

T: Nadzorowanie uprawnień plików.

Domyślna maska uprawnień zapisana jest w plikach: `”~/bash_profile”` lub `”~/profile”` oraz w `”/etc/skel/.bash_profile”` dla nowotworzonych kont użytkowników.

UMASK - to maska uprawnień nowotworzonych plików (w momencie tworzenia nowego pliku). W większości dystrybucji domyślny umask to 022, co oznacza, że nowoutworzony plik będzie miał uprawnienia 0644.

Obliczenia: $0666-022 = 0644$ (rw-r-xr-x). Tak więc 022 to nasz umask (maska) uniemożliwiający zapis do pliku użytkownikom innym niż właściciel.

Wady maski 022 - Maska 022 wymaga, by użytkownik sam ”pilnował” uprawnień nowotworzonych plików prywatnych i odbierał innym prawo ich odczytu jeśli nie chce, by mogli oni zobaczyć ich zawartość. Maska 077, to nic innego jak odebranie pozostałym użytkownikom jakichkolwiek uprawnień ($0666-077 = 0600$ (rw-----)). Dzięki takiemu rozwiązaniu, żaden z pozostałych użytkowników nie będzie mógł odczytać przypadkiem naszych plików, jednak dla plików, które będziemy później udostępniać innym użytkownikom musimy ręcznie (chmod) nadać odpowiednie uprawnienia.

Sprawdzenie bieżącej maski:

```
umask
umask -S
```

Zmiana bieżącej maski z wiersza poleceń:

```
umask 077
umask -p u=rwx,g-rwx,o-rwx
```

Domyślne uprawnienia do plików i katalogów możemy konfigurować w pliku `/etc/fstab`. Potrzebne informacje na temat tego pliku uzyskasz podając w linii komend następujące polecenie

```
man fstab
```

Ręcznej konfiguracji uprawnień do plików dokonamy za pomocą polecenia

```
chmod
```

Ręczna konfiguracja praw własności (właściciela i grupę właścicieli) dokonamy za pomocą polecenia

```
chown
```

Domyślne uprawnienia do katalogów systemowych:

```
chmod 755 /bin
chmod 750 /boot
chmod 751 /etc
chmod 751 /home
chmod 751 /lib
chmod 751 /lib/modules
chmod 750 /mnt
chmod 750 /root
chmod 750 /sbin
chmod 755 /usr/bin
chmod 755 /usr/doc
chmod 751 /usr/etc
chmod 751 /usr/include
chmod 751 /usr/info
chmod 751 /usr/lib
chmod 751 /usr/man
chmod 750 /usr/sbin
chmod 750 /usr/src
chmod 755 /usr/local/bin
chmod 751 /usr/local/etc
chmod 751 /usr/local/games
chmod 751 /usr/local/include
chmod 751 /usr/local/info
chmod 751 /usr/local/lib
chmod 751 /usr/local/man
chmod 750 /usr/local/sbin
```

```
chmod 750 /usr/local/src
```

Polecenie **ls -l** wyświetli szczegółowe informacje na temat zawartości katalogu bieżącego. W wyświetlonych w ten sposób danych zobaczymy mniej więcej:

```
-rwx-w-r-x 2 root users 132 Oct 4 12:36 plik.txt
```

gdzie:

- pierwszy znak określa czy jest to plik (-), katalog (d) lub skrót (l),
- następne trzy pola (rwx) określają prawa właściciela pliku do pliku,
- kolejne trzy pola (-w-) oznaczają prawa grupy do pliku
- dalsze trzy pola (r-x) określają prawa pozostałych użytkowników (jakichkolwiek) przysługujące do pliku,
- następne pole (2) określa ilość dowiezań do pliku występujących w systemie,
- kolejne pole (root) określa właściciela pliku,
- następne pole (users) określa grupę właścicieli pliku,
- dalsze pole (132) opisuje objętość obiektu w systemie w B,
- pola Oct 4 12:36 oznacza datę i czas ostatniej modyfikacji pliku,
- ostatnie pole plik.txt oznacza nazwę pliku (lub katalogu).

Kolejne prawa oznaczają:

- r - czytanie,
- w - zapisywanie,
- x - wykonywanie,
- - - prawo wyłączone.

Zmian praw dostępu dokonujemy poleceniem

```
chmod 753 plik
```

gdzie:

- 7 - prawa właściciela,
- 5 - prawa grupy,
- 3 - prawa reszty świata.

Stosowane wartości liczbowe oznaczają:

- 0 = ---,
- 1 = --x,
- 2 = -w-,
- 3 = -wx,
- 4 = r--,
- 5 = r-x,
- 6 = rw-,
- 7 = rwx.

Inne przykłady wykorzystania polecenia:

```
chmod a+r plik
chmod g-x plik
chmod o=rw
```

gdzie:

- u - użytkownik - właściciel pliku,
- g - grupa do której przynależy plik,
- o - pozostali użytkownicy,
- a - wszyscy - użytkownik, grupa i pozostali użytkownicy,
- + - dodanie uprawnień,
- - - zdjęcie uprawnień,
- = - dokładne określenie uprawnień.

Właściciela i grupę właścicieli pliku modyfikujemy poleceniem

```
chown pracownia:klasa plik
```

gdzie:

- pracownia oznacza nowego właściciela pliku (z dostępnych w pliku /etc/passwd),
- klasa nową grupę właścicieli pliku (z dostępnych w pliku /etc/group).

Przykłady wykorzystania polecenia:

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