

Screen

Potęga wirtualnych ekranów

Niektóre narzędzia od razu rodzą się wielkie, niektóre osiągają wielkość, inne zaś ciągle dążą ku wielkości. W tym miesiącu skierujemy blask światła na wielkie narzędzie o nazwie screen. **STEVEN GOODWIN**



Przeznaczeniem screen jest możliwość utworzenia kilku wirtualnych konsoli wewnątrz pojedynczej sesji terminalowej. Każda konsola ma, niezależnie od innych, możliwość uruchamiania programów w swoim własnym oknie – nawet, gdy użytkownik wylogował się z sesji. Używając screen-a można sprawować kontrolę nad kilkoma konsolami za pomocą jednego połączenia i przełączać się do nich w późniejszym czasie. Jest to niemożliwe do osiągnięcia za pomocą zwykłego nohup.

Pierwsze symptomy

Screen uruchamiany jest po prostu przez podanie jego nazwy po czym wyświetlany jest komunikat powitalny. Przyciśnięcie [Enter] usuwa napis i powoduje powrót do ekranu z znakiem zachęty powłoki. Polecenie ls wyświetla zawartość katalogu, tak jak dzieje się to w każdym dniu tygodnia. Jedyną różnicą wskazującą, że coś się stało, jest to, że ekran został wyczyszczony i kursor pojawił się w pierwszej linii ekranu.

To co się właśnie wydarzyło, to start screena i uruchomienie go w osobnym procesie. Przydzielona została domyślna powłoka, czyli zazwyczaj bash, jednakże pracująca pod kontrolą screen. Oznacza to, że oprócz przejęcia kontroli nad sesją, przydzielona została dodatkowo specjalna kombinacja klawiszy Ctrl+a; kombinacja ta używana jest do przekazywania poleceń do screen. Na początek otworzymy stronę pomocy wciskając razem Ctrl+a (nazywane odąd [C-a]). Powrót do pracy i usunięcie wyświetlania strony pomocy nastąpi po na-

ciśnięciu [Enter].

Na początek należy stworzyć nową powłokę. Uzyskuje się to poprzez wcisnięcie [C-a], a następnie [c] (bez Shift-a), co oznacza komendę *create*. Także i tym razem nic nie wydarzyło się na ekranie: ciągle widzimy znak zachęty bash. Powołany został jednak nowy znak zachęty. Aby się o tym przekonać, możemy wcisnąć kombinację klawiszy [C-a „], powinniśmy wówczas zobaczyć listę wirtualnych okien będących pod kontrolą screen'a. Konkretnie okno możemy wybrać korzystając z klawiszy kursora i wciskając [Enter].

Num	Name	Flags
0	bash \$	
1	bash \$	

To tylko jeden z wielu sposobów zmiany okna (zobacz RAMKE: Wielokątne Okno).

Chociaż można otwierać tak wiele wirtualnych konsol, na ile pozwoli tylko wielkość pamięci, to jednak tylko pierwszych dziesięć będzie posiadać eleganckie skróty klawiszowe i tylko jedna będzie widoczna w danej chwili. Niezależnie od wyglądu, wszystkie programy kontynuują działanie. Dzięki temu jedno z tych okien może być używane do edycji programu, inne do kompilacji, a jeszcze inne do uruchamiania i testowania – pozwala to uzyskać podobny efekt, jak w przypadku wirtualnych konsol przełączanych klawiszami [alt+F1] – [alt+F6].

Domyślnie wyświetlana nazwa odpowiada

działającemu programowi, zazwyczaj jest to bash. Zamiast tego możemy nadać bardziej opisową nazwę bieżącemu oknu, używając komendy 'title', która jest wywoływana skrótem [C-a A] (wielkieA).

Podwójna becinka

W razie potrzeby kopiowania i wklejania pomiędzy oknami należy przełączyć się do trybu kopiowania. W trybie tym można poruszać się przewijając widoczny tekst (jak również linie będące w buforze historii screen) w celu wybrania początku i końca tekstu przeznaczonego do skopiowania. Przejście do trybu kopiowania następuje po wcisnięciu kombinacji klawiszy [C-a], a następnie [Escape]. Poruszanie się wewnątrz tekstu odbywa się za pomocą klawiszy [H], [J], [K] i [L], przemieszczających kursor odpowiednio w lewo, dół, górę i prawo. W zależności od typu terminala możliwe jest także korzystanie z klawiszy kursora. Gdy już umieścimy kursor w punkcie

Wielokątne Okno

C-a C-a	Przełączenie pomiędzy dwoma ostatnimi oknami
C-a '	Wyświetlenie listy wyboru okien
C-a 0	Przełączenie do okna 0
C-a 1	Przełączenie do okna 1 (itd. aż do 9)
C-a n	Następne okno
C-a p	Poprzednie okno

Złe sny

Nie wszystko jest różowe w krainie screen. Po pierwsze, domyślny skrót klawiszowy do wywoływania poleceń ([C-a]) jest taki sam, jak skrót używany przez powłokę do ustawienia kursora na początku bieżącej linii. Także komunikat startowy jest trochę dokuczliwy. Obydwa problemy można jednak rozwiązać przez drobną edycję pliku \$HOME/.screenrc.

```
escape '^Ss'
startup_message off
```

startu, należy nacisnąć klawisz spacji.

Teraz przemieścić kursor do punktu końcowego i ponownie nacisnąć spację, dokonując w ten sposób wyboru zaznaczonego obszaru (na ekranie obszar ten zostanie podświetlony w trybie inwersji). Zaznaczenie punktu końcowego spowoduje automatyczne umieszczenie tekstu w buforze kopiowania. Zawartość bufora może być następnie wklejona do innego okna za pomocą kombinacji klawiszy [C-a] (prawy nawias kwadratowy). Używając edytora należy pamiętać o przełączeniu go w tryb wprowadzania przed rozpoczęciem wklejania tekstu. Screen wysyła tekst do okna w miarę wprowadzania na klawiaturze, bez względu na działanie programu.

Bufor wklejania istnieje wyłącznie w pamięci, ale może być zapisany na dysk za pomocą właściwej instrukcji dostępnej z trybu linii poleceń screen. Wykonuje się to za pomocą polecenia [C-a:]. Pojawia się wówczas znak zachęty w dolnej linii ekranu, co przypomina oczekiwania na instrukcję, znane z vi.

```
:writebuf ~/my_file_of_buffer?
_text
```

Należy być ostrożnym, ponieważ każdy plik, którego nazwa zostanie podana bez pełnej ścieżki, zostanie zapisany w katalogu, z którego screen został uruchomiony po raz pierwszy nie zaś w bieżącym katalogu. To samo dotyczy polecenia readbuf.

Dodatkowo screen umożliwia wymianę dużych bloków tekstu pomiędzy sesjami. Działa to w ten sposób, że bufor najpierw jest zapisywany do pliku na dysku, a następnie ponownie zaczytywany. Zapisywanie wykonywane jest za pomocą polecenia [C-a >], otwieranie natomiast przez [C-a <], co wymaga mniejszego wysiłku niż wprowadzenie poleceń writebuf i readbuf. Plik tymczasowy ma domyślną nazwę /tmp/screen-xchg, można ją jednak zmienić za pomocą polecenia bufferfile.

Odtwarzanie w tle

Jedną z największych zalet screen jest możliwość odłączania. Dzięki temu można odłączyć wszystkie wirtualne okna od terminala użytkownika, pozwalając zarazem programom kontynuować ich pracę w tle. Możliwe jest nawet wylogować się, a wszystkie procesy będą kontynuować swoje działanie. Pozwala to utrzymać aktywną sesję IRC, gdy w tym samym czasie trwa ściąganie plików albo oczekiwać na wyniki trylogii./configure && make && make install, robiąc coś zupełnie innego.

Procesy odesłane w tło mogą być przywołane kilkoma magicznymi klawiszami. [C-a C-d] (Control+d) odłącza screen od terminala i zwraca „prawdziwy” znak zachęty basha. Jeśli teraz napiszesz screen -list, będziemy mogli zobaczyć, że jedna odłączona sesja screen'a działa w tle. Teraz można zrobić wszystko (włącznie z wylogowaniem), także podłączyć ją ponownie z dowolnego miejsca, nawet z zupełnie innej maszyny

```
$ screen -r
```

Jeśli bieżący użytkownik ma kilka działających sesji screen, wyświetlona zostanie lista zawierająca id procesu, terminal i nazwę maszyny dla każdej z nich. W tym przypadku należy ponownie podłączyć się do określonego terminala, podając id jego procesu:

```
$ screen -list
There are several suitable >
screens on:
17169.pts-1.tori (Detached)
17200.pts-1.tori (Detached)
Type 'screen [-d] -r [pid.]tty>
.host' to resume one of them.
$ screen -r 17169
```

W trosce o użytkowników żyjących w ciągłym pośpiechu, autorzy screen przewidzieli specjalną opcję odłączania [C-a D D], pozwalającą za jednym zamachem odłączyć screen i wylogować bieżącego użytkownika.

Istnieje także możliwość podłączenia do procesu, który nawet nie był odłączony. Wymaga to podania specjalnej opcji -x, która pozwala w tym samym czasie kontrolować dwóm wirtualnym oknom ten sam program z dwóch różnych miejsc, np. klienta poczty elektronicznej. Uaktualnienia ekranu pojawiają się od razu na wszystkich podłączonych maszynach (choćby możliwe są pewne problemy z wyświetlaniem, zwłaszcza gdy poszczególne terminale mają różne wielkości). Właściwość ta może być wykorzystywana do

pracy grupowej albo jako narzędzie monitorujące, tym bardziej, że nie ma potrzeby odłączania oryginalnego okna. Te same zasady odłączania screen mają zastosowanie w trybie 'wielo-wyświetlaczowym', gdy uruchomiona jest więcej niż jedna sesja screen.

```
$ screen -x
```

Porzuć go

Możliwości odłączania i podłączenia screen są mile widziane jeszcze w jednym przypadku. Ma to miejsce np. wtedy gdy wcale nie zamierzaliśmy się wylogować, jednak coś (lub ktoś) sprawiło, że rozłączyliśmy się z maszyną, na której pracowaliśmy. Jeśli wszystkie zadania ukryte były w sesjach screen, powinny dalej działać bezpiecznie. Ponowne podłączenie do tych sesji możliwe jest za pomocą opisanych powyżej metod.

Za pomocą screen możemy ponadto uruchomić klienta pocztowego z pominięciem basha. Wystarczy napisać:

```
screen mutt
```

Dowolny program uruchomiony w ten sposób może także pobrać zestaw argumentów. Dlatego dla uniknięcia zamieszania, opcje przeznaczone dla screen muszą poprzedzać nazwę programu. W dodatku pozwala to na automatyczne ukrycie ekranu z napisanymi licencyjnymi. Screen zakończy także działanie wraz z zakończeniem pracy obsługiwanego przez niego programu, wyświetlając komunikat:

```
[screen is terminating]
```

Lekki d eszcz

W rzeczywistości Screen posiada dużo więcej opcji, my mieliśmy okazję zapoznać się zaledwie z kilkoma z nich. Podręcznik pomocy man zawiera pełną listę poleceń, przykładów i skrótów klawiszowych, na których przedstawienie zabrakło tutaj po prostu miejsca. Do ciekawszych właściwości należą m.in. obsługa znaków Braille'a i Kanji. Niektóre z nich, jak np. możliwość zablokowania konsoli ekranowej (za pomocą kombinacji [C-a x]), są bardzo proste, inne, dotyczące konfiguracji ustawień terminala, są bardziej skomplikowane. Ale teraz czas na dalsze odkrywanie własności screen... ■

INFO

[1] Screen: <http://ftp.uni-erlangen.de/pub/utilities/screen/>