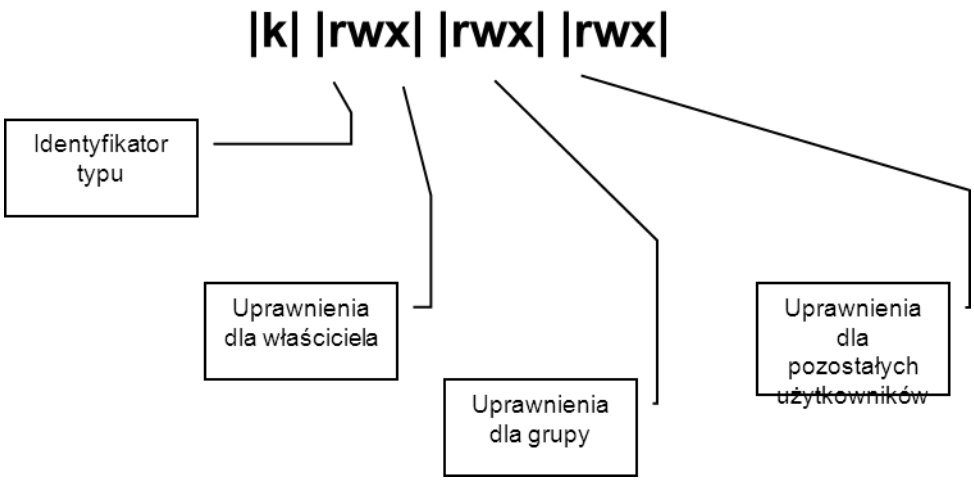
Linux

Uprawnienia pliku / katalogu,

właściciel pliku, UID, GID, sticky bit.

Spis treści

Ogólny schemat uprawnień.



Aby wyświetlić uprawnienia wydaj polecenie **ls –l**

Identyfikatory typu

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

Sposoby nadawania uprawnień

Prawa dostępu mogą być nadawane na dwa sposoby:
a) system kodów znakowych **r,w,x**
b) system kodów numerycznych **4,2,1** (zwany systemem oktetowym)

Uprawnienia znakowe i ich znaczenie.

prawo	**Katalog**	**plik**
R	przeszukanie zawartości	czytania
W	zmiany zawartości	modyfikacji
X	otwarcia katalogu	uruchomienia

Zapis uprawnień systemem numerycznym i znakowym.

Prawa dostępu	**Zapis numeryczny**	**Zapis znakowy**
tylko do odczytu	4	r--
tylko do pisania	2	-w-
tylko do uruchamiania	1	--x
do czytania i pisania	6	rw-
do czytania i uruchamiania	5	r-x
wszystkie uprawnienia	7	rwx

chmod - nadawanie, odbieranie uprawnień

Polecenie służące do nadania, zmiany praw dostępu to **chmod.**

Składnia tego polecenia wygląda następująco:

chmod [-opcje] [uprawnienie] plik/katalog

Przy zapisie znakowym należy pamiętać, że uprawnienie określamy w następujący sposób:

podajemy użytkownika/ów („**a**”-wszyscy, „**u**”-użytkownik, „**g**”-grupa, „**o**”-inni), oraz

określamy czynność („**+**” nadanie praw, „**-**” odebranie praw).

Przykłady: nadajemy właścicielowi pliku prawo do jego czytania:

chmod 400 nazwa_pliku

chmod u+r nazwa pliku

Nadajmy teraz wszystkie prawa właścicielowi, a grupie tylko prawo odczytu pliku:

chmod 740 nazwa_pliku

chmod u+rwx, g+r nazwa pliku

Uprawnienia ACL.

Aby nadać pojedynczym użytkownikom specyficzne uprawnienia, inne niż te ustalone np. dla grupy, czy pozostałych użytkowników możemy skorzystać z tzw. uprawnień ACL.

Odczyt uprawnień: **getfacl nazwa_obiektu**

```
linux-3kiy:~ # getfacl plik.txt
# file: plik.txt
# owner: root
# group: root
user::rw-
user:pkania:rwx
group::r--
mask::rwx
other::r--
```

Nadanie uprawnień: **setfacl –m user:pkania:rwx plik.txt**

Dodatkowe uprawnienia acl symbolizowane są również znakiem **+** po wydaniu polecenia ls –l

```
-rw-rwxr--+ 1 root root     0 Oct  7 15:14 plik.txt
```

Modyfikacja (zmiana) uprawnień: **setfacl –m u:pkania:r plik.txt**

Usunięcie uprawnień: **setfacl –x u:pkania plik.txt**

Najważniejsze parametry setfacl to:

- -m - modyfikuje/dodaje wpis na liście ACL,
- -x - usuwa wpis z listy ACL,
- -d - usuwa całą zawartość listy ACL,
- -b - usuwa całkowicie rozszerzone uprawnienia z listy ACL,
- -k - usuwa domyślne uprawnienia z listy ACL.

Uprawnienia domyślne (dotyczy katalogów) – powodują automatyczne nadanie rozszerzonych uprawnień do nowotworzonych obiektów wewnątrz tego katalogu.
Uprawnienia domyślne do katalogu nadajemy następująco:
setfacl –d –m group:pracownicy:wx katalog_testowy.

Modyfikacja i usuwanie uprawnień domyślnych katalogu jest analogiczne jak w powyższych przykładach dla pliku.

Typ	Definicja
właściciel	u[ser]::rwx
dowolny użytkownik	u[ser]:nazwa:rwx
grupa będąca właścicielem	g[roup]::rwx
dowolna grupa	g[roup]:nazwa:rwx
maska	m[ask]::rwx
inni	o[ther]::rwx
domyślne prawa dla właściciela	d[efault]:u[ser]::rwx
domyślne prawa dla dowolnego użytkownika	d[efault]:other-userid:rwx
domyślne prawa dla grupy będącej właścicielem	d[efault]:g[roup]::rwx
domyślne prawa dla innych	d[efault]:g[roup]:other-groupid:rwx

(wyrażenie "user", "group" oraz "default" w definicji można podać całym wyrazem lub również skrótowo "u", "g" i "d").

Uwaga ! Aby móc stosować uprawnienia ACL dana partycja musi być odpowiednio zamontowana manualnie lub przez plik /etc/fstab (opcja acl przy danym punkcie montowania).

Właściciel pliku/katalogu

Prawo do własności pliku (zmiana właściciela pliku) możemy zmienić poleceniem **chown** (odniesienie do pojedynczego użytkownika), lub **chgrp** (odnosi się do grupy).

Składnia powyższych poleceń przedstawia się następująco:
chown nazwa_użytkownika nazwa pliku
chgrp nazwa_grupy nazwa_pliku

Można też tak: chown właściciel.grupa nazwa_pliku. W ten sposób ustawiamy naraz właściciela i grupę (oddzielamy ich kropką).

Bit „lepkości” - sticky bit

Aby chronić plik/katalog przed usunięciem przez użytkownika, który nie jest właścicielem pliku/katalogu można użyć tzw. bitu lepkości (ang. sticky).
Dla pliku/katalogu nadanie tego bitu wygląda następująco:

chmod **1**751 nazwa_pliku

chmod **o+t** nazwa_katalogu

Jażeli chcemy usunąć bit lepkości wydajemy polecenie: chmod **–t** nazwa_pliku.

Zmiana domyślnych prawa dostępu przy tworzeniu plików / katalogów.

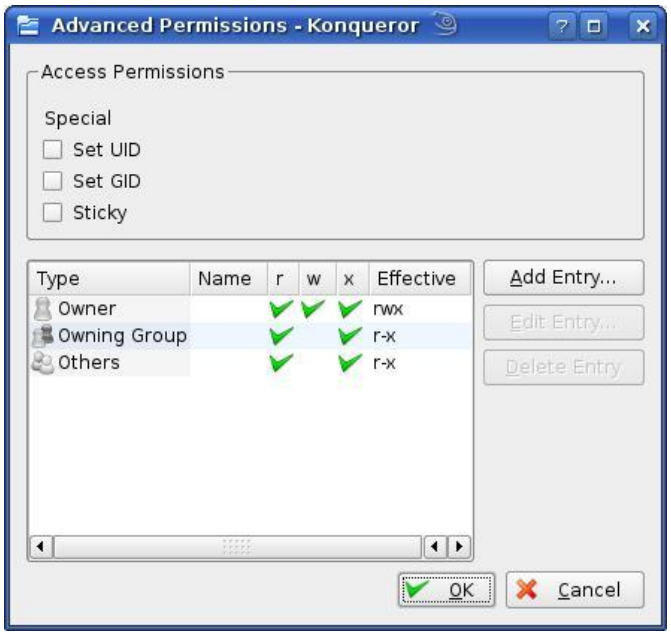
Podczas tworzenia plików, katalogów nadawane są im domyślne prawa dostępu.

Jeżeli chcemy to zmienić użyjemy polecenia: umask. Załóżmy, że chcemy aby domyślnie prawa wyglądały następująco: **644** to wydajemy polecenie **umask 133** ponieważ stosujemy tu operację odejmowania : 777 (max uprawnienia) – 644 (uprawnienia oczekiwane) = 133.

	Directories	Files
Default Permissions	rwx rwx rwx 7 7 7	rw- rw- rw- 6 6 6
umask	--- -w- -w- 0 2 2	--- -w- -w- 0 2 2
Result	rwx r-x r-x 7 5 5	rw- r-- r-- 6 4 4

Yast a uprawnienia, właściciel itp.

Pracując w konsoli graficznej po kliknięciu na pliku bądź katalogu wybieramy właściwości, zakładka permissions.

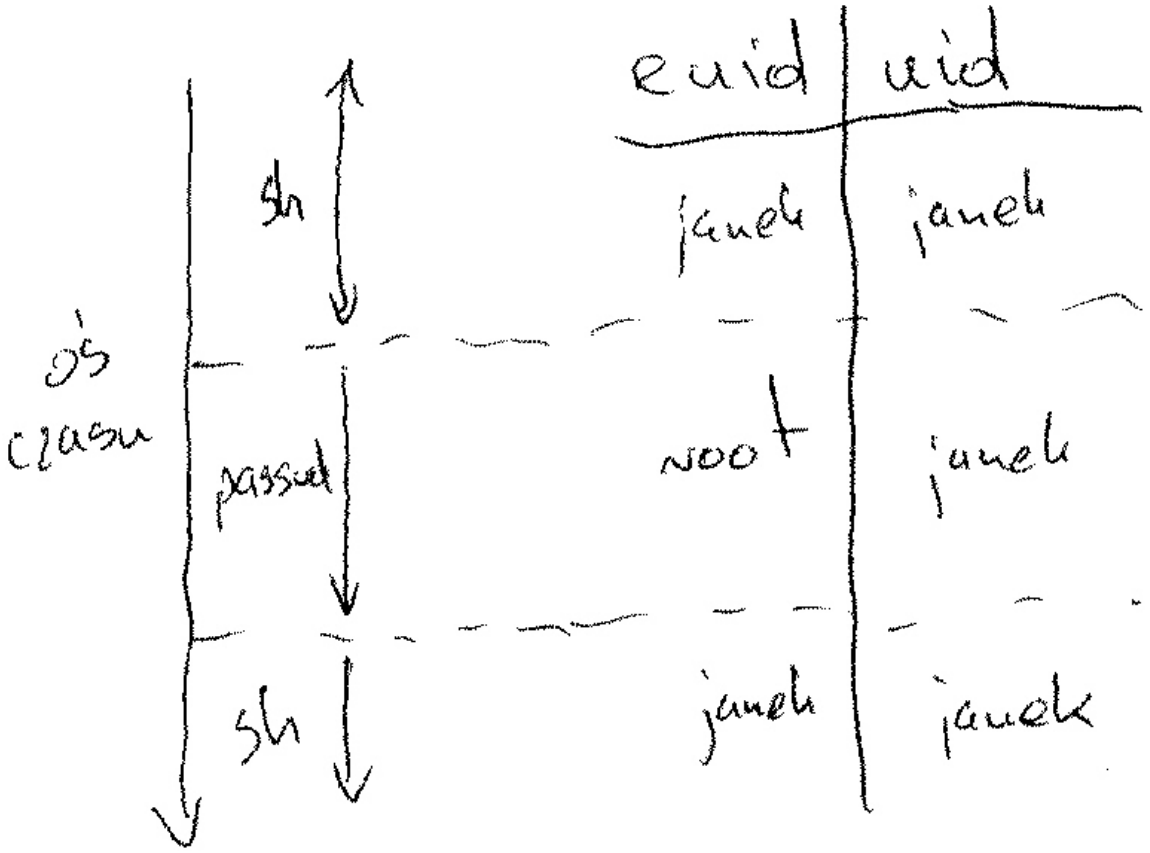


UID, GID.

Warto jeszcze wspomnieć o identyfikatorach użytkowników UID, oraz grup GID, które wskazują jednoznacznie na właściciela i grupę. Dla super użytkownika parametr UID wynosi „**0**”. Każde konto z UID = 0 będzie miało nieograniczone uprawnienia niezależnie od nazwy konta. Aby wyjaśnić znaczenie tych identyfikatorów podczas dostępu do plików prześledźmy proces tworzenia pliku przez użytkownika. Użytkownik zapisuje plik w NFS. System automatycznie zapisze wraz z plikiem UID użytkownika. W ten sposób zapamięta, kto jest właścicielem tego pliku. Podczas późniejszej próby otwarcia pliku system sprawdza UID. Jeżeli UID użytkownika otwierającego plik jest identyczne z UID zapisanym podczas tworzenia pliku system stwierdza, że to właściciel i otrzymuje on pełny dostęp do niego. Jeżeli GID pokryje się z GID pliku, wówczas zostaną przydzielone użytkownikowi prawa określone dla grupy. Bardzo ważnym zagadnieniem, którego nie można pominąć są tryby dostępu SUID (set user ID – zwiększone prawo dostępu użytkownika), oraz SGID (set group ID – zwiększone prawo dostępu dla grupy). Obydwa w/w tryby zapewniają czasowe przyznanie rozszerzonych uprawnień. Uprawnienia takie są przydzielane np. do pliku /etc/passwd podczas własnoręcznej zmiany hasła, przez użytkownika. Na czas wprowadzania nowego hasła „zwykły” użytkownik przejmuje uprawnienia roota. Administrator powinien kontrolować, czy pliki wykonywalne nie mają ustawionych SUID, oraz SGID, gdyż taka konfiguracja pozwalałaby na przejęcie w niektórych przypadkach uprawnień super użytkownika.

SUID – na czym to polega.

Przykładem wykorzystania SUID jest program passwd, który służy do zmiany hasła użytkownika. Jeżeli chcemy, aby użytkownik sam zmienił swoje hasło musi być ustawiony SUID na tym programie, ponieważ program ten modyfikuje plik /etc/shadow do którego uprawnienia ma tylko użytkownik root. Jednak na czas wykonywania programu passwd zwykły użytkownik otrzymuje uprawnienia root-a aby była możliwa modyfikacja pliku /etc/shadow. Jak można zauważyć uprawnienia SUID są bardzo niebezpieczne ponieważ powodują przejęcie uprawnień super użytkownika. Specjalne metody np. zawieszenia aktualnie wykonywanego programy mogą powodować uzyskanie dostępu do powłoki z uprawnieniami super użytkownika.



Reprezentacja bitu SUID, SGUID w listingach obiektów.

Flaga SUID w listingach plików oznaczona jest literą „s” w prawach dla właściciela pliku:

- rw**s** r-x r-x

Flaga SGID reprezentowana jest również przez literę „s” z tym, że widnieje w sekcji uprawnień dla grupy:

- rwx r-**s** r-x

Przykład :

```
laptop:~ # cd /usr/bin
laptop:/usr/bin # ls -l passwd
-rwsr-xr-x 1 root shadow 77096 Nov 29 18:47 passwd
laptop:/usr/bin #
```

Nadawanie, odbieranie uprawnień SUID, GUID.

chmod u+s nazwa_pliku.
chmod g+s nazwa_pliku.
Aby zdjąć powyższe bity należy w powyższych przykładach zamiast znaku + (plus) zastosować znak – (minus).

Monitorowanie możliwych zagrożeń związanych z SUID GUID.

Administrator systemu powinien weryfikować zasadność ustawienia w/w bitów dla obiektów. Aby wyszukać obiekty posiadające ustawiony jakikolwiek dodatkowy bit możesz skorzystać z następujących poleceń:

find / -perm -04000 ! –type l –ls dla SUID

find / -perm -02000 ! –type l –ls dla GUID

find / -perm -01000 ! –type l –ls dla sticky bit