

## Polecenia systemu Unix/Linux

### Polecenia 3

1. uname
2. who
3. whoami
4. passwd
5. id
6. useradd (adduser)
7. userdel
8. groups
9. addgroup
10. delgroup

### Zarządzanie uprawnieniami w systemie Linux

W systemie Unix/Linux każdy plik posiada właściciela i grupę do której jest przyporządkowany oraz zestaw uprawnień definiujących dostęp właściciela, grupy oraz wszystkich pozostałych użytkowników. Uprawnienia te dotyczą możliwości odczytu (ang. read), zapisu (ang. write) oraz wykonywania (ang. execute) plików. Informacje o uprawnieniach plików uzyskamy wydając polecenie `ls -l` lub `stat`.

Przykład:

`ls -al`

```
fred@i7:/test$ ls -al
total 8
drwxr-xr-x  2 root root 4096 mar 20 23:40 .
drwxr-xr-x 20 root root 4096 mar 20 23:39 ..
-----  1 root root    0 mar 20 23:39 01.txt
-rw-r--r--  1 root root    0 mar 20 23:39 02.txt
-rwxr-xr-x  1 root root    0 mar 20 23:39 03.txt
-rwxrwxrwx  1 root root    0 mar 20 23:40 04.txt
---x-----  1 fred fred    0 mar 20 23:39 a.txt
-rw-rw-rw-  1 fred fred    0 mar 20 23:39 b.txt
-rw-r--r--  1 fred fred    0 mar 20 23:40 c.txt
```

W kolejnych kolumnach otrzymujemy następujące informacje:

10 znaków określających typ i uprawnienia pliku, ilość dowiązań do pliku, nazwa właściciela, nazwa grupy, rozmiar w bajtach, data i godzina modyfikacji oraz nazwa pliku.

Tab.1 - Znaczenie poszczególnych pozycji dotyczących uprawnień do pliku

|   | u |   |   | g |   |   | o |   |   |
|---|---|---|---|---|---|---|---|---|---|
| d | r | w | x | r | w | x | r | w | x |

Pierwszy znak oznacza typ pliku: d to katalog, - to zwykły plik.

Następne 9 znaków określa uprawnienia pliku (r - odczyt, w - zapis, x - wykonywanie) odpowiednio dla trzech zbiorów użytkowników: dla właściciela, dla grupy oraz dla wszystkich innych.

Polecenia pozwalające na zmianę uprawnień do pliku:

- chmod - zmienia prawa dostępu do pliku
  - ✓ Postać: chmod [opcje] prawa plik...
  - ✓ Polecenie służy do ustawiania praw odczytu (r), zapisu (w) i wykonywania (x) pliku. Prawa te można nadać jednemu z trzech zbiorów użytkowników: właścicielowi pliku (u), grupie (g) i całej reszcie (o). Można też zmienić prawa wszystkim użytkownikom (a). Możliwe są trzy operacje: (+) dodanie uprawnień, (-) cofnięcie uprawnień lub (=) zastąpienie starych uprawnień nowymi.
  - ✓ Składnia zmiany uprawnień powinna wyglądać tak: [[ugoa...][+|=][rwx...],...]

Przykłady:

- ✓ \$ chmod a+r dane.txt  
nadanie prawa do odczytu dla wszystkich
- ✓ \$ chmod u-x \*.sh  
cofnięcie prawa do wykonywania przez właściciela plików o rozszerzeniu .sh
- ✓ \$ chmod ug=r dane.txt  
nadanie prawa do odczytu przez właściciela i grupę
- ✓ \$ chmod ugo=rwx dane.txt  
nadanie wszystkim uprawnień do odczytu, zapisu i wykonywania pliku - równoważne poleceniu
- ✓ \$ chmod a=rwx dane.txt

Najważniejsze opcje:

- v wyświetla komunikat o każdym zmienionym uprawnieniu
- R zmień uprawnienia katalogów i całej ich zawartości

- ✓ Przykład:  
\$ chmod -R a+r ~  
zezwoli wszystkim na możliwość czytania wszystkich plików w domowym katalogu

- ✓ Innym sposobem nadawania uprawnień jest zapis numeryczny, np:

✓ \$ chmod 703 dane.txt

Pierwsza cyfra określa uprawnienia użytkownika, druga - grupy a trzecia - reszty. Wartość (od 0 do 7) oznacza rodzaj uprawnień: wykonywanie (1), zapis (2) lub odczyt (4). W celu nadania kilku uprawnień należy zsumować odpowiednie cyfry. W powyższym przykładzie 7 oznacza nadanie praw do odczytu, zapisu i wykonywania (4+2+1) dla właściciela, 0 oznacza, że grupa nie posiada żadnych uprawnień a 3 odpowiada nadaniu prawa do zapisu i wykonywania (2+1) dla pozostałych użytkowników.

- chown - zmienia właściciela i grupę pliku

✓ Postać: chown [opcje] użytkownik[:grupa] plik...

- chgrp - zmienia grupę użytkowników pliku

✓ Postać: chgrp [opcje] grupa plik...

- Uwaga: polecenia chown oraz chgrp mogą być niedostępne dla zwykłego użytkownika.

- umask - ustawienie maski uprawnień tworzenia plików i katalogów

✓ Postać: umask [tryb]

✓ Polecenie

\$ umask

wypisze domyślnie ustawioną maskę w notacji ósemkowej.

✓ \$ umask 027

ustawi maskę na wartość ósemkową 027

✓ \$ umask -S

wypisze ustawienia maski w postaci symbolicznej rwxrwxrwx

✓ \$ umask u=rwx,g=rx,o=

ustawi maskę 027 w zapisie symbolicznym

✓ Maska jest używana do nadawania domyślnych ustawień nowych plików i katalogów. Domyślne uprawnienia tworzenia plików bez maski to 666 a dla katalogów to 777. Maska jest odejmowana (bitowo) od domyślnych wartości.

Przykłady masek:

|                                  |               |
|----------------------------------|---------------|
| ✓ umask                          | 027 ----w-rwx |
| ✓ dopełnienie maski              | 750 rwxr-x--- |
| ✓ domyślna wartość dla plików    | 666 rw-rw-rw- |
| ✓ uprawnienia nowego pliku       | 640 rw-r----- |
| ✓ domyślna wartość dla katalogów | 777 rwxrwxrwx |
| ✓ uprawnienia nowego katalogu    | 750 rwxr-x--- |