

# Laboratorium 1-2

## Podstawy systemu operacyjnego UNIX

### Wprowadzenie

System operacyjny jest w ogólności zestawem programów nadzorujących prace komputera. W skład tego zestawu mogą wchodzić również programy o charakterze czysto użytkowym (np. edytory tekstu, kompilatory, itd.). UNIX to system operacyjny, który jest:

- ❑ **Wielodostępny** - tzn. może nadzorować równoczesną pracę wielu użytkowników komputera. Użytkownik jest rozpoznawany przez identyfikator swojego konta w systemie (*userid*). Konto użytkownika jest chronione prywatnym hasłem (*password*).
- ❑ **Wielozadaniowy** - tzn. każdy użytkownik może zlecać wykonanie wielu zadań równocześnie (dokładniej - z podziałem czasu).

Bezpieczeństwo systemu wymaga aby mogli w nim pracować tylko zarejestrowani użytkownicy. Rejestracja użytkowników i ogólnym nadzorem zajmuje się użytkownik **root**, zwany również *administratorem systemu*. Jest on użytkownikiem mającym najszersze uprawnienia w systemie.

W strukturze logicznej systemu UNIX można wyodrębnić następujące pojęcia:

### **Jądro**

Zbiór najważniejszych podprogramów, zarządzających zasobami systemu. Od momentu uruchomienia komputera jądro sprawuje nad nim pełną kontrolę. Użytkownik komunikuje się z jądrem przez *powłokę*.

### **Powłoka**

Interpreter poleceń użytkownika. Powłoka pośredniczy pomiędzy użytkownikiem a jądrem. Kiedy użytkownik rozpoczyna pracę (*loguje* się do systemu podając poprawny identyfikator konta i hasło) system operacyjny uruchamia program powłoki, który pobiera polecenia do wykonania z klawiatury terminala użytkownika.

W systemie UNIX istnieje wiele różnych powłok (np. sh, ksh, zsh, csh itd.). Administrator przydziela każdemu użytkownikowi jego *powłokę standardową* (uruchamiana w chwili logowania się do systemu); użytkownik może też uruchomić dowolną powłokę zainstalowaną w systemie.

### **Komendy**

Są to programy, dostarczane wraz z systemem operacyjnym lub zainstalowane przez administratora, niekoniecznie należące do samego systemu. Komendy wywołuje się podając ich nazwę i ew. opcje i argumenty.

Przykłady:

```
ls -l
vi plik
cat p1 p2 >p3
sort p
mail
sh
```

Komendy na ogół współpracują ze sobą (poprzez mechanizm *potoków*) i z jądrem systemu.

### **Pliki**

Plik jest porcją informacji (np. tekst napisany pod edytorem) zapisana na dysku. Użytkownik gromadząc pliki tworzy swoje własne zasoby i może nimi zarządzać w ramach systemu plików. *Nazwa pliku* może składać się z dowolnych znaków mających reprezentację graficzną (litery i cyfry, znaki interpunkcji itp.) oraz niektórych znaków specjalnych. Długość nazwy pliku jest zwykle ograniczona do 255 znaków (w starszych implementacjach UNIX'a - do 14 znaków). Wszystkie dane o pliku, poza nazwą, są przechowywane w strukturze systemowej zwanej *i-węzłem* (ang. *i-node*).

### **Katalogi**

Katalog jest specjalnym rodzajem pliku, w którym jest zapisana informacja o innych plikach (i/lub katalogach). Pozwala to utworzyć hierarchiczną, przypominającą drzewo, strukturę katalogów z umieszczonymi w katalogach plikami. Główny katalog drzewa oznaczany jest znakiem / Użytkownik wraz z kontem otrzymuje swój *prywatny katalog* (ang. *HOME directory*), gdzie ma wszystkie prawa (może tworzyć, modyfikować i usuwać pliki/katalogi).

W chwili zalogowania się do systemu prywatny katalog użytkownika staje się *katalogiem roboczym* (nazywanym także katalogiem *aktualnym* lub *bieżącym* - ang. *working or current directory*). Katalog roboczy można zmienić komendą cd.

### **Ścieżka dostępu do pliku**

Opis lokalizacji pliku w drzewie katalogów: lista nazw katalogów oddzielonych znakiem / zakończona nazwą pliku. Istnieją dwa rodzaje ścieżek dostępu do pliku:

- ❑ *ścieżka bezwzględna*, zaczynająca się od głównego katalogu, np. /home/student/ula/doc/sales.95.raport
- ❑ *ścieżka względna*, podająca położenie pliku względem katalogu roboczego. Np., jeśli katalogiem roboczym jest /home/student/ala, to ścieżka względna do podanego wyżej pliku będzie:  
../ula/doc/sales.95.raport

### Ścieżka przeszukiwań

Ciąg nazw katalogów zapisanych w *zmiennej środowiskowej* PATH. Jeżeli użytkownik napisze z klawiatury nazwę komendy, to system operacyjny będzie poszukiwał programu o podanej nazwie w kolejnych katalogach tworzących ścieżkę przeszukiwań. W odróżnieniu od niektórych innych systemów operacyjnych, katalog roboczy w ogóle nie jest przeszukiwany (chyba, że występuje w sposób jawny w ścieżce przeszukiwań).

### Prawa dostępu do pliku/katalogu

Prawo wykonania przez użytkownika lub grupę użytkowników określonych operacji na pliku lub katalogu. Prawa dostępu są zwykle zapisywane w postaci dziewięcioznakowego kodu:

rwxrwxrwx

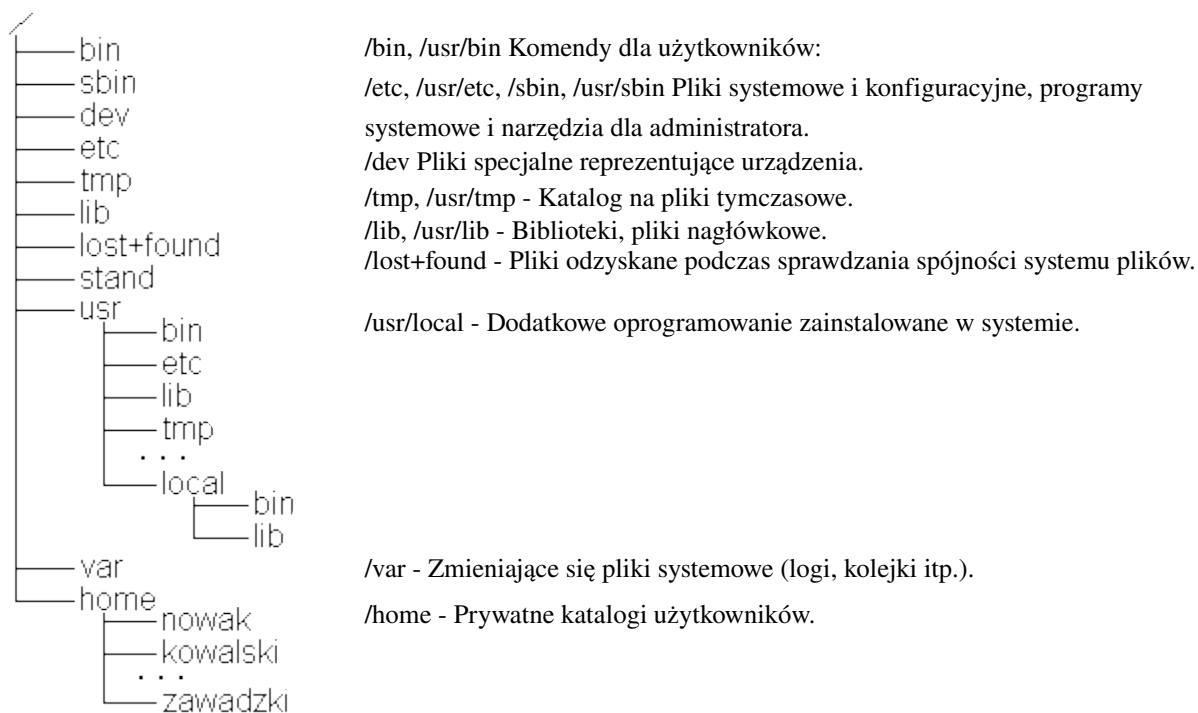
Znaczenie poszczególnych kodów jest następujące:

Kod	Plik	Katalog
r	prawo odczytu pliku	prawo odczytu zawartości katalogu
w	prawo zapisu do pliku	prawo zmiany zawartości katalogu (utworzenia lub usunięcia pliku)
x	prawo wykonania pliku	prawo przejścia przez katalog

Pierwsza grupa znaków rwx odnosi się do praw właściciela, druga grupy, a trzecia - pozostałych użytkowników.

### Katalogi systemowe

Poniżej przedstawiono typowy układ katalogów systemu UNIX.



## Wybrane polecenia systemu UNIX

### Dokumentacja elektroniczna

Użytkownik systemu UNIX ma do dyspozycji systemy dokumentacji elektronicznej, które dostarczają wyczerpujące opisy składni i zastosowań wszystkich poleceń dostępnych w tym systemie.

Podstawowym podręcznikiem elektronicznym są *man pages*. Podręcznik ten wywołuje się poleceniem:

man [-s nr\_sekcji] temat

gdzie:

*temat* jest nazwa komendy, funkcji systemowej, struktury danych, o której chcemy się uzyskać informacje,

*nr\_sekcji* jest numerem rozdziału *UNIX Reference Manual*, i podajemy go w przypadku gdy *temat* może mieć wiele znaczeń.

Przykłady:

man man  
man ls  
man -s l zsh

Dla użytkowników pracujących w środowisku graficznym *X-Windows* dostępne są dwa podręczniki: *xman* oraz *answerbook*. Wystarczy je wskazać myszką i postępować zgodnie z wyświetlanymi wskazówkami.

### Operacje na plikach i katalogach

ls  
wyświetla listę plików z katalogu roboczego

ls *kat*  
wyświetla zawartość katalogu *kat*

ls -l  
wyświetla listę plików w postaci długiej

ls -F  
zaznacza rodzaj pliku (dodając znak / po nazwie katalogu, \* po nazwie pliku wykonywalnego, @ po nazwie łącznika itp.

ls -a  
wyświetla wszystkie pliki z katalogu bieżącego, łącznie z plikami ukrytymi

ls -d  
wyświetla tylko nazwy podkatalogów w katalogu bieżącym

cd *kat*  
zmienia katalog roboczy na *kat*

pwd  
wyświetla nazwę katalogu roboczego

mkdir *kat*  
zakłada w katalogu roboczym podkatalog *kat*

rmdir *kat*  
usuwa katalog *kat* (jeżeli jest pusty)

cp *pl1 pl2*  
kopiuje zawartość pliku *pl1* do nowego pliku *pl2*

cp *pl1 pl2 pl3 kat*  
kopiuje pliki *pl1, pl2, pl3* do katalogu *kat*

rm *pl1 pl2*  
kasuje pliki *pl1, pl2*

mv *pl1 pl2*  
zmienia nazwę pliku *pl1* na *pl2*

mv *pl1 pl2 pl3 kat*  
przenosi pliki *pl1, pl2, pl3* do katalogu *kat*

ln *pl1 nazwa2*  
tworzy łącznik (tzw. "twardy") *nazwa2* do pliku *pl1*

ln *pl1 pl2 pl3 kat*  
tworzy łącznik do plików *pl1, pl2, pl3* w katalogu *kat*

ln -s *ścieżka skr*  
tworzy łącznik symboliczny ("miękki") o nazwie *skr*, zastępujący ścieżkę dostępu *ścieżką*

find *kat* -name *pl* -print  
wyszukanie plików o nazwie *pl* w drzewie katalogów zaczynającym się od katalogu *kat*

### Prawa dostępu

chown *piotr pl1 pl2 pl3*  
przekazuje prawa własności do plików *pl1, pl2, pl3* użytkownikowi *piotr*

chgrp *stud pl1 pl2 pl3*  
zmiana identyfikatora grupy plików *pl1, pl2, pl3* na *stud*

umask  
wyświetla aktualną maskę praw dostępu dla nowo tworzonych plików

umask 022  
zmienia maskę praw dostępu dla nowo tworzonych plików wg znaczenia 3-cyfrowej liczby ósemkowej będącej argumentem komendy - bity reprezentacji dwójkowej tej liczby określają maskowane prawa dostępu. W powyższym przykładzie użytkownicy inni niż właściciel nie będą mieli prawa zapisu w do nowo tworzonych plików i katalogów:  
000 010 010  
rwx rwx rwx

chmod 744 pl1

chmod u=rwx og=r pl1

ustawienie praw dostępu do pliku lub katalogu należącego do użytkownika (prawa rwx dla właściciela, prawo r dla pozostałych użytkowników).

### Wybrane komendy działające na plikach tekstowych

more

pg

less

programy stronicujące - wyświetlają zawartość pliku strona po stronie

cat

łączenie plików; może być wykorzystany (jeśli jest wołany z jednym argumentem) do wyświetlenia zawartości pliku

head

wyświetlenie kilku pierwszych wierszy pliku tekstowego

tail

wyświetlenie kilku ostatnich wierszy pliku tekstowego

sort

sortowanie linii pliku

grep

wyszukiwanie linii w plikach tekstowych według podanego wzorca

wc

wyznaczenie liczby linii/słów/znaków w pliku tekstowym

### Wybrane komendy związane z użytkownikiem

id

wyświetlenie UID/GID użytkownika

logname

whoami

wyświetlenie nazwy użytkownika

su

chwilowa zmiana identyfikatora użytkownika

who

wyświetlenie listy pracujących użytkowników

finger

wyświetlenie informacji o innym użytkowniku

### Wymiana informacji między użytkownikami

write

wypisanie komunikatu na ekranie innego użytkownika

mesg

zablokowanie/odblokowanie wypisywania komunikatów od innych użytkowników

mail

elm

poczta elektroniczna

### Niektóre inne komendy

cal

kalendarz

bc

kalkulator