

# Linux – wielozadaniowość, wielodostępność, zarządzanie pamięcią

## Wielozadaniowość

Linux jest systemem **wielozadaniowym** – co umożliwia mu równoczesne wykorzystywanie więcej niż jednego **procesu** (**proces** to egzemplarz wykonywanego programu, posiadający własną przestrzeń adresową). Za realizację wielozadaniowości odpowiedzialne jest jądro systemu operacyjnego (*kernel*).

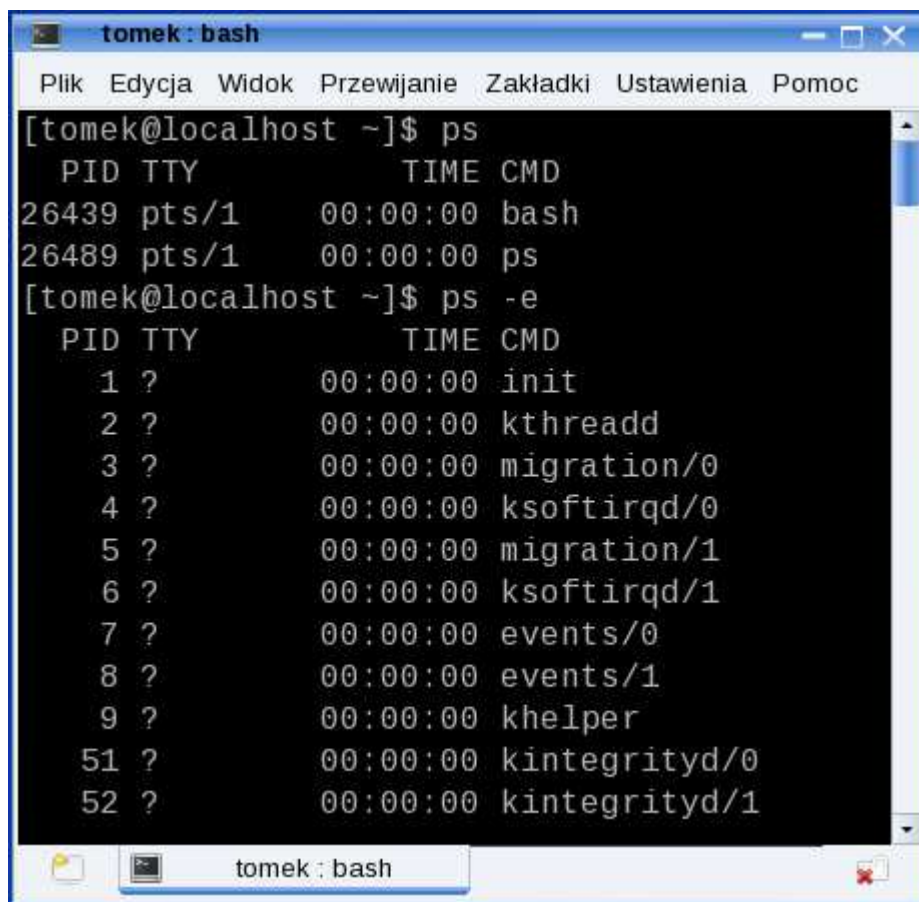
**Wielozadaniowość** zapewniona jest przez program nazywany **planistą**, który realizuje algorytm szeregowania zadań w kolejce do przyznania czasu procesora. Linux może wykorzystywać wiele procesorów. Gdy system ma mniej dostępnych procesorów niż zadań to czas działania procesora jest dzielony pomiędzy wszystkie zadania.

Linux operuje także tzw. **wyłączenie**, które polega na przerwaniu wykonywania procesu, odebraniu mu procesora i przekazaniu sterowania do **planisty**.

Do wyświetlania listy uruchomionych procesów służy polecenie **ps**. Polecenia tego może używać każdy użytkownik, jednak wyświetlane będą tylko te procesy, których jest właścicielem. Najważniejszą spośród wyświetlanych informacji to:

- **PID** - identyfikator procesu
- **TTY** – identyfikator konsoli, z której proces został uruchomiony
- **TIME** – czas procesora wykorzystany do tej pory
- **CMD** – polecenie, którym uruchomiono proces

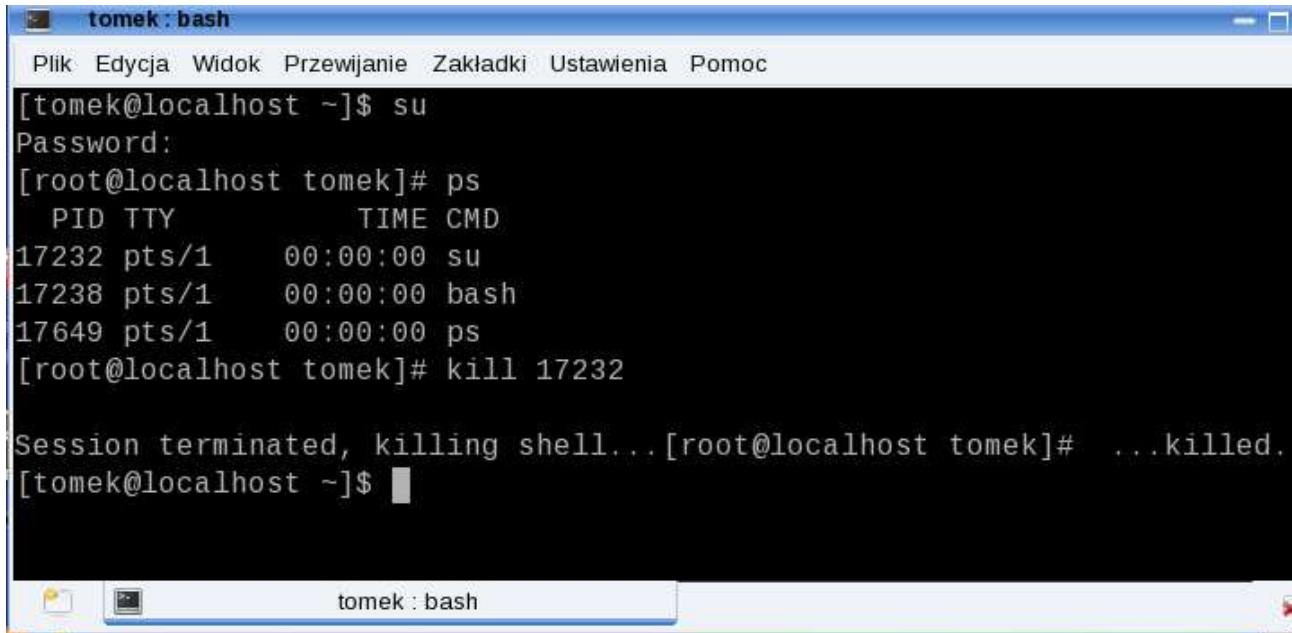
Listę wszystkich procesów można wyświetlać za pomocą komendy: **ps -e**.



```
tomek : bash
Plik  Edycja  Widok  Przewijanie  Zakładki  Ustawienia  Pomoc
[tomek@localhost ~]$ ps
  PID TTY          TIME CMD
26439 pts/1    00:00:00 bash
26489 pts/1    00:00:00 ps
[tomek@localhost ~]$ ps -e
  PID TTY          TIME CMD
   1 ?           00:00:00 init
   2 ?           00:00:00 kthreadd
   3 ?           00:00:00 migration/0
   4 ?           00:00:00 ksoftirqd/0
   5 ?           00:00:00 migration/1
   6 ?           00:00:00 ksoftirqd/1
   7 ?           00:00:00 events/0
   8 ?           00:00:00 events/1
   9 ?           00:00:00 khelper
  51 ?           00:00:00 kintegrityd/0
  52 ?           00:00:00 kintegrityd/1
```

## Zarządzanie procesami w systemie Linux

Linux jest systemem bardzo stabilnym, lecz również i w tym systemie zachodzi możliwość zawieszania procesu. Czasami zachodzi konieczność usunięcia procesu działającego. Zwykły użytkownik może usuwać tylko te procesy, których jest właścicielem. Administrator może usuwać wszystkie procesy. Do usuwania procesów służy polecenie **kill**. Polecenie to musi otrzymać jako argument identyfikator **PID** procesu. Usuwa ono proces w sposób bezpieczny (zamyka otwarte pliki, zwalnia obszary pamięci). Jeżeli procesu w ten sposób usunąć nie można to możemy wymusić jego zakończenie dodając opcję **-9** do polecenia. Np. `kill -9 2463`.



```
tomek : bash
Plik  Edycja  Widok  Przewijanie  Zakładki  Ustawienia  Pomoc
[tomek@localhost ~]$ su
Password:
[root@localhost tomek]# ps
  PID TTY          TIME CMD
17232 pts/1        00:00:00 su
17238 pts/1        00:00:00 bash
17649 pts/1        00:00:00 ps
[root@localhost tomek]# kill 17232

Session terminated, killing shell...[root@localhost tomek]# ...killed.
[tomek@localhost ~]$
```

Powyżej uruchomiłem proces o nazwie **su**, który jest logowaniem się jako administrator, następnie poleceniem **ps** wyświetliłem listę moich procesów, następnie poleceniem **kill** unicestwiłem proces **su** i zostałem automatycznie wylogowany jako administrator i przełączony na poprzedniego użytkownika.

### UWAGA

Usuwanie procesów jest zabiegiem niebezpiecznym. Usunięcie ważnego procesu może spowodować unieruchomienie całego systemu.

## Wielodostępność

Linux jest systemem wielodostępnym. Oznacza to, że wielu użytkowników może jednocześnie z niego korzystać.

Każdy użytkownik ma przydzielone zasoby:

- pamięć
- czas procesora
- miejsce na dysku

Do jego zasobów nie nikt dostępu chyba, że sam je udostępni. Wyjątkiem jest administrator. Użytkownik jest niezależny i odizolowany od innych. Aby zalogować się do systemu możemy skorzystać z klawiatury podłączonej do komputera, lub sieci komputerowej. Po zalogowaniu się uzyskujemy dostęp do konsoli. W linuxie zdefiniowanych jest wiele konsoli tekstowych, na których mogą logować się różni użytkownicy.

### KONSOLA – LOGOWANIE

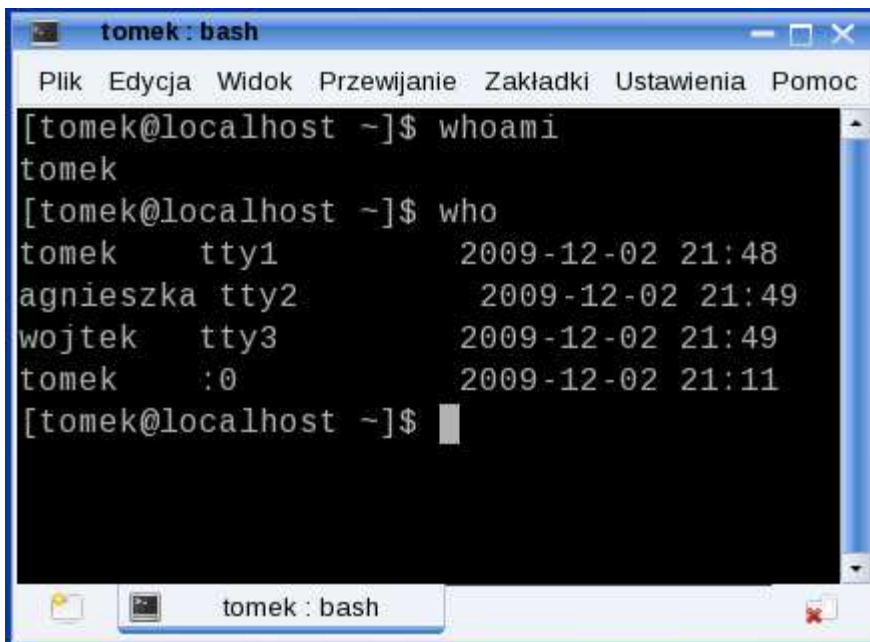
[Ctrl] + [Alt] + [Fn] np. [Ctrl] + [Alt] + [Fn] – aby zalogować się do pierwszej konsoli.

[Ctrl] + [Alt] + [F7] – powrót do środowiska graficznego.

## KONTO

**whoami** - sprawdza z jakiego konta aktualnie korzystamy.

**who** – kto aktualnie jest zalogowany w systemie.



```
tomek : bash
Plik Edycja Widok Przewijanie Zakładki Ustawienia Pomoc
[tomek@localhost ~]$ whoami
tomek
[tomek@localhost ~]$ who
tomek      tty1          2009-12-02 21:48
agnieszka tty2          2009-12-02 21:49
wojtek    tty3          2009-12-02 21:49
tomek     :0            2009-12-02 21:11
[tomek@localhost ~]$
```

## WIADOMOŚCI PRZESYŁANE POMIĘDZY UŻYTKOWNIKAMI

**write** – polecenie służy do wysyłania wiadomości do innego użytkownika.

**wall** – polecenie służy do przesyłania wiadomości do wszystkich zalogowanych użytkowników.

### KONSOLA TOMKA PODCZAS WYSYŁANIA

```
[tomek@localhost ~]$ write wojtek
Witam. Jestem Tomek. Pozdrawiam.
```

### KONSOLA WOJTKA PODCZAS ODCZYTU

```
[wojtek@localhost ~]$
message from tomek@localhost on pts/3 at 19:21 ...
Witam. Jestem Tomek. Pozdrawiam.
EOF
```

## Zarządzanie pamięcią

Czym jest pamięć wirtualna? Linux obsługuje *pamięć wirtualną* - wykorzystuje część dysku jako rozszerzenie fizycznej pamięci. Jądro zapisuje zawartość nieużywanych bloków pamięci fizycznej na dysku, umożliwiając tym samym wykorzystanie ich do innych celów. Jeżeli oryginalna zawartość jest potrzebna następuje proces odwrotny. Wszystko to odbywa się niewidocznie dla użytkownika; działające programy również nie dostrzegają różnicy. Oczywiście operacje dyskowe są znacznie wolniejsze (tysiące razy) niż analogiczne działania na fizycznej pamięci, programy zwalniają. Część dysku twardego wykorzystywana jako pamięć wirtualna nosi nazwę *obszaru wymiany* - SWAP.

Linux potrafi wykorzystywać zwykły plik na systemie plików, lub oddzielną partycję jako obszar wymiany. Partycja wymiany jest szybsza, jednak trudniej zmienić jej rozmiar, niż w wypadku pliku. Możesz w celach testowych stworzyć plik wymiany, a gdy już będziesz pewien co do potrzebnego rozmiaru stworzyć osobną partycję.