

Pracownia Technologii Informacyjnej

LINUX

1. Podstawowe polecenia powłokowe

Podstawowe pojęcia:

1. Linia poleceń (interpreter linii poleceń, znak zachęty).

Linia poleceń to tradycyjny interfejs ze starszych komputerów. W tych modelach system uruchamiał program, zwany interpreterem linii poleceń, który jak wskazuje nazwa, interpretuje pisane przez nas polecenia/komendy i odnajduje właściwe programy uruchamiając je zgodnie z wypisanymi przez nas poleceniami. Po wykonaniu komendy interpreter wyświetla sekwencje znaków zwane znakami zachęty (np.: > lub \$). Wyświetlenie tych znaków oznacza, że program jest gotowy do przyjęcia następnej komendy.

2. Co to jest powłoka (ang. *shell*).

Powłoka to nazwa nadana każdemu linuxowemu interpreterowi poleceń, pochodzi z Unixa z którego wywodzi się Linux. Unix był pierwszym systemem operacyjnym, którego interpreter poleceń nie był wbudowany w system operacyjny. Ta idea została zastosowana (i stosuje się ją do dziś) w każdym systemie wywodzącym się z Unixa. Ponieważ powłoka jest samodzielnym narzędziem, każdy użytkownik może wybrać dowolną powłokę, która mu odpowiada. Co więcej uaktualnienie jej jest niezależne od systemu.

3. Rodzaje powłok: Powłoka Bourne'a (*sh*), Powłoka C (*cs*h), Powłoka Korna (*ksh*), Powłoka Bash (*bash*)

Powłoka Bourne'a zaprojektowana została ok. 1975 roku przez S.R. Bourne'a pracownika Bell Laboratories. Obecnie jest podstawowym interpreterem poleceń we wszystkich systemach Linux i Unix. Powłoka ta nie posiada wielu udogodnień (np. edycji wiersza poleceń) ale jest mniejsza od pozostałych i większość operacji wykonuje się wydajniej.

Powłoka C jest powłoką najczęściej używaną przez programistów, gdyż składnia jej poleceń przypomina język programowania C i zawiera wszystkie operatory warunkowe używane w tym języku. Powłoka ta w porównaniu z poprzednią została wzbogacona o pewne udogodnienia np.: udostępnia historię wydawanych poleceń, pozwala bezpośrednio obliczać i wykonywać warunki i polecenia wbudowane. Implementuje również mechanizm aliasów, które umożliwiają tworzenie alternatywnych poleceń i dają większą kontrolę nad programami wykonywanymi w tle.

Powłoka Korna jest unowocześnieniem Powłoki Bourne'a (została opracowana w 1983 w tym samym laboratorium co Powłoka Bourne'a). Zachowuje pełną funkcjonalność Powłoki Bourne'a, ale ma również wiele udogodnień Powłoki C i zaimplementowane ma też zupełnie nowe polecenia (np.: został udoskonalony sposób edycji wiersza poleceń (można go edytować za pomocą edytorów tekstu), czy sposób udostępniania historii poleceń (teraz można dostać się do dowolnego polecenia listy historii), można również sterować zadaniami jak również prowadzić bezpośrednie obliczenia wyrażeń warunkowych).

Powłoka Bash to skrót od Bourne-Again Shell. Posiada wszystkie udogodnienia Powłok Korna i C. Ma również zaimplementowane własne udogodnienia np.: udoskonalona edycja wiersza poleceń, która oferuje automatyczne uzupełnienia nazw plików i ścieżek dostępu. Obecnie jest używana jako domyślne narzędzie w większości systemów linuxowych.

Oprócz tych głównych istnieje jeszcze wiele innych powłok, które użytkownik może wybrać, część z nich jest po prostu połączeniem części możliwości w/w powłok, lub nawet są one okrojonymi wersjami tych powłok (np.: *rksh* czy *rsh*).

4. Co to są pliki i katalogi.

Pliki to pojedynczy zestaw informacji, z którego korzystają system operacyjny oraz programy. Pliki to pojemniki na różnego rodzaju dane.

Katalogi to pojemniki na zbiory plików. Mogą one zawierać również inne katalogi (podkatalogi). Taki hierarchiczny układ katalogów nazywamy strukturą drzewiastą i jest on opisywany jako lista katalogów, które kolejno się zawierają. Katalog, który jest katalogiem startowym przy opisywaniu takiej listy nazywamy katalogiem głównym.

5. Składnia poleceń: *polecenie -opcja argumenty*

Aby komendy wydawane przez nas były zrozumiałe dla systemu muszą mieć odpowiednią składnię: *polecenie -opcja argumenty*.

Polecenie – to komenda, dostępna w systemie, lub własny program użytkownika tzw. skrypt. Polecenia działają na plikach, katalogach i urządzeniach. Polecenie jest również plikiem, którego nazwa (rozszerzenie) powinno wskazywać na plik wykonywalny.

Opcja – to najczęściej pojedyncze litery poprzedzone myślnikiem, które modyfikują sposób funkcjonowania polecenia. Opcje można grupować np.:

```
ls -a -l
```

```
ls -al
```

Są polecenia, które po opcji następuje wartość (może być ona oddzielona spacją od opcji).

Argumenty – to pliki, katalogi i urządzenia na których wykonywane są polecenia. Można podać kilka argumentów, co wywoła realizację polecenia tyle razy ile zostało podanych argumentów.

6. Ścieżka dostępu

7. Katalog bieżący

Podstawowe komendy systemu Linux - nawigacja i operacje na katalogach i plikach:

mkdir – utworzenie nowego katalogu (składnia: **mkdir katalog**)

-p – tworzy „gałęzie” katalogów jeden w drugim

ls – wypisanie zawartości katalogu

-l – wyświetla informacje szczegółowe (rodzaj pliku, uprawnienia, nazwę właściciela, datę modyfikacji)

-a – wyświetla wszystkie pliki, także pliki ukryte (ich nazwa zaczyna się od kropki)

-s – wyświetla dodatkowo rozmiar plików

-R – rekurencyjne wyświetlanie zawartości katalogu (wraz z podkatalogami)

-t – posortowanie wyniku według czasu modyfikacji pliku

-s – posortowanie wyniku według rozmiaru plików (może nie działać)

cd – przejście do wskazanego katalogu (składnia: **cd katalog lub ścieżka**)

.. – przejście do katalogu wyżej

klawisz tab – rozwija nazwę pliku lub katalogu

Ćwiczenia

1. W bieżącym katalogu utwórz katalog *imię_użytkownika* następnie wejdź do niego i utwórz katalog *zadanie*.

2. Przejdź do katalogu *zadanie* i utwórz trzy gałęzie katalogów: *kat1/kat2/kat3*, *cw1/cw2*, *zad1/zad2/zad3*.

3. Wyświetl zawartość katalogu *zadanie*.

4. Przejdź do katalogu *zad3* za pomocą ścieżki dostępu.

rmdir – usuwanie pustego katalogu, który nie jest katalogiem bieżącym

rm – usuwanie pliku lub katalogu

-i – usuwanie pliku z potwierdzeniem decyzji

-r – usuwanie pliku lub niepełnego katalogu bez potwierdzenia decyzji (usuwanie rekurencyjne)

-f – polecenie usunięcia pliku bez czekania na potwierdzenie

-v – polecenie informuje co zrobiło

touch – tworzy pusty plik

mv – przenosi pliki, może to być przeniesienie z katalogu do katalogu (składnia: **mv plik katalog lub ścieżka do tego katalogu**), lub zawartość pliku do nowego pliku (zmiana nazwy) (składnia: **mv plik1 plik2**)

`cp` –kopiuje plik (składnia: `cp plik nowy_plik`) lub katalog (składnia: `cp plik katalog`)
`-r` –kopiowanie rekurencyjne pozwala kopiować całe katalogi
`pwd` –wyświetla ścieżkę do bieżącego katalogu

Ćwiczenia

- Przejdź do katalogu *kat3* i utwórz w nim puste pliki: *plik1*, *plik2*, *plik3*, *plik4*, *plik5*
- Wyświetl zawartość katalogu *kat3* z rozmiarem plików i czasem modyfikacji.
- Skopiuj *plik1*, *plik2* do katalogu *cw1*.
- Przenieś *plik5* i *plik4* do katalogu *zad3* i zmień nazwę tych plików na *plik.a* i *plik.ab*.
- Utwórz *plik4* i *plik5* znowu w katalogu *kat3* znajdując się w katalogu *zad3*.
- Usuń *plik.a* z potwierdzeniem decyzji.
- Zmień nazwę pliku *plik2* w katalogu *cw1* na *plik.abc* będąc w katalogu *zad3* i skopiuj go do tego katalogu

Podstawowe pojęcia:

- Metaznaki
Metaznaki – znaki uogólniające. Służą do tworzenia uogólnień rozpoznawalnych przez powłoki. Słowo zawierające meta znaki jest zastępowane listą słów, które pasują do podanego uogólnienia.

Metaznaki

- * -dowolny ciąg złożony z zera lub większej liczby znaków
- ? -dowolny pojedynczy znak
- [abc...] -dowolny z otoczonych nawiasami znaków, myślnik może zostać użyty do określenia zakresu np.: a-z, B-G, 2-7.
- [!abc...] -dowolny znak oprócz wyszczególnionych

Ćwiczenia

- Przejdź do katalogu *kat3* i utwórz pliki: *plik4* i *plik5*, następnie użyj polecenia:

```
ls plik*  
ls plik?  
ls plik[123]
```

jaki jest wynik dla każdego z tych poleceń?
 - Jednym poleceniem przenieś wszystkie pliki do katalogu *kat2*, jakiego meta znaku użyjesz?
 - Skopiuj *plik1*, *plik2* i *plik3* do katalogu *kat1* jednym poleceniem, jakiego meta znaku użyjesz?
- ?(a|b|..) –0 lub jedno wystąpienie wzorca z listy. Lista umieszczona w nawiasach to zestaw wzorców (jeden lub więcej oddzielonych znakiem |.)
*(a|b|..) –0 lub więcej wystąpień wzorca z listy
+(a|b|..) –jedno lub więcej wystąpień wzorca z listy
@(a|b|..) –dokładnie jedno wystąpienie wzorca z listy
!(a|b|..) –dowolne ciągi znaków niezawierające wzorca z listy

Ćwiczenia

- W katalogu *zad3* twórz pliki: *plik.b*, *plik.cba*, *plik.cb*, *plik.ac*, *plik.cbb*. Następnie wyświetl tylko pliki z rozszerzeniem: a, ab, abc. Potem wyświetl wszystkie inne pliki za wyjątkiem tych z podanym powyżej rozszerzeniem.
- Posługując się wzorcami wyświetl w katalogu *zad3* wszystkie pliki mające w nazwie tekst „plik.cb”. Jaki jest wynik i dlaczego? Czy rzeczywiście są to wszystkie pliki? Jak wyświetlić to prościej? Co się stanie jeśli nie będzie podanego wzorca?
- Jaki będzie rezultat jeśli zamiast polecenia: `ls plik.cb+(a|b)*` użyjemy polecenie `ls plik.cb*(a|b)*`?
- Przejdź do katalogu *kat1*, stosując meta znak `?(a|b|..)` wyświetl *plik2* i *plik3*.
- Jaki inny metaznak można użyć aby osiągnąć rezultat taki jak w ćwiczeniu 18?

Metaznaki rozwijające nazwę ścieżek

- ~ -katalog główny użytkownika
- ~- -katalog poprzedni użytkownika
- ~+ -katalog bieżący użytkownika
- ~login -katalog główny użytkownika

Ćwiczenia

20. Wyświetl zawartość katalogu *głównego/domowego* przy pomocy meta znaku.
21. Jesteśmy w katalogu *zad3*. Wykonujemy polecenie przejścia do katalogu *zad1*.
Jeśli będąc w tym katalogu chcemy wyświetlić zawartość poprzedniego katalogu nie przechodząc do niego to jakiego meta znaku użyjemy?
22. Przejdź do katalogu *kat2* jednym poleceniem usuń z niego pliki: *plik1, plik2, plik3*.
Następnie za pomocą znaków rozwijających nazwy ścieżek przenieś pliki: *plik1, plik2, plik3* z katalogu *kat1* do *kat2*
23. Usuń katalog *kat3*.
24. Usuń katalog *cw1* z całą zawartością.