

Uprawnienia plików i katalogów - chmod i chowa

1. Podstawy modelu uprawnień

Każdy plik i katalog ma przypisane uprawnienia dla trzech grup użytkowników:

- **u** (user) – właściciel pliku
- **g** (group) – grupa, do której należy plik
- **o** (others) – wszyscy pozostali użytkownicy

Dla każdej z tych grup można przypisać trzy typy uprawnień:

- **r** (read) – prawo do odczytu
- **w** (write) – prawo do zapisu/modyfikacji
- **x** (execute) – prawo do wykonania (dla plików) lub wejścia (dla katalogów)

2. Wyświetlanie uprawnień

```
ls -l plik.txt
```

Wyświetla szczegółowe informacje, w tym uprawnienia w formacie:

```
-rwxr-xr--
```

Gdzie:

- Pierwszy znak: typ pliku (**-** zwykły plik, **d** katalog, **l** link symboliczny)
- Kolejne 3 znaki: uprawnienia właściciela (**rwx**)
- Kolejne 3 znaki: uprawnienia grupy (**r-x**)
- Ostatnie 3 znaki: uprawnienia innych (**r--**)

3. Komenda `chmod` – zmiana uprawnień

Format symboliczny (literowy):

```
chmod u+x skrypt.sh
```

Dodaje właścicielowi prawo wykonywania pliku `skrypt.sh`.

```
chmod g-w dokument.txt
```

Odbiera grupie prawo do zapisu w pliku `dokument.txt`.

```
chmod o+r raport.pdf
```

Dodaje innym użytkownikom prawo do odczytu pliku `raport.pdf`.

```
chmod a+r plik.txt
```

Dodaje prawo do odczytu dla wszystkich (właściciel, grupa, inni).

```
chmod go-rwx prywatny.txt
```

Odbiera grupie i innym wszystkie uprawnienia do pliku.

Format numeryczny (oktalny):

Każde uprawnienie ma wartość liczbową:

- r (read) = 4
- w (write) = 2
- x (execute) = 1

Sumując wartości, otrzymujemy kod uprawnień dla każdej grupy:

- 7 = rwx (4+2+1)
- 6 = rw- (4+2)
- 5 = r-x (4+1)
- 4 = r-- (4)
- 0 = --- (brak uprawnień)

```
chmod 755 skrypt.sh
```

Właściciel: rwx (7), grupa: r-x (5), inni: r-x (5).

```
chmod 644 dokument.txt
```

Właściciel: rw- (6), grupa: r-- (4), inni: r-- (4).

```
chmod 700 prywatny_skrypt.sh
```

Tylko właściciel ma pełny dostęp, inni nie mają żadnych uprawnień.

4. Rekurencyjna zmiana uprawnień

```
chmod -R 755 /var/www/strona
```

Opcja (recursive) zmienia uprawnienia dla katalogu i wszystkich plików oraz podkatalogów w nim zawartych.

```
chmod -R u+rwX /home/user/dokumenty
```

Duże dodaje prawo wykonywania tylko do katalogów i plików, które już mają to prawo dla kogokolwiek.

5. Komenda `chown` – zmiana właściciela i grupy

```
chown user plik.txt
```

Zmienia właściciela pliku na użytkownika `user`.

```
chown user:grupa plik.txt
```

Zmienia właściciela na `user` i grupę na `grupa`.

```
chown :grupa plik.txt
```

Zmienia tylko grupę (właściciel pozostaje bez zmian).

```
chown -R www-data:www-data /var/www
```

Rekurencyjnie zmienia właściciela i grupę dla całego katalogu i jego zawartości.

6. Praktyczne przykłady zastosowań

```
# Nadanie uprawnień wykonywania dla skryptu
chmod +x deploy.sh

# Zabezpieczenie pliku konfiguracyjnego (tylko właściciel może czytać i pisać)
chmod 600 config.ini

# Typowe uprawnienia dla katalogu www
chmod -R 755 /var/www/html
chown -R www-data:www-data /var/www/html

# Odebranie wszystkich uprawnień innym użytkownikom
chmod go-rwx /home/user/tajne_dane

# Sprawdzenie uprawnień przed i po zmianie
ls -l plik.txt
chmod 644 plik.txt
ls -l plik.txt
```

Revision #1

Created 23 November 2025 14:17:29 by Tomasz Konieczny

Updated 23 November 2025 14:18:38 by Tomasz Konieczny