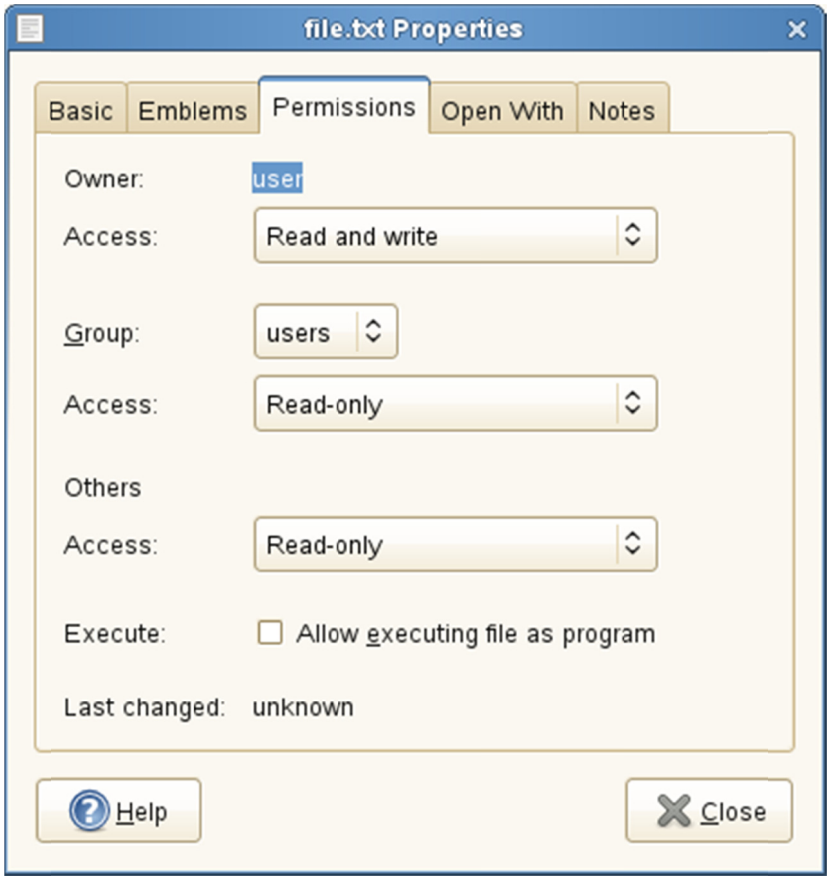
Współczesne systemy komputerowe

Prawa dostępu

1. Interfejs graficzny.

- **PPM** (*prawy przyciski myszy*) na katalogu lub pliku i następnie **Properties** zakładka ***Permissions***;



2. Powłoka tekstowa.

- Wyświetlanie praw dostępu do pliku, polecenie **`ls`**;

```
user@suse:~> touch file.txt
user@suse:~> ls -l file.txt
-rw-r--r-- 1 user users 0 Mar 21 18:28 file.txt
user@suse:~> su -
Password:
suse:~ # touch file.txt
suse:~ # ls -l file.txt
suse:~ # mkdir dir
suse:~ # ln -s file.txt symbolic-link
suse:~ # ls -l
...
drwxr-xr-x 2 root root 4096 Mar 21 18:31 dir
```

```
-rw-r--r-- 1 root root    0 Mar 21 18:29 file.txt
lrwxrwxrwx 1 root root    8 Mar 21 18:33 symbolic-link -> file.txt
```

- Składnia: typ pliku, prawa dostępu właściciela pliku, prawa dostępu grupy właściciela pliku, prawa dostępu dla innych użytkowników;

```
[ d | - | l ][ r w x ][ r w x ][ r w x ][ owner ][ group ]
```

d	katalog
-	plik
l	link
r	czytanie pliku, listowanie katalogu
w	zmiana pliku, tworzenie i kasowanie plików w katalogu
x	wykonanie pliku, zmiana katalogu

- Zmiana uprawnień pliku, poleceniem **chmod** polega na dodaniu **+**, odebraniu **-** lub ustawieniu **=** poszczególnych uprawnień **rwx** dla właściciela **u**, grupy **g**, innych użytkowników **o** lub wszystkich **a**, przełącznik **-R** pozwala na zmianę rekursywnie;

`chmod u+x`	dodanie prawa do uruchamiania dla właściciela
`chmod g=rw`	ustawienie prawa dla grupy do czytania i zapisywania
`chmod u=rwx`	ustawienie wszystkich praw dla właściciela
`chmod u=rwx,g=rw,o=r`	ustawienie wszystkich praw dla właściciela, czytanie i zapisywanie dla grupy i czytanie dla wszystkich
`chmod +x`	dodanie prawa do uruchamiania dla wszystkich użytkowników (zależy od wartości **umask**)
`chmod a+x`	dodanie prawa do uruchamiania dla wszystkich użytkowników

```
suse:~ # ls -l file.txt
-rw-r--r-- 1 root root 0 Mar 21 18:29 file.txt
suse:~ # chmod g+w file.txt
suse:~ # ls -l file.txt
-rw-rw-r-- 1 root root 0 Mar 21 18:29 file.txt
```

- Zmiana uprawnień do pliku, polecenie **chmod**;
  - r = 4;
  - w = 2;
  - x = 1;

rwx	421	4 + 2 + 1 = 7
r-x	4-1	4 + 1 = 5
rw-	42-	4 + 2 + =6
r--	4--	4

```
suse:~ # ls -l file.txt
-rw-rw-r-- 1 root root 0 Mar 21 18:29 file.txt
suse:~ # chmod 754 file.txt
suse:~ # ls -l file.txt
-rwxr-xr-- 1 root root 0 Mar 21 18:29 file.txt
suse:~ # chmod 777 file.txt
suse:~ # ls -l file.txt
-rwxrwxrwx 1 root root 0 Mar 21 18:29 file.txt
```

- Zmiana właściciela i grupy pliku, polecenie **chown**;

```
suse:~ # ls -l file.txt
-rwxrwxrwx 1 root root 0 Mar 21 18:29 file.txt
suse:~ # chown user.users file.txt
suse:~ # ls -l file.txt
-rwxrwxrwx 1 user users 0 Mar 21 18:29 file.txt
suse:~ # chown .root file.txt
suse:~ # ls -l file.txt
-rwxrwxrwx 1 user root 0 Mar 21 18:29 file.txt
suse:~ # chown root file.txt
suse:~ # ls -l file.txt
-rwxrwxrwx 1 root root 0 Mar 21 18:29 file.txt
```

3. Domyślne prawa dostępu.

- Domyślne prawa dostępu dla nowo tworzonego pliku wynoszą **666** a katalogu **777**, można zmienić ustawienia domyślne za pomocą polecenia **umask** (prawa zdefiniowane w **umask** są odejmowane od domyślnych);

domyślne prawa dostępu	rwx	rwx	rwx	rw-	rw-	rw-
	7	7	7	6	6	6
umask	---	-w-	-w-	---	-w-	-w-
	0	2	2	0	2	2
wynik	rwx	r-x	r-x	rw-	r--	r--
	7	5	5	6	4	4

```
suse:~ # umask
0022
suse:~ # touch file1
suse:~ # mkdir dir1
suse:~ # umask 000
suse:~ # touch file2
suse:~ # mkdir dir2
suse:~ # umask 777
suse:~ # touch file3
suse:~ # mkdir dir3
drwxr-xr-x 2 root root 4096 Mar 22 14:16 dir1
drwxrwxrwx 2 root root 4096 Mar 22 14:16 dir2
d--------- 2 root root 4096 Mar 22 14:16 dir3
-rw-r--r-- 1 root root    0 Mar 22 14:15 file1
-rw-rw-rw- 1 root root    0 Mar 22 14:16 file2
---------- 1 root root    0 Mar 22 14:16 file3
```

4. Prawa specjalne.

- Dla plików i katalogów można ustawiać tzw. prawa specjalne: **Sticky bit**, **SGID** (set GroupID) i **SUID** (set UserID);
  - Sticky bit: `t = 1`;
  - SGID: `s = 2`;
  - SUID: `s = 4`;

	pliki	katalogi
Sticky bit	-	użytkownik może usunąć pliki których jest właścicielem lub gdy jest właścicielem katalogu w którym znajdują się pliki (patrz **/tmp**)
SGID	program uruchamia się z identyfikatorem grupy właściciela	pliki tworzone w katalogu mają identyfikator grupy właściciela katalogu
SUID	program uruchamia się z identyfikatorem użytkownika właściciela	-

```
suse:~ # ls -ld /tmp
drwxrwxrwt 19 root root 4096 Mar 23 12:45 /tmp
```

- Każdy użytkownik może zmienić hasło (przy pomocy **passwd**), ale żeby je zapisać do pliku **/etc/shadow**, trzeba mieć prawa użytkownika *root*;

```
suse:~ # ls -l /usr/bin/passwd
-rwsr-xr-x 1 root shadow 80268 Feb  1  2012 /usr/bin/passwd
```

- Poleceniem **wall** można wysłać komunikat do wszystkich wirtualnych terminali, polecenie uruchamia się z prawami grupy *tty*;

```
suse:~ # ls -l /usr/bin/wall
-rwxr-sr-x 1 root tty 14044 Jan 13  2012 /usr/bin/wall
```