista nie jest długa, ale jeśli klawiatura użytkownika jest tam wymieniona, to nie ma problemu. Jeśli jednak jej nie ma, należy użyć jednej z definicji klawiatury programu Hotkeys jako szablonu do utworzenia nowej definicji. Na przykład, aby utworzyć definicję klawiatury Logitech Elite, należy zalogować się jako użytkownik *root* i wydać następujące polecenie:

```
# cd /usr/share/hotkeys
# cp itouch.def lelite.def
```

Należy dowiedzieć się, jakie są kody dodatkowych klawiszy dla klawiatury Logitech Elite i zmienić ich definicje w pliku. Jest to najtrudniejsza część tego zadania. Najlepszym sposobem na poznanie kodów klawiszy jest wydanie poniższego polecenia w terminalu X:

```
$ xev
```

Powinno pojawić się małe okno z czarnym kwadratem w środku. Należy kliknąć w oknie, aby stało się aktywne. Teraz trzeba przyciskać kolejne klawisze specjalne na klawiaturze. Po naciśnięciu każdego klawisza w terminalu powinna pokazać się informacja o nim. Jeśli po naciśnięciu klawisza nie pojawią się żadne komunikaty, oznacza to, że nie będzie można do tego klawisza przypisać żadnej akcji. Jeśli informacja wygląda podobnie do poniższego zapisu, prawdopodobnie również nie będzie można przypisać akcji do klawisza:

```
KeymapNotify event, serial 30, synthetic NO, window 0x0,
  keys:  4294967222  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0          0  0  0  0  0  0  0  0  0  0  0  0  0  0

ButtonRelease event, serial 30, synthetic NO, window 0x1400001,
  root 0xb6, subw 0x1400002, time 2414137, (35,22), root:(1053,797),
  state 0x200, button 2, same_screen YES
```

Do klawisza można przypisać akcję, gdy informacja o nim jest podobna do poniższego zapisu:

```
KeyPress event, serial 27, synthetic NO, window 0x1400001,
  root 0xb6, subw 0x1400002, time 31223, (44,18), root:(1085,771),
  state 0x0, keycode 162(keysym0x0, NoSymbol), same_screen YES,
XLookupString gives 0 bytes:
XmbLookupString gives 0 bytes:
XFilterEvent returns: False

KeyRelease event, serial 30, synthetic NO, window 0x1400001,
  root 0xb6, subw 0x1400002, time 31447, (44,18), root:(1085,771),
  state 0x0, keycode 162(keysym0x0, NoSymbol), same_screen YES,
XLookupString gives 0 bytes:
```

W tym przykładzie do naciśniętego klawisza przypisany jest kod 162. Należy zapamiętać ten kod i klawisz, który go wywołał. W przypadku klawiatury Logitech Elite naciskany był klawisz *Play*.

Po uzyskaniu informacji o wszystkich klawiszach klawiatury Logitech Elite można rozpocząć edycję pliku z definicją klawiatury. Poniżej znajduje się skrócona wersja pliku `/usr/share/hotkeys/lelite.def` utworzona na podstawie pliku `itouch.def`:

```
<?xml version="1.0"?>

<definition>

  <config model=" Logitech Elite Keyboard">

    <Favorites      keycode="230"/>
    <Shopping        keycode="148"/>
    <PrevTrack       keycode="144"/>
    <Play            keycode="162"/>
    <Stop            keycode="164"/>
    <NextTrack       keycode="153"/>
```

```

<VolUp          keycode="176" adj="2"/>
<VolDown        keycode="174" adj="2"/>
<Mute           keycode="160"/>
<Email          keycode="236"/>
<Search         keycode="136"/>
<Sleep          keycode="223"/>
<Go             keycode="233"/>
<!-- Media      keycode="129"/ -->
<userdef        keycode="129" command="/usr/bin/eject">Eject</userdef>
<!-- My Home    keycode="130"/ -->
<userdef        keycode="130" command="/usr/bin/xfe">Home</userdef>
<!--Go         keycode="233"/ -->
<userdef        keycode="233" command="/usr/bin/firefox \
-remote openURL()"></userdef>

</config>

<contributor>
  <name>nicholas petreley</name>
  <email>nicholas at petreley dot com</email>
</contributor>

</definition>

```

Warto zauważyć, że niektóre definicje zaczynają się poleceniem `userdef`. Program Hotkeys jest bardzo rygorystyczny w stosunku do nazw klawiszy dostępnych na klawiaturze. Nie rozpoznaje niektórych nazw klawiszy i nie potrafi powiązać tych nazw z kodami klawiszy. Można sobie z tym poradzić, tworząc definicje klawiszy użytkownika. W przypadku klawiatury Logitech Elite są to klawisze *Media*, *My Home* i *Go* (spośród innych niezdefiniowanych przez program). Dodałem te klawisze i ich funkcje do pliku `/usr/share/hotkeys/lelite.def`, używając polecenia `userdef` programu Hotkeys:

```

<!-- Media      keycode="129"/ -->
<userdef        keycode="129" command="/usr/bin/eject">Eject</userdef>
<!-- My Home    keycode="130"/ -->
<userdef        keycode="130" command="/usr/bin/xfe">Home</userdef>
<!--Go         keycode="233"/ -->
<userdef        keycode="233" command="/usr/bin/firefox \
-remote openURL()"></userdef>

```

Teraz za pomocą pliku `~/hotkeys/hotkeys.conf` można przypisać akcje, które będą wykonywane przez program Hotkeys za każdym razem, gdy użytkownik naciśnie klawisz specjalny. Oto przykładowa konfiguracja:

```

### Należy określić domyślną klawiaturę (nazwa pliku bez rozszerzenia .def)
### dzięki temu wpisywanie -t przy każdym uruchomieniu programu jest zbędne.
Kbd=lelite

CDROM=/dev/cdroms/cdrom0

# PrevTrack=xmms --rew
# Play=xmms --play-pause
# Stop=xmms --stop
# Pause=xmms --pause
# NextTrack=xmms --fwd
# Rewind=

WebBrowser=firefox
Email=thunderbird

```

```

Calculator=xcalc
FileManager=gentoo
MyComputer=gentoo
MyDocuments=gentoo
Favorites=firefox
Shell=aterm
ScreenSaver=xscreensaver-command -activate
NewsReader= thunderbird -news
Communities=firefox -remote 'openURL(http://slashdot.org,new-tab) '
Search=firefox -remote 'openURL(http://www.google.com,new-tab) '
Idea=firefox -remote 'openURL(http://sourceforge.net,new-tab) '
Shopping=firefox -remote 'openURL(http://thinkgeek.com,new-tab) '
Go=firefox -remote 'openURL()'
Print=lp
osd_font=-adobe-helvetica-bold-r-normal-**-240-**-p-***
osd_color=LawnGreen
osd_timeout=3
osd_position=bottom
osd_offset=

```

Niektóre z przypisanych akcji są proste. Klawisz *Email* otwiera program Thunderbird, więc w pliku wystarczy wiersz `Email=thunderbird`. Jednak inne definicje to bardziej skomplikowane akcje, jak na przykład otwieranie nowego panelu w przeglądarce internetowej (zakładając, że jest już uruchomiona) i ustawianie w nim określonej strony.

Najważniejszą definicją jest `Kbd=lelite` — wskazuje ona programowi Hotkeys plik z definicją klawiatury, który ma zostać użyty. W tym przypadku użyta będzie klawiatura zdefiniowana w pliku `/usr/share/hotkeys/lelite.def`.

Na klawiaturze Logitech Elite nie ma większości klawiszy wypisanych w pliku konfiguracyjnym, więc ich definicje są niepotrzebne. Mi jednak wystarczy, że za pomocą klawiatury mogę sterować głośnością, odtwarzać płyty audio, włączać przeglądarkę Firefox i otwierać w niej wyszukiwarkę Google. Jeśli klawiatura użytkownika jest oficjalnie obsługiwana przez program, możliwości są dużo większe.



SPOSÓB 29.

Włączanie klawiszy szybkiego dostępu za pomocą programu LinEAK

Program LinEAK to nowoczesny sposób na dostosowanie klawiatur multimedialnych do potrzeb użytkownika

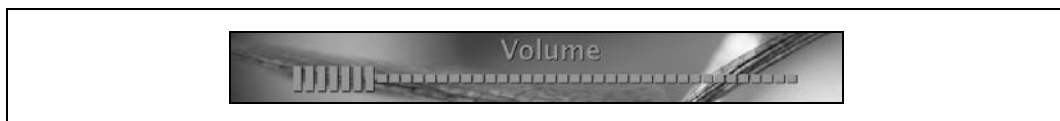
Ten sposób opisuje nowoczesny program będący zamiennikiem programu Hotkeys. LinEAK jest zbiorem pakietów, w skład których wchodzi główny demon programu i kilka modułów rozszerzających. Jest trochę nietypowy i nie zawsze poprawnie się uruchamia (z doświadczenia autora), ale obsługuje znacznie więcej klawiatur niż program Hotkeys, który od lat nie jest rozwijany. Trzeba więc dużo samozaparcia, żeby skonfigurować w nim nowocześniejszą klawiaturę. Z drugiej strony, projekt LinEAK jest regularnie uśprawniany i rozwijany.

W czasie pisania tej książki ze strony <http://lineak.sourceforge.net> można było pobrać pakiety programu dla dystrybucji Debian. Są tam również dostępne kody źródłowe dla wszystkich pakietów wchodzących w skład programu. Można je pobrać i skompilować

samodzielnie. Kompilacja tego programu jest bardzo łatwa. Pakiety RPM dostępne na tej stronie są dość stare, ale możliwe, że w czasie, gdy Czytelnik ma książkę w ręku, pojawiły się nowe.

Póki tak się nie stanie, pakiety RPM programu LinEAK dla dystrybucji Mandrake i innych można znaleźć na stronie <http://rpm.pbone.net/index.php3?stat=3&search=lineakd&srodzaj=3>. Warto też zajrzeć na stronę <http://www.rpmfind.net>. Aby działała funkcja wyświetlania na ekranie, program XOSD i jego biblioteki muszą być zainstalowane w systemie. W sposobie „Jak nie przeoczyć przypomnienia” [Sposób 26.] znajduje się więcej informacji na ten temat.

Przykładowo — wskaźnik głośności pokazany na rysunku 4.2 jest jedną z ciekawych funkcji zarówno programu Hotkeys, jak i LinEAK [Sposób 28.].



Rysunek 4.2. Wskaźnik głośności wyświetlony przez program LinEAK

Następujące programy stanowią zawartość pakietu LinEAK:

- *lineakd*,
- *lineak_defaultplugin*,
- *lineak_xosdplugin*,
- *lineak_kdeplugins*,
- *Media Detect*,
- *lineakconfig*,
- *Klineakconfig*.

Niezbędnym minimum wymaganych do działania programu są: *lineakd*, *lineak_defaultplugin* i *lineak_xosdplugin*. Jeśli nie ma pakietów dla dystrybucji użytkownika, należy pobrać kod źródłowy, skompilować i zainstalować program poniższymi poleceniami (zakładając, że kod źródłowy znajduje się w */usr/local/src*):

```
# cd /usr/local/src/lineakd-0.8.1
# ./configure
# make
# make install
# cd /usr/local/src/lineak_defaultplugin-0.8.1
# ./configure
# make
# make install
# cd /usr/local/src/lineak_xosdplugin-0.8.1
# ./configure
# make
# make install
```

Aby obejrzeć listę klawiatur obsługiwanych przez program, należy zalogować się jako zwykły użytkownik i wpisać poniższe polecenie:

```
$ lineakd -l
(Cała lista ma 3 strony, to jest tylko kilka początkowych wierszy)
LinEAK v0.8.1 -- supported keyboards:

[TYPE]           [Full name]
A4-KBS21         A4Tech Wireless Desktop KBS-21533RP & Office/Multimedia
Keyboard
A4-KBS8          A4Tech KBS-8
A4-RFKB23        A4Tech RFKB-23
A4-RFKB25        A4Tech RFKB-25 (KBS-2548RP & KBS-2548RPC)
ACE-6512UV       Acer 6512-UV
ACE-TM290        Acer Laptop/notebook Travelmate 290LCi
ACEAKV12         Acer AirKey V (12 keys)
ADEL-9805        Adesso EL-9805
APK7             Apple Pro Keyboard (7 keys)
BEN-AM805        BenQ AM805
```

Lista ta wygląda imponująco w porównaniu z listą klawiatur obsługiwanych przez program Hotkeys.

Należy znaleźć swoją klawiaturę na liście. Przykładowo — kod klawiatury Logitech Elite to LTCElite. Następnie trzeba utworzyć domyślny plik konfiguracyjny dla klawiatury użytkownika następującym poleceniem (należy wpisać własny kod klawiatury):

```
$ lineakd -c LTCElite
```

Opcja `-c` tworzy domyślny plik konfiguracyjny *lineakd.conf* dla klawiatury Logitech Elite i umieszcza go w katalogu `~/lineak`. Ustawienia pliku można zmieniać według upodobań. Oto plik dla mojej klawiatury:

```
# LinEAK - Linux support for Easy Access and Internet Keyboards
# Copyright (c) 2001, 2002, 2003 Sheldon Lee Wen <leewsb@hotmail.com>
#                                     and Mark Smulders <Mark@PIRnet.nl>
# http://lineak.sourceforge.net
#
# example key configuration:
#           play           = "xmms --play-pause"
#           eject          = EAK_EJECT
#
# Lineak supports the following modifier keys:
# control alt shift mod2 mod3 mod4 mod5

# Normally /dev/cdrom, but UDEV likes /dev/cdroms/cdrom0
CdromDevice = /dev/cdroms/cdrom0
Display_align = center
Display_color = 77FF00
Display_font = "-adobe-helvetica-bold-r-normal-*--240---p-*--*"
Display_hoffset = 0
Display_plugin = xosd
Display_pos = bottom
Display_soffset = 1
Display_timeout = 6
Display_voffset = 50
Keyboard Type = LTCElite
MixerDevice = /dev/mixer
```

```

Screensaver =
conffilename = /home/nicholas/.lineak/lineak.conf
keystate_capslock =
keystate_numlock =
keystate_scrolllock =

Arrow =
Email = "thunderbird"
Favorities = "firefox"
Go = "firefox -remote 'openURL()' "
Media = "cdeject"
Messenger =
Mute = "EAK_MUTE"
MyHome =
Next = "cdplay +"
Play = "cdplay"
Pause = "cdpause"
Previous = "cdplay -"
Search = "firefox -remote 'openURL(http://www.google.com,new-tab)'"
Shopping =
Sleep =
Stop = "cdstop"
VolumeDown = "EAK_VOLDOWN"
VolumeUp = "EAK_VOLUP"
Webcam =
iTouch =

```

Należy zauważyć, że używane są wbudowane polecenia takie jak `EAK_MUTE`, `EAK_VOLUP` i `EAK_VOLDOWN`. Niestety LinEAK podczas tworzenia pliku konfiguracyjnego nie wstawia domyślnie tych poleceń dla parametrów `Mute`, `VolumeUp` i `VolumeDown`, pozostawiając puste miejsca. Użytkownik musi wpisać je sam, a jeśli nie zna tych poleceń, może stracić dużo czasu na ich wymyślanie.

Poza wewnętrznymi poleceniami `EAK_`, w powyższym pliku wykorzystano pakiet programów *cdtool* do obsługi płyt audio. Pakiet ten ma wielu twórców, obecnie zajmuje się nim Max Vozeler. Program *cdtool* można pobrać ze strony <http://hinterhof.net/cdtool>. Umożliwia on sterowanie dowolnym napędem CD z wiersza poleceń. Używanie graficznego interfejsu umożliwiającego sterowanie napędem CD staje się zbędne, jeśli wszystkie potrzebne funkcje dostępne są z klawiatury.



Jeśli w jednym systemie używane są programy Hotkeys i LinEAK, po skończeniu pracy z Hotkeys należy ponownie uruchomić pulpit. Nawet jeśli proces Hotkeys zakończy się poleceniem *kill*, zostawia on pulpit w stanie, który uniemożliwia poprawną pracę programowi LinEAK.

Autorem programu jest Mark Smudlers (msmudlers@elsar.nl). Można mu wysłać podziękowania za stworzenie użytecznego oprogramowania. Sheldon Lee Wen (leewsb@hotmail.com) obecnie zarządza projektem i publikuje nowe wersje, pisze także moduły rozszerzające. Opiekunem projektu jest Phil Woodland (sir_taco@yahoo.ca), natomiast Chris Peterson (rpm@forevermore.net) przygotowuje pakiety RPM.



SPOSÓB

30.

Dostęp do systemów Windows i Mac OS X z Linuksa

Nie ma potrzeby przesiadania się do innego komputera w celu uzyskania dostępu do innych systemów operacyjnych

Chociaż wielu użytkowników wychwala zalety Linuksa i darmowego oprogramowania, czasami nadal zachodzi potrzeba skorzystania z innych systemów, jak Windows lub Mac OS X. Niektóre potrzebne programy mogą być uruchomione tylko w określonym systemie operacyjnym, pracodawca nakazuje używanie jednego systemu lub napisany program trzeba sprawdzić na różnych platformach. Dla niektórych rozwiązaniem jest ogromne biurko, na którym mieszczą się trzy komputery, monitory, klawiatury i myszy. Jednak istnieje też inny sposób.

W tym sposobie będzie wykorzystany program VNC (ang. *Virtual Network Computer*). To użyteczne narzędzie pozwala przekierować informacje wyświetlane na ekranie na inny komputer w sieci. Pozwala również na sterowanie komputerem za pomocą myszy i klawiatury podłączonych do innego komputera. Dzięki temu można uruchomić serwer VNC na komputerze z systemem Windows i wyświetlać jego ekran w Linuksie. Można też zrobić odwrotnie — uruchomić serwer VNC w Linuksie i wyświetlić jego ekran w systemie Windows. Program VNC dostępny jest dla większości uniksowych systemów operacyjnych, takich jak Linux, FreeBSD, Solaris itd., jak również dla systemów Windows i Mac OS X. Program VNC daje możliwość połączenia tych systemów na jednym monitorze.

Konfiguracja serwera VNC w Linuksie

Istnieje kilka wersji programu VNC, ale największą uwagę przyciągają RealVNC i TightVNC. Z tych dwóch TightVNC wydaje się być lepiej zoptymalizowany i szybszy. Można go pobrać ze strony <http://www.tightvnc.com/download.html>. Na stronie dostępne są różne pakiety dla obsługiwanych systemów. Do zestawienia połączenia VNC potrzebny będzie serwer (udostępnia komputer w sieci) oraz przeglądarka (ang. *viewer*) (pozwala na połączenie się z udostępnionym komputerem). Używając menedżera pakietów dystrybucji, należy zainstalować najnowszą wersję programu TightVNC.

Aby w trakcie pracy w systemie Mac OS X mieć dostęp do pulpitu Linuksa, potrzebny jest pakiet OSXvnc, który można znaleźć na stronie <http://www.redstonesoftware.com/vnc.html> (RealVNC i TightVNC nie obsługują systemu Mac OS X). Klienta VNC dla Macintosha można też pobrać przez system pakietów Fink ze strony <http://fink.sourceforge.net>.

Aby włączyć serwer VNC w Linuksie, należy uruchomić serwer i przydzielić mu specjalny numer ekranu, poprzez który będzie dostępny z zewnątrz. Numery zwykle zaczynają się od 1 i rosną o 1 dla każdego nowo utworzonego serwera. Przykładowo — jeśli serwer jest uruchomiony na komputerze 192.168.0.2, to dostęp do pierwszego ekranu uzyskuje się poprzez połączenie z adresem 192.168.0.2:1. Aby uruchomić serwer, należy określić rozdzielczość ekranu i głębię kolorów poleceniem z opcjami `-geometry` i `-depth`:

```
foo@bar:~$ vncserver -geometry 1024x768 -depth 24
```

Te ustawienia to parametry wirtualnego ekranu, a nie prawdziwe ustawienia na komputerze z serwerem. Oryginalny ekran może mieć rozdzielczość 1280×1024 z 8-bitową głębią koloru, ale można go zdalnie udostępnić z ustawieniami 1024×768 i 8-bitową głębią. Oczywiście komputer, na którym jest wyświetlany udostępniony ekran, musi obsługiwać te ustawienia.



Można ustawić niestandardową rozdzielczość, np. 990×745. Dzięki temu zdalny ekran nie zasłoni pasków zadań i paneli na lokalnym pulpicie.

Podczas pierwszego uruchamiania serwera należy podać hasło. Zabezpiecza ono przed niepożądanym dostępem do serwera. Jest zapamiętywane i pozostaje niezmienione podczas używania serwera (hasło można zmienić poleceniem `vncpasswd`). Po wpisaniu właściwego hasła serwer wyświetla swój numer.

Gdy serwer pracuje, wszystkie uruchomione aplikacje pojawiają się na monitorze podłączonego komputera i równocześnie na monitorze komputera z uruchomionym serwerem (jeśli monitor jest podłączony). Można też wyświetlić aplikacje jedynie na serwerze VNC, używając zmiennej środowiskowej `DISPLAY` oraz określając adres komputera i numer serwera:

```
foo@bar:~$ mozilla -display 192.168.0.2:1 &
```

Aby wyłączyć serwer VNC, należy użyć opcji `kill` i przypisanego numeru ekranu:

```
foo@bar:~$ vncserver -kill :1
```

Połączenie z serwerem VNC

Aby połączyć się z serwerem VNC z komputera z Linuksem, można użyć narzędzia `vncviewer`, dołączonego do oprogramowania VNC. Tego prostego programu używa się w następujący sposób:

```
foo@bar:~$ vncviewer 192.168.0.2:1
```

W poleceniu określony jest adres IP komputera, a po dwukropku (:) numer ekranu, z którym ma być nawiązane połączenie. Po wydaniu polecenia pojawia się pytanie o hasło do serwera VNC. Po podaniu poprawnego hasła zostaje wyświetlony ekran serwera.

Konfiguracja serwera VNC w systemie Windows

Instalacja serwera VNC w systemie Windows jest dość łatwa. Może on być skonfigurowany jako usługa systemu Windows — będzie wtedy uruchamiany przy starcie systemu (jak demon w Linuksie). Dzięki temu serwer będzie dostępny nawet wtedy, gdy komputer jest zablokowany lub gdy użytkownik się wylogował.

Wersję instalacyjną dla systemu Windows należy pobrać ze strony internetowej programu TightVNC. Jest to typowy instalator.

Aby serwer VNC był usługą, w czasie instalacji należy zaznaczyć każdą opcję związaną z *VNC Server System Service*. Po zakończeniu instalacji pojawi się okno dialogowe *Server Options* i, aby uruchomić serwer, trzeba skonfigurować przynajmniej zakładkę *Authentication*. Na tej zakładce należy wybrać opcję *VNC 3.3 Authentication* i wpisać w polu *Set Password* hasło dostępu do serwera. Nie powinno się wyłączać opcji uwierzytelnienia, jeśli nie ma całkowitej pewności, że sieć jest bezpieczna.

Aby włączyć lub wyłączyć serwer, należy użyć narzędzia do konfiguracji usług uruchamiania i zatrzymywania usługi.

Konfiguracja serwera VNC w systemie Mac OS X

Oficjalna dystrybucja VNC nie obsługuje systemu Mac OS X, jednak na stronie <http://www.redstonesoftware.com/vnc.html> dostępna jest wersja dla tego systemu, zwana *OSXvnc*. Oprogramowanie jest dostępne w formacie obrazu dysku Mac OS X (*.dmg*). Należy pobrać oprogramowanie i podwójnie kliknąć je w *Finderze*. Pojawi się okno z programem. Należy przeciągnąć program na pulpit. Następnie po podwójnym kliknięciu ikony programu wyświetli się okno dialogowe *VNC Settings*. Domyślne ustawienia wystarczają do uruchomienia serwera, więc aby rozpocząć pracę, można kliknąć przycisk *Start Server*.

Wyświetlanie pulpitu w przeglądarce internetowej

Ciekawą funkcją serwera VNC jest mały serwer WWW. Eksportuje on pulpit VNC do przeglądarki, używając specjalnego apletu Javy. Aby wyświetlić pulpit, należy za pomocą przeglądarki z włączoną obsługą apletów Javy połączyć się z portem 5801. Numer portu jest dołączany do adresu komputera, a całość podana jest w formie zwykłego adresu www:

`http://foo.com:5801`

Numer portu określa numer uruchomionego serwera VNC, z którym zostanie nawiązane połączenie. Jeżeli uruchomiony jest serwer z numerem 1, należy wpisać jako numer portu 5801, jeśli serwer 2 to port numer 5802 itd.



W niektórych dystrybucjach (np. w Debianie) aplet Javy znajduje się w innym pakiecie niż serwer VNC.

Przy uruchamianiu serwera przez internet, należy się upewnić, że jest on dobrze chroniony przez firewall, blokujący dostęp do serwera niepowołanym osobom. Jeśli zapora sieciowa jest już uruchomiona, należy upewnić się, że porty 5800 – 5805 są dostępne. Aby jeszcze zwiększyć bezpieczeństwo, należy otworzyć jedynie port 5801 i używać tylko serwera z numerem 1. Można też włączyć tunelowanie połączenia VNC za pomocą SSH [Sposób 32].



SPOSÓB

31.

Udostępnianie pulpitu przez internet

Uruchamianie środowiska graficznego na zdalnym komputerze przez internet

Zdalny dostęp do komputera może być potrzebny z wielu powodów. Na przykład podczas podróży z laptopem pojawia się konieczność edycji ważnego pliku, który został w domu. Trzeba go zmienić, a potrzebny do tego program jest na komputerze domowym. Dobrze by było połączyć się z komputerem w domu przez internet i zmienić plik, używając oprogramowania na tym komputerze, a potem zapisać wynikowy plik na laptopie. System X Window ma wbudowaną przezroczystość sieciową, która pozwala uruchamiać program na jednym komputerze, a wyświetlać na drugim [Sposób 32.]. Różne wersje protokołu VNC pozwalają wyświetlić pulpit na innym komputerze [Sposób 30.]. Choć każda z tych metod ma swoje przeznaczenie, w którym doskonale się sprawdza, to obie mają pewną wadę — potrzebują bardzo wysokiej przepustowości łącza. Zdalne programy X i pulpity VNC działają wolno nawet przy połączeniu DSL lub przy stałym łączy kablowym.

Program NX firmy NoMachine (<http://www.nomachine.com>) jest dodatkiem do systemu X, który uruchamia zdalnie aplikacje X. Może być użyty do uruchomienia zdalnego pulpitu i działania niemal w czasie rzeczywistym, nawet przy łączy 56 kbps. NX działa podobnie do pamięci podręcznej proxy dla protokołu X. Obniża to znacznie ruch w sieci. Wszystkie komponenty i biblioteki NX oraz aplikacja klienta dostępne są jako *open source*. Programy serwerowe firmy NoMachine są płatne, dostępna jest jednak wersja testowa, pozwalająca na obsługę jednego klienta.

Instalacja serwera i klienta NX

Na komputerze, na którym ma być uruchomiona zdalna sesja, należy zainstalować aplikacje serwera i klienta NX. Na stronie <http://www.nomachine.com> dostępne są pakiety dla poszczególnych dystrybucji. Należy pobrać *NX Server Personal Edition* oraz oprogramowanie klienckie dla odpowiedniej dystrybucji i zainstalować je. Na przykład:

```
foo@bar:~# rpm -i nxserver-1.4.0-99.i386.rpm
foo@bar:~# rpm -i nxclient-1.4.0-75.i386.rpm
```

Na komputerze, na którym ma być wyświetlana zdalna sesja, należy zainstalować klienta NX. Programy-klienty dla Linuksa, Mac OS X i Windows dostępne są na tej samej stronie, co serwer.

Uruchomienie serwera NX

Po zainstalowaniu NX można uruchomić serwer NX poniższym poleceniem (jako użytkownik *root*):

```
foo@bar:~# /usr/NX/bin/nxserver --start
```

Następnie należy dodać sesję użytkownika do serwera następującym poleceniem:

```
foo@bar:~# /usr/NX/bin/nxserver --useradd nazwa_uzytkownika
```

Parametr `nazwa_uzytkownika` to nazwa użytkownika istniejącego już w systemie, w którym pracuje serwer NX. Po wydaniu powyższego polecenia program pyta o hasło do tej sesji. Po wpisaniu go, można połączyć drugi komputer z serwerem, używając klienta NX. Aby to zrobić, należy przejść do innego komputera i uruchomić klienta poniższym poleceniem:

```
foo@bar:~# /usr/NX/bin/nxclient
```

Przy pierwszym uruchomieniu pojawia się okienko kreatora, do którego należy wpisać szczegółowe informacje o połączeniu. Na drugiej stronie kreatora trzeba wpisać nazwę sesji, numer IP serwera NX oraz wybrać rodzaj połączenia między klientem a serwerem.

Na trzeciej stronie kreatora można wybrać typ protokołu (klient NX może pracować także z serwerami RDP i VNC), typ sesji oraz rozmiar okna, w którym będzie wyświetlony zdalny pulpit. Jeśli jest włączony firewall [**Sposób 81.**] i dostęp do serwera jest możliwy tylko poprzez SSH, należy zaznaczyć opcję *Enable SSL encryption of all traffic*. Jeśli ruch nie ma być tunelowany przez SSH, dla pierwszej sesji należy otworzyć porty 1000, 5000 i 7000, dla drugiej: 1001, 5001 i 7001 itd. — zawsze po trzy porty dla jednej sesji.

Na czwartej stronie kreatora można utworzyć skrót na pulpicie do tej sesji i zmienić zaawansowane ustawienia. Gdy kreator zakończy pracę, powinno pokazać się okno dialogowe z wypełnionymi polami *Login* i *Session*.

Należy wpisać wcześniej ustawione hasło do serwera NX i kliknąć *Login*. Po uzyskaniu połączenia i autoryzacji powinien zacząć działać zdalny pulpit na komputerze klienckim.

Inne polecenia serwera NX

Kiedy użytkownik potrafi już nawiązać podstawowe połączenie, może zainteresować się kilkoma użytecznymi poleceniami serwera NX. Muszą być one wydawane przez użytkownika *root*. Aby wyświetlić listę wszystkich użytkowników NX, należy wpisać:

```
foo@bar:~# /usr/NX/bin/nxserver --userlist
```

Aby wykasować użytkownika NX (polecenie kasuje użytkownika tylko z NX, a nie z systemu), należy wpisać:

```
foo@bar:~# /usr/NX/bin/nxserver --userdel nazwa_uzytkownika
```

Parametr `nazwa_uzytkownika` jest nazwą konta w systemie NX, które ma być wykasowane.

— Paul Cooper



SPOSÓB
32.

Zdalny dostęp do programów

Pełny dostęp do komputera i zainstalowanych na nim programów przez połączenie sieciowe

System graficzny X Window, który jest fundamentem KDE, GNOME i innych środowisk graficznych, od wielu lat ma funkcje sieciowe, o jakich inne systemy mogą tylko pomarzyć. Pozwalają one nie tylko na uruchomienie programu zainstalowanego na zdalnym komputerze, ale także na uruchomienie całego pulpitu z innego komputera.

Za pomocą tej funkcji można korzystać z dowolnego komputera, na którym można uruchomić podstawową sesję X, a środowisko graficzne uruchamiać z szybszego komputera. Przy dużej przepustowości łącza nie ma zauważalnej utraty prędkości przy takiej pracy. W rzeczywistości, ponieważ wszystkie programy pracują na szybszym komputerze, wydajność powinna być dużo lepsza.

Pomimo tych imponujących zalet, funkcje sieciowe systemu X są ignorowane przez większość użytkowników. Niewielu eksperymentuje z możliwościami uruchamiania programów przez sieć, mimo że ta funkcja zaoszczędziłaby im wielu spacerów między komputerami.



Końcowa praca nad książkami O'Reilly przebiega przy użyciu programów uruchomionych zdalnie przez połączenie sieciowe. Dla przykładu — program FrameMaker 5.5.6 jest uruchomiony na serwerze Solaris i wykorzystywany na innych komputerach-klientach. Wszystkie komputery z Linuksem to słabe maszyny, które uruchamiają swoje środowiska graficzne przez połączenie sieciowe, używając systemu X.

Bezpieczny dostęp do programów

Uruchamianie aplikacji X z innego komputera jest dość łatwym procesem. Wystarczy zmienić zmienną środowiskową `$DISPLAY` (która wskazuje miejsce, do którego ma być przekierowany wynik aplikacji X) tak, aby wskazywała inny komputer w sieci, na przykład:

```
foo@bar:~$ export DISPLAY="192.168.0.1:0.0"
```

Gdy aplikacja zostanie uruchomiona, jej wynik pokaże się na monitorze komputera o wskazanym adresie IP, a nie na lokalnym monitorze. Tego typu wysyłanie obrazu jest funkcją klienta X. Natomiast wyświetlaniem obrazu zajmuje się serwer X. Aby komputer o podanym adresie IP wyświetlił przekazany obraz, musi być na nim uruchomiony serwer X, który akceptuje połączenia z zewnętrznego źródła. Aby zdalny komputer akceptował wysyłane obrazy X z innego komputera, należy użyć polecenia `xhost`:

```
foo@bar:~$ xhost +
access control disabled, clients can connect from any host
```

Znak plus (+) oznacza, że będą akceptowane połączenia ze wszystkich komputerów. Można również dokładnie określić komputer, którego próby połączenia mają być zaakceptowane, wpisując jego nazwę DNS lub adres IP po znaku plus.

Chociaż za pomocą tej techniki można zdalnie uruchamiać aplikacje X, ruch między komputerami jest niezaszyfrowany i nie jest bezpieczny. Cały ten ruch odbywa się na porcie 6000. Jeśli funkcja ma być też dostępna z zewnątrz sieci, trzeba otworzyć kolejny port. Tworzy się dziura w firewallu i sieć jest bardziej narażona na ataki. Rozwiązaniem tego problemu jest szyfrowanie sesji X, co podniesie poziom bezpieczeństwa. Protokół SSH (ang. *Secure SHell*) ma wbudowaną obsługę zdalnych aplikacji X i jest łatwy w użyciu.

Większość dystrybucji Linuksa instaluje SSH domyślnie. Jeśli nie jest zainstalowany, należy skorzystać z menedżera pakietów dystrybucji i zainstalować pakiety *openssh*.

Na komputerze, który będzie wysyłał aplikacje X, należy otworzyć plik */etc/ssh/sshd_config* i upewnić się, że opcja *X11Forwarding* jest włączona:

```
X11Forwarding yes
```

W wielu domyślnych instalacjach serwera SSH opcja ta oznaczona jest jako komentarz. Aby ją włączyć, należy usunąć znak komentarza (#). Po wprowadzeniu zmian do pliku konfiguracyjnego należy ponownie uruchomić serwer SSH. Można to zrobić następującym poleceniem:

```
foo@bar:~$ /etc/init.d/ssh restart
```

Teraz można uruchomić aplikację na zdalnym komputerze z opcją *-X* przekierowującą połączenia do serwera X. Poniższy przykład pokazuje sposób, w jaki można uruchomić program Gimp na komputerze 192.168.0.2:

```
foo@bar:~$ ssh -X jono@192.168.0.2 gimp
```

Aby włączyć na stałe przekierowanie połączeń z serwerem X na komputerze lokalnym, należy zmienić plik */etc/ssh/sshd_config*. Trzeba dodać do niego poniższy wiersz (lub usunąć znak komentarza znajdujący się na początku wiersza):

```
ForwardX11 yes
```

Komputer lokalny będzie zawsze uruchamiał sesję SSH z włączonym przesyłaniem X. To coś innego niż wcześniejsze polecenie włączające zgodę na przesyłanie X na zdalnym komputerze. Tym razem nie ma potrzeby ponownego uruchamiania demona *sshd*. Zmiana będzie zastosowana automatycznie dla wszystkich nowych połączeń SSH.

Jeśli przy próbie uruchomienia zdalnej aplikacji X pojawią się błędy, należy włączyć „pewne” przekierowania poprzez dodanie opcji *-Y* przy łączeniu się przez SSH. Jeśli ta metoda działa, można włączyć ją na stałe, dodając do pliku */etc/ssh/sshd_config* poniższy wiersz:

```
ForwardX11Trusted yes
```

Dostęp do całego pulpitu

Zdalne uruchamianie aplikacji jest bez wątpienia przydatne, ale naprawdę imponującą funkcją jest możliwość uruchomienia całego pulpitu zdalnego komputera na lokalnym. Aby to osiągnąć, potrzebna będzie część systemu X zwana XDMCP (ang. *X Display Manager Control Protocol*). Protokół ten zezwala innym komputerom na dostęp do menedżera logowania KDM, GDM lub XDM, który włącza zdalne środowisko graficzne. Przy szybkim połączeniu sieciowym, 10 Mb/s lub więcej, wolny komputer będzie pracował tak szybko, jak szybka maszyna zdalna. Jest to możliwe, ponieważ komputer lokalny służy tylko do wyświetlania informacji. Prawdziwa praca jest wykonywana na szybszej zdalnej maszynie.

Aby skorzystać z XDMCP, na zdalnym komputerze musi być uruchomiony jeden z menedżerów logowania: KDM, GDM lub XDM. Każdy z nich obsługuje protokół XDMCP, trzeba tylko aktywować tę opcję.

Aby włączyć XDMCP w programie GDM, potrzebne jest narzędzie *gdmconfig*. Po jego uruchomieniu na zakładce XDMCP należy zaznaczyć opcję *Uaktywnienie XDMCP*. Po wprowadzeniu zmian należy ponownie uruchomić GDM.

Nie ma sposobu na włączenie protokołu XDMCP w KDM, korzystając z interfejsu graficznego, zmiany należy więc wprowadzić do pliku *kdmrc*. Istnieją dwa takie pliki — jeden ogólny dla całego systemu (prawdopodobnie w */etc/kde3/*) i drugi dla konkretnego użytkownika (*~/kde/share/config*). Wewnątrz pliku znajduje się fragment `[Xdmcp]`, w którym opcję `Enable` należy ustawić na `true`. Po zapisaniu zmian trzeba ponownie uruchomić KDM.

Po włączeniu obsługi XDMCP należy wrócić do komputera lokalnego i poszukać w sieci komputerów zezwalających na połączenia za pomocą protokołu XDMCP. Należy wylogować się z systemu X i wydać następujące polecenie:

```
foo@bar:~$ X -broadcast
```

Uruchomi się system X i pojawi się ekran logowania zdalnego komputera. Można się zalogować i używać komputera tak jak zwykle. Aby połączyć się z określonym komputerem w sieci, należy wydać poniższe polecenie na komputerze lokalnym (należy zmienić adres IP na odpowiedni):

```
foo@bar:~$ X -query 192.168.0.2
```



SPOSÓB
33.

Dodanie głębi do pulpitu

Cienie dodają wrażenie głębi wszystkim oknom na pulpicie, a efekty 3D są naprawdę fascynujące

Funkcje programu *composite manager* (*xcompmgr*) łączą się z niektórymi nowymi funkcjami systemu Xorg. Dzięki nim można dodać atrakcyjne cienie do okien i menu. Dają także możliwość powolnego znikania i pojawiania się obiektów na pulpicie zamiast standardowego — natychmiastowego.

Z narzędzia *xcompmgr* można skorzystać tylko wtedy, gdy jest używany system Xorg — gałąź projektu XFree86. Nie można go używać z żadną wersją XFree86, nie działa także ze starą wersją Xorg. Aby sprawdzić, jaka wersja serwera X jest używana, należy wydać poniższe polecenie:

```
# X -version
X Window System Version 6.8.0
```

Sposób ten będzie działał z Xorg w wersji co najmniej 6.8.0.

Pobieranie composite manager

Composite manager łączy się z nowymi funkcjami systemu Xorg. Pozwala to na mieszanie ze sobą różnych elementów ekranu. Na przykład, poprzez wymieszanie tła z zawartością okna, może ono być częściowo przezroczyste. Niektóre programy także oferują tę funkcję, np. częściowo przezroczyste terminale X. Nie korzystają one jednak z funkcji Xorg, a jedynie symulują to sprytnymi sztuczkami graficznymi. Efekt jest podobny, ale znacznie różni się prędkością. Funkcje Xorg są częścią silnika X11, więc efekty są tworzone i renderowane znacznie szybciej.

Aby wypróbować te funkcje, należy zainstalować program *xcompmgr*. Program jest w fazie tworzenia i może być jeszcze niestabilny, więc prawdopodobnie nie ma dla niego standardowego pakietu. Wyszukiwarka Google znajduje pakiety RPM z programem *xcompmgr* na stronie <http://rpm.pbone.net>.

Jeśli na tej, ani na żadnej innej stronie nie ma gotowego pakietu dla dystrybucji Czytelnika, należy skompilować go samodzielnie. Nie ma gwarancji, że kompilacja się powiedzie, ponieważ kod jest w fazie tworzenia. Aby pobrać kod źródłowy, należy połączyć się z serwerem CVS. Należy zalogować się jako użytkownik *root* i przejść do katalogu, w którym ma być zapisany kod, np. */usr/local/src*. Następnie należy wydać poniższe polecenia (nie jest potrzebne hasło; po pytaniu o nie należy wcisnąć klawisz *Enter*):

```
# cvs -d :pserver:anoncvs@cvs.freedesktop.org:/cvs/xorg login
CVS password: (wcisnąć Enter)
# cvs -d :pserver:anoncvs@pdx.freedesktop.org:/cvs/xapps co xcompmgr
# cd xcompmgr
# sh autogen.sh
# ./configure
# make
# make install
```

Uruchamianie środowiska graficznego lub menedżera okien

Program *xcompmgr* nie współpracuje ze wszystkimi środowiskami graficznymi i menedżerami okien. Zauważyłem, że nie działa dobrze z programem Metacity — domyślnym menedżerem okien w GNOME. Program Metacity ma wbudowane niektóre możliwości *xcompmgr*, ale nie udało mi się do nich dostać. Niestety nie można wywołać programu *xcompmgr* przed uruchomieniem środowiska lub menedżera, należy więc to zrobić, gdy środowisko jest już uruchomione.

Strona man programu *xcompmgr* objaśnia różne możliwości tego programu. Efekt zastosowania poniższej kombinacji przełączników powinien zadowolić prawie wszystkich. Należy otworzyć terminal X i wpisać w nim następujące polecenie:

```
$ xcompmgr -cCfF -l 0 -t 0 -r 5 -o .6 &
```

Jeśli polecenie zadziała prawidłowo, zawartość ekranu na chwilę zniknie, a następnie wszystkie otwarte okna będą rzucały cień. Inne elementy na ekranie, np. menu, również mają cień, a także powoli znikają i pojawiają się. Efekt ten pokazano na rysunku 4.3.



Rysunek 4.3. Program Firefox i menu z cieniami

Widoczne efekty zostały ustawione przełącznikami polecenia `xcompmgr`. Przełączniki `-l 0` i `-t 0` wskazują, że cień szerokości 0 pikseli (więc bez cienia) ma znajdować przy lewej (`-l 0`) i przy górnej (`-t 0`) krawędzi okna. Oznacza to, że cień pojawi się tylko przy prawej i dolnej krawędzi tak, jakby źródło światła znajdowało się u góry po lewej stronie. Standardowo program tworzy cień wokół całego okna. Efekt ten nie jest zły, ale inne obiekty na ekranie, np. kursor myszy, rzucają cień w stronę prawego dolnego rogu. Powyższe ustawienie cienia sprawia, że dla wszystkich obiektów źródło światła jest umieszczone w tym samym miejscu.

Przełączniki `-r 5` i `-o .6` ustawiają promień cienia na 5 i zaciemnienie na 0,6, dzięki czemu cienie są delikatniejsze.

Kombinacja przełączników `-cCfF` niesie ze sobą bardziej ogólne ustawienia. Przełącznik `c` nakazuje włączenie cieni i przezroczystości, `C` wskazuje, aby nie tworzyć cienia dla elementów takich jak panele uruchamiania. Dzięki zastosowaniu kombinacji `fF` wszystkie elementy powoli pojawiają się i zanikają.



Program `xcompmgr` jest w trakcie rozwoju, więc jest całkiem możliwe, że pracuje teraz inaczej niż w chwili pisania książki. Gdy książka była wysyłana do druku, dostępna już była przynajmniej jedna nowa wersja programu, która pobierała ustawienia z pliku `~/xcompmgrrc` zamiast z wiersza poleceń.



SPOSÓB

34.

Rentgen pulpitu

Przez okna na monitorze można patrzeć tak samo jak przez okna w ścianie

Ten sposób nie pozwoli na patrzenie przez czyjeś ubranie, ale dzięki jego zastosowaniu aplikacje staną się przezroczyste. Wszystko, co będzie pod nimi, będzie widoczne jak na zdjęciu rentgenowskim. Należy tylko wydać polecenie i kliknąć wybrane okno, a stanie się ono przezroczyste.

Żeby ten sposób zadziałał, potrzebna jest najnowsza wersja systemu Xorg. Należy również zainstalować program *xcompmgr* opisany w sposobie „Dodanie głębi do pulpitu” [Sposób 33.]. Sposób nie zadziała, jeśli w tle nie będzie uruchomiony *xcompmgr*.

Gdy program *xcompmgr* jest uruchomiony, potrzebny będzie program *transset*. Ten program również jest w fazie tworzenia, więc prawdopodobnie nie istnieje pakiet dla dystrybucji Czytelnika. Niektóre pakiety RPM programu *xcompmgr* zawierają też program *transset*, w innym przypadku należy poszukać oddzielnego pakietu z programem. Jeśli pakietu nigdzie nie ma, można pobrać kod źródłowy programu i samodzielnie go skompilować. Poniżej przedstawiono proces pobierania i instalowania programu (przy pytaniu o hasło należy wcisnąć klawisz *Enter*):

```
# cvs -d :pserver:anoncvs@cvs.freedesktop.org:/cvs/xorg login
CVS password: (wcisnąć Enter)
# cvs -d :pserver:anoncvs@pdx.freedesktop.org:/cvs/xapps co transset
# cd transset
# make
# chmod +x transset
# cp transset /usr/bin
```

Należy uruchomić menedżer okien lub środowisko graficzne i włączyć program *xcompmgr* (w poprzednim sposobie przedstawiono przykładowe polecenie uruchamiające program *xcompmgr*). Teraz trzeba tylko wydać polecenie *transset* z parametrem określającym poziom przezroczystości. Przykładowo — w terminalu można wpisać następujące polecenie:

```
# transset 0.7
got arg 0.7
d is 0.7
```

Kursor powinien zamienić się w krzyżyk. Należy kliknąć w oknie, które ma stać się przezroczyste, najlepiej w jego belkę tytułową. Teraz przez okno powinna być widoczna tapeta lub inne okna. Rysunek 4.4 pokazuje przezroczysty terminal *aterm*. Terminal ten nie ma wbudowanej funkcji przezroczystości, nie można więc uzyskać tego efektu inaczej niż za pomocą programu *transset*.

W terminalu, w którym wydane było polecenie *transset*, powinien ukazać się poniższy tekst:

```
opacity 0xb3333332
Set Property to 0.7
```



Rysunek 4.4. Terminal *aterm* z prawdziwą przezroczystością

Aby uzyskać jeszcze bardziej ekscytujący efekt (przy założeniu, że menedżer okien użytkownika wyświetla zawartość okien podczas ich przeciągania), można przeciągnąć okno po pulpicie. Widać teraz, że to nie jest „sztuczna” przezroczystość — menedżer okien nie czeka, aż okno przestanie się ruszać, by zacząć odrysowywać jego tło. Tło zmienia się na bieżąco nawet wtedy, gdy okno jest przesuwane nad tapetą i innymi oknami.

Prace nad programem *transset* są jeszcze mniej zaawansowane niż w przypadku programu *xcompmgr*. Na razie nie ma funkcji, która przekazywałaby menedżerowi okien wiadomość, które okno ma być przezroczyste. Proces nadawania przezroczystości trzeba więc przeprowadzać ręcznie po każdym uruchomieniu środowiska graficznego.

Bez wątpienia w przyszłości program stanie się przyjazny dla użytkownika. W międzyczasie można ułatwić sobie proces nadawania przezroczystości, dodając pozycję do menu lub tworząc ikonę na pulpicie wywołującą polecenie **transset 0.7**. Wystarczy tylko kliknąć ikonę (lub wybrać pozycję menu) i kliknąć w oknie, a stanie się ono przezroczyste.