

## Cechy systemu Linux

- pełna wielozadaniowość – wielu użytkowników w tym samym czasie może wykonywać kilka zadań na tym samym komputerze;
- pamięć wirtualna – Linux może używać części dysku twardego jako pamięci wirtualnej, co zwiększa wydajność systemu poprzez umieszczanie aktywnych procesów w pamięci RAM, a tych rzadziej używanych lub nieaktywnych na dysku;
- wbudowana obsługa sieci – poprzez modem lub kartę sieciową Linux może porozumiewać się z innymi komputerami wykorzystując protokoły TCP/IP, NFS, NIS i inne;
- współużytkowanie bibliotek – programy w systemie Linux nie przechowują własnych kopii standardowych bibliotek tylko wspólnie współużytkują zestaw procedur, do których mogą odwoływać się w trakcie działania. Dzięki temu otrzymujemy więcej przestrzeni dyskowej;
- kompatybilność z norma POSIX – dzięki czemu Linux obsługuje wiele standardów ustalonych dla systemów Unix;
- X Window – jest to środowisko graficzne;
- publiczny kod źródłowy – dzięki czemu możemy aktywnie uczestniczyć w rozwoju systemu;
- darmowe oprogramowanie – dostajemy wraz z systemem na płytach CD, poza tym istnieje wiele stron internetowych, na których możemy znaleźć ciekawe oprogramowanie.

## Logowanie się do systemu

Systemy operacyjne z rodziny Linux mogą równocześnie obsługiwać wielu użytkowników. Muszą oni posiadać konto na danym komputerze pracującym pod kontrolą tego systemu. Konto na danym komputerze może założyć administrator danego systemu.

Ustala on ponadto przynależności do grup, dostęp do odpowiednich zasobów w systemie, itp. Po uruchomieniu systemu na ekranie pojawia się monit z prośbą o wprowadzenie nazwy użytkownika i odpowiedniego hasła. Po zakończeniu sesji użytkownik powinien się z systemu wylogować. Służy do tego polecenie **exit**.

## Powłoka systemowa

Powłoka systemowa (*ang. shell*) jest to interfejs użytkownika, który pozwala na wprowadzanie komend oraz ich interpretację. Powłoka umożliwia również komunikację z niższymi warstwami systemu Linux. Powłoka pełni w systemie Linux funkcję zbliżoną do interpretera poleceń `command.com` znanego z systemu DOS oraz Windows 9x.

Formatem polecenia jest nazwa polecenia, za którą następują opcje, a dopiero za nimi argumenty:

### *połączenie opcje argumenty*

- Większość poleceń systemu Linux znajduje się w katalogu */bin* oraz */usr/bin*.
- Przykładowo polecenie *ls*, służące do wyświetlania zawartości katalogu, ma 74 dostępne opcje.
- Większość poleceń wyświetla bardzo skąpe informacje wynikowe. Często więc, brak odpowiedzi po wykonaniu polecenia należy traktować jako jego poprawne wykonanie. Przykładowo polecenie *ls* zastosowane do pustego katalogu nie wyświetli żadnego komunikatu.

Opcje poleceń poprzedza się myślnikiem i muszą one być oddzielone spacją od polecenia oraz innych opcji. Powłoka systemowa pozwala jednak na łączenie kilku opcji. Stosuje się wówczas zapis, w który wprowadza się jeden myślnik, a następnie podaje się opcje.

## **System pomocy**

System Linux posiada rozbudowany i bardzo przydatny system pomocy. Pierwszym źródłem pomocy są tzw. strony *man* (*ang. manual*). Zawierają one definicje i objaśnienia poleceń systemu wraz z opisem opcjonalnych parametrów dla specjalnych funkcji poleceń. Aby wywołać stronę *man* dla wybranego polecenia należy wydać komendę:

*man polecenie*

np.:

*man ls*

Spowoduje to wyświetlenie opisu danego polecenia. Jeśli chcemy wyszukać potrzebna informacje należy nacisnąć klawisz `\`, wpisać szukaną frazę i nacisnąć ENTER. Jeśli chcielibyśmy ponowić wyszukiwanie danej frazy wystarczy nacisnąć klawisz `\` i ENTER. Aby zakończyć przeglądanie *manuala* należy nacisnąć klawisz *q*.

## **Wiele sesji na komputerze lokalnym**

System Linux posiada siedem wirtualnych konsol, do których mamy dostęp za pomocą kombinacji klawiszy: ALT+F1 do F7. Podczas uruchamiania systemu X Window ręcznie lub automatycznie, zablokowana zostaje konsola, z której zostały uruchomione X-y, a nasz menadżer wyświetlania dostępny jest na konsoli 7.

Przejsie z poziomu X-ów na pozostałe konsole odbywa się za pomocą następującej kombinacji klawiszy: CTRL+ALT+F1 do F6. Powrót do X Window odbywa się za pomocą kombinacji CTRL+ALT+F7. Na każdej z wirtualnych konsol możemy zalogować się, jako inny użytkownik. Takie rozwiązanie umożliwia nam uruchomienie na każdej z konsol jakiegoś zadania.

## **Operacje na katalogach i plikach**

Kropka nie posiada specjalnego znaczenia, ponieważ Linux traktuje kropkę jak każdy inny znak. Pliki, których nazwy rozpoczynają się kropką są "ukryte". Katalogi systemowe zawierają pliki i programy używane do uruchomienia i utrzymywania systemu. Znaczenie poszczególnych katalogów systemowych jest następujące:

*/bin, /usr/bin* – zawierają większość komend systemowych i programy usługowe w wersji binarnej,  
*/dev* – mieści się w nim pliki specjalne, reprezentujące urządzenia rzeczywiste (dyski twarde, elastyczne, drukarki) i pseudourządzenia (konsola systemowa, obszar swap, wirtualny terminal),  
*/etc* – przechowywana jest w nim większość plików i programów umożliwiających konfigurację systemu,  
*/tmp* – używany przez komendy, jak i przez użytkowników do przechowywania plików tymczasowych,  
*/home* – katalog przeznaczony na katalogi domowe użytkowników systemu,  
*/proc* – wirtualny system plików, zawierający informacje o systemie i uruchomionych procesach.

## Poruszanie się po katalogach

- Przy rejestrowaniu nazwy użytkownika w systemie zostaje z nią związany katalog osobisty danego użytkownika, oznaczony symbolem `~`. Katalog ten staje się katalogiem bieżącym w chwili rozpoczynania przez użytkownika sesji przy terminalu.
- W każdym katalogu (oprócz katalogu `root` → `/`) występuje element domyślny, stanowiący nazwę katalogu **nadrzędnego**, oznaczony dwiema kropkami (`..`) oraz inny element, stanowiący nazwę katalogu, w którym on sam się znajduje, oznaczony (`.`).
- Położenie pliku jest określane poprzez ciąg nazw prowadzących do niego katalogów. Ciąg ten nazywany jest ścieżką dostępu. Ścieżka dostępu do obiektu określa umiejscowienie obiektu w systemie plików. Ścieżka dostępu może być określana na dwa sposoby: jako ścieżka **bezwzględna** albo **względna**.
- Bezwzględna ścieżka dostępu rozpoczyna się od folderu głównego. Folder główny jest pojedynczym folderem, od którego rozpoczyna się cała hierarchia. Pierwszym znakiem ścieżki bezwzględnej jest ukośnik (`/`).

## Znaki specjalne

Nazwy plików są najczęściej argumentami stosowanymi w poleceniach. Często można znać jedynie część nazwy pliku lub chcieć odwołać się do kilku nazw plików. Powłoka oferuje zestaw znaków specjalnych, które poszukują, dopasowują, generują listę plików. Tymi znakami specjalnymi są gwiazdka, znak zapytania i nawiasy kwadratowe.

- Gwiazdka `*` oznacza dowolny ciąg znaków, można jej używać do oznaczania nazw plików zaczynających się lub kończących dowolnym zestawem znaków.
- Znak zapytania `?` oznacza dokładnie jeden dowolny znak. Można używać więcej niż jednego znaku zapytania w każdym miejscu wzorca.
- Dowolne znaki umieszczone w nawiasach kwadratowych `[]` oznaczają dokładnie jeden spośród wymienionych znaków.
- Znaki specjalne mogą być maskowane za pomocą znaku `,`.

Przykłady operacje na plikach z zastosowanie znaków specjalnych:

```
rm *
```

```
cp /bin/a*b $HOME/
```

```
cp ??? $HOME/
```

```
rm [Aa-f0-9]*
```

```
cd ../..
```

## Polecenia dotyczące katalogów

### Listowanie zawartosci katalogów

*ls*

*ls -la*

#### **Przykładowe opcje**

-a – wypisuje wszystkie pliki, razem z plikami ukrytymi,

-l – podaje wszystkie dane (pełna informacje) o plikach i katalogach,

-x – wyświetla pliki posortowane według rozszerzeń

### Zmiana katalogu

*cd nazwa\_katalogu* – przejście do katalogu o podanej nazwie

*cd \$HOME, cd ~, cd* – powrót do katalogu domowego

*cd ..* – przejście do katalogu nadrzędnego

### Tworzenie katalogów

*mkdir /usr/students*

#### **Przykładowe opcje**

-p – pozwala utworzyć zagnieżdżoną strukturę katalogów

-m – pozwala na utworzenie katalogu z określonymi prawami dostępu.

### Usuwanie katalogów

*rmdir /home/Kowalski*

### Przenoszenie katalogów, zmiana nazwy katalogu

*mv old\_name new\_name*

*mv kat1 kat2* – przeniesienie katalogu kat1 do katalogu kat2

### Gdzie jestem?

*pwd* (ang. *print working directory*)

## Polecenia dotyczące plików

### Kopiowanie plików

*cp plik\_zródłowy plik\_docelowy*

#### **Przykładowe opcje**

-b – utworzenie kopii plików przed zniszczeniem ich zawartości,

-v – wyświetlenie nazwy każdego pliku w czasie kopiowania,

-r – kopiowanie całych katalogów wraz z podkatalogami,

-p – kopiowanie plików do katalogu docelowego z zachowaniem hierarchii podkatalogów

### Przenoszenie plików, zmiana nazwy

*mv plik\_zródłowy plik\_docelowy*

*mv stara\_nazwa nowa\_nazwa*

#### **Przykładowe opcje**

-b – brak pytania o potwierdzenie przed zniszczeniem zawartości plików,

-f – przeniesienie plików tylko wtedy, kiedy są nowsze niż pliki docelowe o tej samej nazwie

### Usuwanie plików

*rm nazwa\_pliku*

*rm -r nazwa\_katalogu*

### ***Przykładowe opcje***

- f – usuwa pliki bez żadnego uprzedzenia,
- r – pozwala na usuwanie katalogu ze wszystkimi podkatalogami i plikami,
- i – żąda potwierdzenia dla każdego usuwanego pliku,
- v – wyświetla nazwy każdego kasowanego pliku.

### **Tworzenie powiązań symbolicznych**

*ln -s nazwa\_linku nazwa\_katalogu*

### **Tworzenie pustego pliku**

*touch nazwa\_pliku*

### **Tworzenie i edycja pliku**

*cat > nazwa\_pliku (zakonczenie edycji CTRL + D)*

### **Tworzenie i przepisanie pliku**

*less nazwa\_pliku > nowy\_plik*

### **Doklejanie do plików**

*less nazwa\_pliku >> istniejacy plik*

### **Wyszukiwanie plików**

*find gdzie filtry obiekt\_poszukiwan*

*find / -name cp*

## ***Prawa dostępu do plików i katalogów***

Prawa dostępu podzielone są na trzy sekcje:

- właściciel pliku lub katalogu
- grupa związana z plikiem lub katalogiem
- wszyscy inni użytkownicy systemu

### **Prawa dostępu do pliku**

Każdy plik ma ściśle określone prawa dostępu stwierdzające, czy określony użytkownik jest uprawniony do odczytania lub zapisania pliku bądź do jego wykonania. Każdy użytkownik może mieć dowolną kombinację tych praw. W przypadku pliku prawa są interpretowane w następujący sposób:

- **r** – prawo czytania umożliwia oglądanie zawartości pliku, oznacza jednocześnie prawo do kopiowania,
- **w** – prawo pisania oznacza zezwolenie na modyfikację zawartości pliku,
- **x** – prawo do uruchomienia pliku wykonywalnego.

### **Prawa dostępu do katalogu**

Te same kategorie praw - czytania, pisania i wykonywania odnoszą się do katalogów:

- **r** – prawo czytania umożliwia przeszukiwanie zawartości katalogu, jest interpretowane jako prawo wypisywania zawartości (komenda `ls`),
- **w** – prawo pisania daje możliwość modyfikowania zawartości katalogów umożliwia dodawanie nowych oraz usuwanie dotychczasowych plików z katalogu,
- **x** – prawo wykonywania w stosunku do katalogu pozwala na dostęp do plików zapisanych w nim oraz na wejście do danego katalogu uczynienie go katalogiem bieżącym (polecenie: `cd katalog`).

Prawa dostępu można odczytać wydając polecenie:

```
ls -l
```

wynik polecenia

```
krwxrwxrwx
```

*k* - jest to identyfikator typu, gdzie:

- - zwykły plik
- **b** specjalny plik blokowy
- **c** specjalny plik znakowy
- **d** katalog
- *l* link symboliczny
- **p** potok
- **s** gniazdo

#### ***prawa dostępu:***

Pierwsza trójka `rwX` – oznacza uprawnienia dla właściciela (`u` – user)

Druga trójka `rwX` – oznacza uprawnienia dla grupy (`g` – group)

Trzecia trójka `rwX` – oznacza uprawnienia dla pozostałych użytkowników (`o` – others)

**slajd 45 z prezentacji**

## Przypisywanie uprawnień

Do zmiany uprawnień użytkowników w stosunku do katalogu lub pliku służy polecenie `chmod`. Wymaga ono określenia, czyje uprawnienia należy zmienić, na jakie oraz jakiego pliku lub katalogu ta zmiana będzie dotyczyć. Prawa dostępu mogą być podane na dwa sposoby:

- przy pomocy systemu kodów numerycznych (w formacie ósemkowym): 4 2 1
- przy pomocy systemu kodów znakowych: r w x

**slajd 47 z prezentacji**

### Domyślne prawa dostępu przy tworzeniu plików i katalogów

Domyślne prawa dostępu dla plików i katalogów nadawane są podczas ich tworzenia. Zmianę tych praw uzyskujemy poleceniem `umask`. Jeśli chcielibyśmy, aby tworzone pliki miały domyślne prawa 644, które zezwalają właścicielowi na czytanie i pisanie, a reszcie tylko na czytanie to od wartości 777 należy odjąć 644, a wynik podać jako parametr polecenia `umask`  
 $777-644 = 133$ :

```
umask 133
```

### suid i sgid - pożyteczne i niebezpieczne narzędzie

Zadaniem tego potężnego, a zarazem niebezpiecznego narzędzia jest uruchamianie programu (nie skryptu) z prawami właściciela lub grupy przypisanej temu programowi, a nie z prawami użytkownika, który ten program uruchamia.

Zagrożenie z używania tych flag może wynikać z możliwości przejęcia kontroli nad systemem. Jeśli zwykłemu użytkownikowi uda się tak zawiesić program (którego właścicielem jest użytkownik root i który ma ustawioną flagę `suid` lub `sgid`), aby dostać się do powłoki to otrzyma on prawa właściciela programu (czyli w tym przypadku użytkownika root) co stanowi ogromne zagrożenie dla systemu. Dlatego należy z dużym rozsądkiem używać tych flag.

Nadawanie plikom `suid` lub `sgid` wygląda następująco:

**suid**

```
chmod u+s nazwa_plku  
chmod 2*** nazwa_pliku
```

**sgid**

```
chmod g+s nazwa_pliku  
chmod 4*** nazwa_pliku
```

**suid i sgid**

```
chmod 6*** nazwa_pliku
```

gdzie \*\*\* oznacza dowolne prawa dla właściciela, grupy i innych użytkowników.

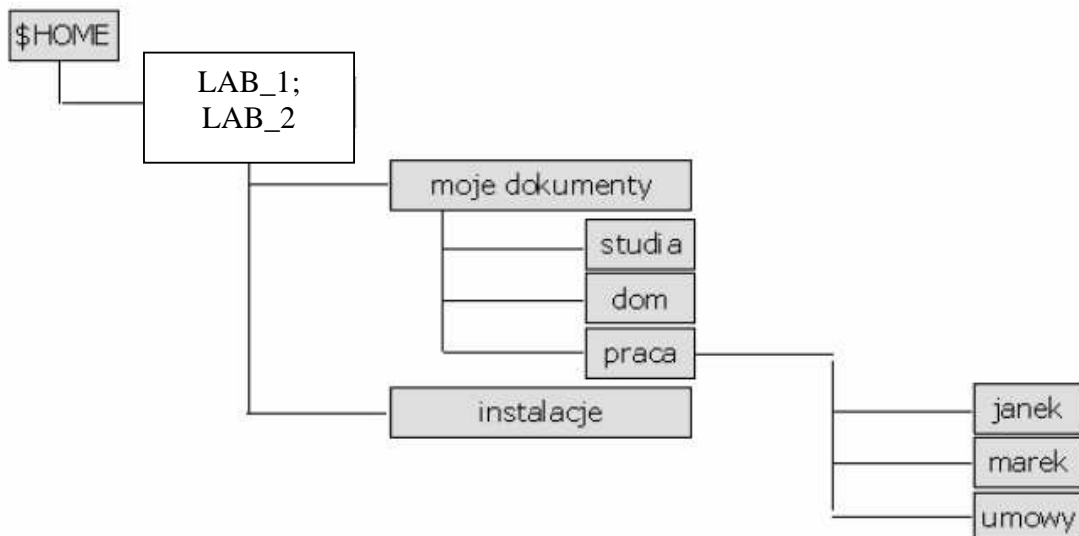


## ZADANIA cz.1.

1. Uruchom system Linux. Po uruchomieniu przełącz na dowolną konsolę tekstową.
2. Wyświetl zawartość katalogu bieżącego. Zbadaj różnice w działaniu poleceń *ls -l*, *ls -a* i *ls -la*. Czym różni się polecenie *ls* od polecenie *ls \**?
3. Zapoznaj się z reakcją systemu na błędnie wprowadzone polecenie np. *ls-la*
4. Zapoznaj się z mechanizmem dokończania poleceń za pomocą klawisza **tab**.
5. Korzystając z klawisza **tab** wyświetl wszystkie katalogi zaczynające się na *s*, które znajdują się w katalogu */etc*
6. Korzystając z klawisza **tab** wyświetl wszystkie polecenia zaczynając się na literę *m*.
7. Zapoznaj się z mechanizmem poruszania się po katalogach za pomocą ścieżki względnej i bezwzględnej.
8. Z katalogu domowego wylistuj jednym poleceniem zawartość katalogu */etc*

## ZADANIA cz.2.

1. Utwórz następującą strukturę katalogów:



LAB\_1 → grupa 1, LAB\_2 → grupa 2.

1. Skopiuj polecenia wszystkie polecenia dwuliterowe zaczynające się od litery *d* z katalogu */bin* do założonego wcześniej katalogu *studia*
2. zmień nazwy tych plików, aby zaczynały się od słów *stary\_*
3. załóż w swoim katalogu domowym plik *info.txt*, w którym zamieścisz podstawowe informacje o sobie – imię, nazwisko, kilka słów o sobie.
4. załóż drugi plik o nazwie *email.txt*, w którym umieść swój adres e-mail

5. doklej plik *email.txt* do pliku *info.txt* i zmień jego nazwę na *.plan*
6. utwórz w katalogu instalacje katalog *rpm* korzystając ze ścieżki bezwzględnej
7. utwórz w katalogu instalacje katalog *deb* korzystając ze ścieżki względnej
8. przejdź do katalogu *deb* (w katalog *instalacje*) a następnie przejdź za pomocą jednego polecenia do katalogu *umowy* znajdującego się w katalogu *praca*.
9. przenieś plik *email.txt* do katalogu *studia*
10. załóż nowy plik, a następnie stwórz link symboliczny do niego. Co się stanie z linkiem, jeśli usuniesz plik bazowy?
11. skopiuj do katalogu *janeek* wszystkie plik z katalogu *studia*
12. przenieś katalog *praca* wraz z podkatalogami do katalogu *instalacje* za pomoca jednego polecenia

### **Uprawnienia**

13. Utwórz w katalogu dom dwa pliki tekstowe plik1.txt oraz plik2.txt. Jakie uprawnienie domyślne mają pliki1.txt oraz plik2.txt?
14. Ustaw dla pliku plik1.txt uprawnienia rw-r--rw-.
15. Ustaw dla pliku plik2.txt następujące uprawnienia właściciel: odczyt, zapis, wykonanie, grupa: tylko zapis, pozostali: odczyt, zapis. Jak to zrobić korzystając z zapisu numerycznego?
16. Wyświetl uprawnienia plików plik1.txt oraz plik2.txt.
17. Wydaj polecenie chmod 574 dla plik1.txt? Jakie zostaną nadane mu prawa?
18. Wydaj polecenie chmod u=rw, g=r, o-r dla pliku plik2.txt. Jakie zostaną nadane mu prawa?
19. Wyświetl uprawnienia plików plik1.txt oraz plik2.txt. Które polecenie (pkt. 16 i 17) nadpisuje uprawnienia, a które je modyfikuje?

### **Po sprawdzeniu przez prowadzącego zajęcia:**

20. usuń cały katalog *studia*
21. usuń katalog *praca* wraz z podkatalogami za pomoca jednego polecenia
22. usuń katalog *instalacje*