

## 4 - Linux jogosultságok

- Jogosultságok
- minden állománynak van tulajdonosa és csoportja
- mindezekhez tartozik
  - **r olvasási jog**
    - Ha a tulajdonosnak az olvasási jogát jelző kapcsoló be van kapcsolva, akkor az adott állományban található adatokat a tulajdonos megnézheti, olvashatja, kinyomtathatja, lemásolhatja, egy szóval az adatokhoz hozzáférhet. Ha a kapcsoló ki van kapcsolva, akkor az állomány nevét látja ugyan a tulajdonos, de a tartalmához nem férhet hozzá. A könyvtáron értelmezett olvasási jog a benne található állományok nevének kilistázását jelenti. Ha ezzel a joggal rendelkezik valaki akkor listázhatja az állományokat az adott könyvtárban, ha nem, akkor ezt nem teheti meg. Ötletes módon, az állományokat esetleg létrehozhatja, és akár módosíthatja is, attól függetlenül, hogy róla listát nem kaphat.
  - **w írási jog**
    - A tulajdonos írási jog kapcsolója megmutatja, hogy a számára engedélyezett-e az adatok írása, vagyis módosítása. Ha ez be van kapcsolva, akkor a tulajdonos módosíthatja, bővítheti és törölheti az adott állományban található adatokat, sőt törölheti akár az egész állományt. Ha a kapcsoló ki van kapcsolva, akkor a tulajdonos nem módosíthatja az adatokat, még az állomány végére sem fűzhet újabb információkat. A könyvtáron értelmezett írási jog újabb fájlok létrehozására és törlésére ad lehetőséget. Ha írási joggal nem rendelkezik valaki az adott könyvtárban, esetleg még írhat a benne lévő állományokba vagy azokból adatokat törölhet. Egész állományokat nem hozhat létre és egész állományokat viszont nem törölhet. Amennyiben a könyvtárból az ott található állományokat nem tudja a felhasználó kitörölni, a könyvtárat magát sem tudja törölni, hiszen csak üres könyvtárakat lehet kitörölni. Ha a könyvtár már üres, a felhasználónak nincs szüksége írási jogra a könyvtárra nézve ahhoz, hogy azt kitörölhesse.
  - **x futtatási jog vagy végrehajtás**
    - A tulajdonos futtatási jogát jelző kapcsoló azt mutatja meg, hogy a tulajdonos elindíthatja-e az adott állományt. Ennek a lehetőségnek csak programok esetében van értelme. A könyvtáron értelmezett futtatási jog a könyvtár megnyitását jelenti, vagyis azt a képességet, hogy a könyvtárba belépünk. • fájl(ok) futtatásához és mappa megnyitáshoz is rx (olvasási és futtatási) jog szükséges
- fájl(ok) futtatásához és mappa megnyitáshoz is rx (olvasási és futtatási) jog szükséges

### A linuxos jogokról

olvasás	Read	r
írás	Write	w
végrehajtás	eXecute	x

A jogok háromféle felhasználónak adható:

- tulajdonos
- csoportba tartozó felhasználó
- mindenki más (aki az elz kettben nincs benne)

A `chmod` paranccsal állítgatjuk a jogokat, ahol néha hivatkozunk a tulajdonosokra, csoportokra, illetve mindenki másra, az alábbi táblázat alapján:

tulajdonos	User	u
csoport	Group	g
mások	Other	o
mindenki	All	a

Van egy negyedik hivatkozás, amely „mindenki” nevet viseli. Ez tartalmazza mind az elz hármat együtt.

A mindenki mind a három fajta felhasználót jelenti, amelyet például a `chmod` paranccsal használhatunk.

Mind a háromfajta felhasználónak állítható a olvasási, írási és végrehajtási jog. A következő táblázat amely az „ls -l” kimenetéből származhat jól szemlélteti ezt:

Az `ls -l` kiemelte:

```
-rw-r--r-- 1 joska joska 3184 dec 13 18.37 .bashrc
```

Táblázatban a jog rész:

típus	tulajdonos jogai	csoport jogai	mindenki más jogai
-	rw-	r--	r--

A típus a következ lehet:

Tartalom	Jelentés
-	Szimpla fájl
d	Könyvtár
D	Solaris kapu (Folyamatok közötti kommunikáció)
c	Karakteres eszköz (tty vagy nyomtató)
b	Blokkeszköz (rendszerint lemez vagy CD-ROM)
l	Szimbolikus link (BSD vagy SVR4)
s	Socket (BSD vagy SVR4)
= vagy p	FIFO (System V, Linux)

A Solaris kapu csak Sun Solaris rendszeren fordul el.

## chown

A chown (change owner, tulajdonosváltás) parancs segítségével megváltoztathatjuk az állomány tulajdonosát és csoporttulajdonosát. Alapesetben – ha csak a tulajdonost változtatjuk – a parancsnak a tulajdonos nevét és az állománynevet kell megadnunk.

```
chown joska.joska fajlnev
```

```
chown joska:joska fajlnev
```

A csoport felhasználónevét nem kötelez megismételni:

```
chown joska: fajlnev
```

vagy:

```
chown joska. fajlnev
```

## Kapcsolók

-R rekurzívan változtatja meg a (paraméterként megadott) könyvtár és annak tartalmának tulajdonosát

## Feladatok:

1.

## chmod

A chmod parancs a jogok beállítására használható.

Két módon adhatunk/vehetünk el jogokat. Az egyik számokkal, a másik betikkel.

Betikkel elsöör megadjuk kinek adunk vagy kitl veszünk el. Ez után +/- attól függen, hogy adunk vagy elveszünk. Majd végül mit adunk vagy mit veszünk el. A felhasználófajta halmazható, vagyis egymás után írható például: ug, vagy uo vagy ugo. Ez utóbbi persze egyenl azzal ha csak „a”-t

adok meg. A jogok szintén halmozhatók.

A jogokat az alábbi táblázat alapján rakhatjuk össze:

	Kinek/kitl	Mvelet	Mit
u	Saját(user)	+	r olvasás
g	csoport(group)	-	w írás
o	Saját(other)		x végrehajtás
a	Saját(all)		

Nézzünk néhány példát. Szeretnénk a felhasználónak írási jogot adni az erdo.txt fájlra.

```
chmod u+w erdo.txt
```

Most adjunk a csoportnak és a tulajdonosnak minden jogot:

```
chmod ug+rw erdo.txt
```

Itt a példában halmoztuk a felhasználók fajtáját és a jogokat is.

## A jogok megadása számokkal

A chmod paranccsal számokkal is megadhatjuk a jogokat:

0	-	-	-	nincs jogosultság
1	-	-	x	végrehajtási jog
2	-	w	-	írási jog
4	r	-	-	olvasási jog

A fenti táblázat alapján a jogok keverése mellett a következő jogok adhatók:

0	-	-	-	nincs jogosultság
1	-	-	x	végrehajtási jog
2	-	w	-	írás jog
3	-	w	x	írás, végrehajtás jog
4	r	-	-	olvasási jog
5	r	-	x	olvasás, végrehajtás jog
6	r	x	-	olvasás, írás jog
7	r	w	x	minden jog

Ha számokkal adjuk meg a jogokat, akkor tulajdonosnak, a csoportnak és mindenki másnak együtt kell megadni a jogokat. Megtehetjük, hogy csak egy számot adunk meg, ekkor a mindenki más jogait állítjuk be, és elveszük a tulajdonostól, illetve a csoporttól a jogokat. Az alábbi példában mindenki másnak megadjuk az olvasási, írási és futtatási jogokat.

```
chmod 7 erdo.txt
```

Futtatás után az ls -l erdo.txt kimenete:

```
-----rwx 1 mari info 28 márc  8 21.05 erdo.txt
```

Ebben az esetben erdo.txt állományt a mari nev felhasználó és az info csoport nem olvashatja, nem írhatja, nem futtathatja, mindenki más igen.

A következő példában a csoportnak olvasási és írási jogot adjunk, mindenki másnak olvasási, írási és végrehajtási jogot adunk.

```
chmod 67 erdo.txt
```

Az ls -l erdo.txt kimenete:

```
---rw-rwx 1 mari info 28 márc  8 21.05 a.txt
```

Adjunk mindenkinek olvasási és írási jogot:

```
chmod 666 erdo.txt
```

Most adjunk a csoportnak és a tulajdonosnak minden jogot, mindenki másról vegyük el:

```
chmod 770 erdo.txt
```

Itt a példában halmoztuk a felhasználók fajtáját és a jogokat is.

#### Feladatok:

- Mindenki (csoport) számára állítsd be az olvasási jogot, minden fájlra az aktuális könyvtárból és rekurzívan az alkönyvtárakban is!
- Adj futtatás jogot a fájl tulajdonos részére!
- Adj olvasás jogot a fájl tulajdonosnak és futtatás jogot a csoport részére!
- Adj futtatás jogot mindenki (a = anybody) részére file!

## Jogok

### Fájlok jogai

Tartalom	Jog	Jelentés
r	Olvasás	Olvasási jog: Megnyithat egy fájlt az open() rendszerhívással a read()-al pedig olvashatja.
w	Írás	Írási jog: Új fájlt hozhatsz létre, vagy egy meglévt módosíthatsz. Használhatod a write(), truncate( ) vagy ftruncate( ) rendszerhívást
x	Futtatás	Futtatási jog: Futtathatod a fájl útvonalának begépelésével, vagy az exec() rendszerhívással. Hogy hogyan fut a program az állomány els két byte függvénye. A #! például futtatható scriptre utal.












### Könyvtárak jogai

Tartalom	Jog	Jelentés
r	Olvasás	Használhatod az opendir( ) és az readdir( ) függvényeket (vagy az ls parancs) megnézni milyen fájlok vannak a könyvtárban.
w	Írás	Hozzáadhatsz, átnevezhetsz, mozgathatsz a könyvtárban

x	Futtatás	Megállapítható a könyvtár tartalma (Pl.: Megadható a fájlok tulajdonosa és hossza a könyvtárban). Futási jog szükséges könyvtárak létrehozásához az adott könyvtárban, vagy fájlok megnyitásához vagy az alkönyvtárakban
---	----------	--

**Példák** Az alábbi táblázatban az egymás mellett lévő parancsok ekvivalensek.

chmod u=rw,g=r,o=r file.txt	chmod 644 file.txt
chmod ugo=rw file.txt	chmod 666 file.txt
chmod a=rwx file.txt	chmod 777 file.txt
chmod ugo-rwx file.txt	chmod 000 file.txt

- Készítsen egy üres fájlt a home könyvtárában Test néven! 
- Állítsa be ennek jogait úgy, hogy a tulajdonos írhatta, olvashassa és futtathassa azt, a csoport olvashassa és futtathassa, mások pedig ne férjenek hozzá! 
- Adja át a fájl tulajdonjogát a rendszergazdának (miért nem működik?! 
- A fájl csoportja legyen a staff (miért nem működik?! 
- Állítsa át a fájl jogait ilyenre: rw-r--r--! Mit jelent ez? 
- Érje el, hogy a fájlt futtató felhasználók a futtatás idejére az ön jogaival rendelkezzenek! 
- Érje el, hogy a fájlt futtató felhasználók a futtatás idejére az ön csoportjának jogaival rendelkezzenek! 
- Készítsen egy test nev könyvtárt! Érje el, hogy ebben minden felhasználó hozhasson létre fájlokat és könyvtárakat, de csak a sajátjait törölhesse! 
- Készítsen egy Kozos nev könyvtárat, melyben minden felhasználó szabadon helyezhet el és törölhet fájlokat és könyvtárakat! 
- Kérdezze le a saját felhasználói azonosítóját és csoport tagsági információit! 
- Átmenetileg váltson át egy másik felhasználóra! (Használja a szomszédja login nevét és jelszavát!) 
- Tekintse át a /etc/passwd és a /etc/group fájlok tartalmát! Értelmezze az abban látható sorok jelentését! 