

Linuxos jogok

Bevezetés

A **DAC** szó a **Discretionary Access Control** szavakból alkotott betűszó, jelentése röviden „**Kizárólagos hozzáférés-vezérlés**”. Ez megengedi az azonosított felhasználóknak az objektumokhoz való hozzáférést azok tulajdonságai alapján. A hozzáférés csak a tulajdonosra, illetve csak a csoportra állítható be.

Ismert még a **MAC** vagy **Mandatory Access Control**, vagy **Rendelkező hozzáférés-vezérlés**. Az objektumok számára hozzáférési szabályokat hozunk létre, amely minden felhasználóra (még a rootra is) érvényesek.

A harmadik a **Role-based Access Control** vagy **RBAC. Szerep alapú hozzáférés-vezérlés**. A rendszergazda különböző szerepeket hoz létre. Az egyes objektumokhoz az adott szerepben lehet hozzáférni, az adott jogokkal.

Mi most a DAC alapú jogosultságrendszert tanulmányozzuk. Ezeket a unixos alapjogoknak is szokás nevezni.

A linuxos jogokról

Alapvetően háromféle jogot különböztetünk meg:

- olvasás
- írás
- végrehajtás

olvasás	Read	r
írás	Write	w
végrehajtás	eXecute	x

A jogok háromféle felhasználónak adhatók:

- tulajdonos
- csoportba tartozó felhasználó
- mindenki más (aki az előző kettőben 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 előző 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 következő 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 elő.

chown

A parancsot egy állomány vagy egy könyvtár tulajdonos illetve csoport beállítása használhatjuk.

```
chown joska.joska fajlnev
```

```
chown joska:joska fajlnev
```

A csoport felhasználónevét nem kötelező megismételni:

```
chown joska: fajlnev
```

vagy:

```
chown joska. fajlnev
```

chmod

A `chmod` parancs a jogok beállítására használható.

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

Betűkkel először megadjuk kinek adunk vagy kitől veszünk el. Ez után +/- attól függően, hogy adunk vagy elveszünk. Majd végül mit adunk vagy mit veszünk el. A felhasználófajta halmozható, vagyis egymás után írható például: `ug`, vagy `uo` vagy `ugo`. Ez utóbbi persze egyenlő azzal ha csak „a”-t adok meg. A jogok szintén halmozhatók.

A jogokat az alábbi táblázat alapján rakhatjuk össze:

	Kinek/kitől	Művelet	Mit
u	Saját(user)	+	r olvasás
g	csoport(group)	-	w írás
o	más(other)		x végrehajtás
a	mindenki(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+rwX erdo.txt
```

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

A jogok megadása számokkal

A jogok kifejezhető egy oktális számmal is. A chmod parancs ezeket képes használni. A következő táblázat megmutatja az egyes számok jelentését.

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 elvesszük 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 nevű felhasználó és az info csoport nem olvashatja, nem írhatja, nem futtathatja, mindenki más igen.

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

```
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 másból vegyük el:

```
chmod 770 erdo.txt
```

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

A `stat` paranccsal lekérdezhetők a számok:

```
stat -c '%A %a %n' *
```

```
stat -c '%A %a %n' adat.txt
```

A fájlok és a könyvtárak jogai

Fájlok jogai

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

Könyvtárak jogai

Tartalom	Jog	Jelentés
r	Olvasás	Használhatod az <code>opendir()</code> és az <code>readdir()</code> függvényeket (vagy az <code>ls</code> parancs) megnézni milyen fájlok vannak a könyvtárban.
w	Írás	Hozzáadhatsz, átnevezhetsz, mozgathatsz a könyvtárban Megállapítható a könyvtár tartalma (Pl.: Megadható a fájlok tulajdonosa és hossza a
x	Futtatás	könyvtárban). Futási jog szükséges könyvtárak létrehozásához az adott könyvtárban, vagy fájlok megnyitásához vagy az alkönyvtárakban

Az umask

Az umask határozza meg, hogy egy fájl vagy egy könyvtár milyen jogokkal jöjjön létre alapértelmezetten.

Az umask kétféle módon határozható meg:

- szimbolikus értékekkel
- oktális értékekkel

Az alapértelmezett érték 022. Az érték központilag felülírható az /etc/profile állományban is. A felhasználók ezt az értéket felülírhatják saját .profile vagy .bashrc állományukban.

Ha az alapértelmezett beállítást szeretnénk megváltoztatni, írjuk a fenti állomány egyikének végére:

```
umask 002
```

A beállított érték az újabb bejelentkezés esetén lesz érvényes.

Minden fájlnak van egy alapértelmezett létrehozási joga, ugyanígy a könyvtáraknak is:

fájlok 666

könyvtárak 777

Az umask segítségével ebből az alapértelmezésből veszünk még el.

Néhány lehetőség röviden:

- 022 – Lehetővé teszi fájlok egyéni írását, amiket mások olvashatnak.
- 077 – Mindenki csak a saját állományait látja.
- 002 – Fájlokat azonos csoportban lévők írhatják.
 - Csoporton kívül csak olvasási jog.
- 007 – aki nem csoport tag teljesen kizárva

Maszk számítás

A használható oktális értékeket a következő táblázat mutatja:

oktális szám	amit jelent
0	olvasás, írás, futtatás
1	olvasás, írás
2	olvasás, futtatás
3	olvasás
4	írás, futtatás
5	írás
6	futtatás
7	nincs jog

A fájl jogainak számítása

- fájl alapjogok: 666
- umask: 022
- számított jog: 644

A 644 kivonással jön létre:

$666 - 022 = 644$

A könyvtár jogainak számítása

- könyvtár alapjogok: 777
- umask: 022
- számított jog: 755

A 755 kivonással jön létre:

$777 - 022 = 755$

Példa

```
umask 077
mkdir dir1
touch file1
stat -c '%A %a %n' *
```

Lássuk a stat parancs kimenetét:

```
drwx----- 700 dir1
-rw----- 600 file1
```

Ugyanez az eredmény érhető el szimbolikus értékekkel:

```
umask u=rwx, g=, o=
```

A „chmod” parancs rövid ismertetése

A CHMOD parancs

Ideje kissé részletesebben beszélnünk a chmod parancsról. Ugyanis nagyon hasznos. Ez határozza meg, hogy ki mivel mit csinálhat.

Némileg pongyolán fogalmazva, mert minden részletbe nem merülök bele e könyvben, e parancs általános alakja:

```
chmod xyz állománynév
```

ahol az xyz egy oktális, azaz 8-as számrendszerbeli szám 3 számjegye, az állománynév meg sejtjük ugye, hogy mi... Az xyz ennek megfelelően bármi olyan 3 jegyű szám lehet, aminek egyes számjegyei a 0 és 7 közt mozognak. A legnagyobb helyiértékű számjegy azt mondja meg, hogy mit művelhetsz a fájlal te (általánosabban: a fájl tulajdonosa), a második azt, mit művelhet vele az, aki a tulajdonossal egy csoportban van, a harmadik, azaz legkisebb helyiértékű azt, mit művelhet a fájlal mindenki más, azaz akárki.

A lehetséges jogok a következők: olvasás, írás, végrehajtás (azaz futtatás). Minden oktális szám 3 bitből tevődik össze, a legnagyobb helyiértékű az olvasást, a középső az írást, a legkisebb

helyiértékű a végrehajtást jelenti. Ezt legkönnyebb úgy megjegyezni, hogy minden oktális számjegy felírható úgy, hogy $4+2+1$, aholis a 4, a 2 és az 1 helyén is lehet az összegben nulla, ha azonban nem nulla van ott akkor a 4 jelenti az olvasást, a 2 az írást, s az 1 a végrehajtást. Eképp például a `chmod 755 enyimfile`

azt jelenti, hogy te mint tulajdonos az `enyimfile` fájlt olvashatod, írhatod és futtathatod (mert $4+2+1=7$), a csoportod tagjai illetve bárki más viszont csak olvashatja és futtathatja (de nem írhatja át), mert $4+1=5$).

Gyakoribb számkombinációk és jelentéseik:

777 Bárki bármit tehet a fájjal. Biztonsági szempontból nem javallott.

755 A tulajdonos bármit tehet a fájjal, mások csak olvashatják és futtathatják. Tipikus alkalmazási területe ha te írsz egy mások által is használható szkriptet.

700 Csak a tulajdonos bírzigálhatja a fájlt, ő mindent megtehet vele, mások azonban semmit.

666 Ez nem a Sátán száma: azt jelenti, a fájlt bárki írhatja és olvashatja, viszont senki által nem végrehajtható (mert nem progi és nem szkript). Ideiglenes fájlok esetén előfordulhat, hogy jól jön ez a beállítás.

644 A tulajdonos írhatja-olvashatja a fájlt, mások csak olvashatják. Nem végrehajtható fájlok esetén használatos.

600 A tulajdonos írhatja és olvashatja a fájlt, mások semmit nem csinálhatnak vele.

A `chmod` parancs értelmezett nemcsak közönséges fájlokra, de tartalomjegyzékekre (mappákra...) is, a következőképp:

az „olvasási” jog azt jelenti, hogy láthatjuk a tartalomjegyzékben szereplő bejegyzéseket (fájlokat illetve alkönyvtárakat). (Ez azt jelenti, hogy például az `ls` paranccsal kilistázhatod a tartalomjegyzéket. Nem jelenti viszont azt, hogy be is léphetsz a könyvtárba!)

Az „írási” jog azt jelenti, hogy létre tudunk hozni benne bejegyzéseket, illetve efféle bejegyzéseket törölhetünk benne. Mindezt természetesen a szokásos parancsokkal, például a `cp`, `ln`, `mkdir`, `mv`, `rm`, `rmdir` parancsokkal.

A „végrehajtási” jog azt jelenti, hogy például beléphetsz a könyvtárba, és indíthatod innen a végrehajtható fájlokat.

Miután az olvasási és végrehajtási jog nehezen érthető még ezek után is könyvtárakra, íme egy kis táblázat, a különböző variációkra. A „+” azt jelenti, sikerülni fog a megkísérelt művelet, a „×” azt jelenti, nem fog sikerülni:

Mit akarsz csinálni.....	milyen jogaid vannak a könyvtáron
.....	csak olvasási.....csak végrehajtási
Belépni a könyvtárba	×+
Kilistázni a könyvtár tartalmát.....	+.....×
Kimásolni onnan egy fájlt.....	×+
Futtatni onnan egy fájlt	×+

És még egy fontos dolog: a jogosultságok rekurzív megváltoztatása. Ha mondjuk az „Innen” nevű alkönyvtár összes (akár alkönyvtárakban levő...) fájljának jogosultságait akarod megváltoztatni mondjuk 755-re, így használd a parancsot:

`chmod -R 755 Innen`

Referencia kézikönyv:

```
chmod
# Linuxon a fájlokra, könyvtárakra vonatkozó jogok állíthatóak be ezen
parancssal
-R : Rekurzívan változtatja meg a jogosultságokat.

# DAC (háromszintű diszkrécionális maszk) szerinti beállítás.
# r-read (olvasás), w-write (írás), x-executable (futtatás) jogot jelent
#
# Általános jogosultság lista felépítése: (ls -la parancssal
lekérdezhető)
# tulajdonos (jele:U) | csoport felhasználó (jele:G) | mindenki más
(jele:O)
#          rwx          |          rwx          |          rwx
#
# A sor elején található "d" a directory, "-" a file jele.
# Jogok nem csak szimbólikus jelekkel de számokkal is meghatározhatóak.
#
# Számokkal.: 4-read, 2-write, 1-executable jog, összeadva, külön U,G,O-
nak
#

chmod 777 file : UGO-nak egyaránt minden jog. (4+2+1 4+2+1 4+2+1)
chmod 751 file : U-nak minden, G-nek írási és futtatási, O-nak futtatási
jog.

# Betűkkel.: kinek+mit
#

chmod u+rwx file : Tulajdonosnak (U) r,w,x jog adása az adott file-ra.
chmod g+rx file  : Csoport felhasználónak (G) r,x jog beáll.
chmod a-rwx      : Mindenkitől (A-all) elveszünk minden jogot.

umask
# A file és könyvtár jogok beállítása kapcsán érdemes megemlíteni az
umask-ot.
# Az umask meghatározza, hogy milyen jogosultságot kapjanak az újonnan
létrehozott file-ok, mappák.
# Értéke alapértelmezés szerint 022.
# Jelentése.: File-ok 644-et, Mappák 755 jogokat kapnak.
# File-ok esetén 666-ból,
# Mappák esetében pedig 777-ből kell levonni a 022-t, így kapjuk meg a
jogokat.

chmod
# Linuxon a fájlokra, könyvtárakra vonatkozó jogok állíthatóak be ezen
parancssal
-R : Rekurzívan változtatja meg a jogosultságokat.

# DAC (háromszintű diszkrécionális maszk) szerinti beállítás.
# r-read (olvasás), w-write (írás), x-executable (futtatás) jogot jelent
#
# Általános jogosultság lista felépítése: (ls -la parancssal
lekérdezhető)
# tulajdonos (jele:U) | csoport felhasználó (jele:G) | mindenki más
(jele:O)
#          rwx          |          rwx          |          rwx
#
```



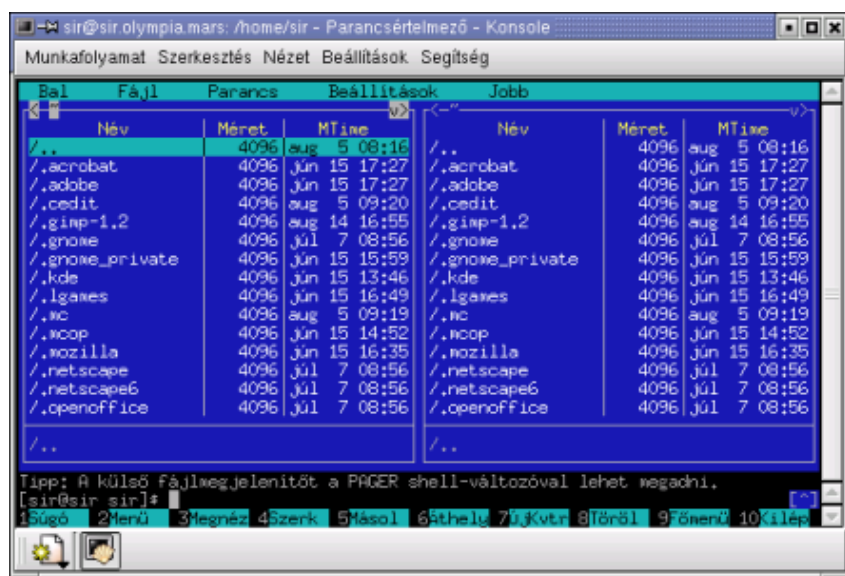
```
# A sor elején található "d" a directory, "-" a file jele.
# Jogok nem csak szimbólikus jelekkel de számokkal is meghatározhatóak.
#
# Számokkal.: 4-read, 2-write, 1-executable jog, összeadva, külön U,G,O-
nak
#
chmod 777 file : UGO-nak egyaránt minden jog. (4+2+1 4+2+1 4+2+1)
chmod 751 file : U-nak minden, G-nek írási és futtatási, O-nak futtatási
jog.

# Betűkkel.: kinek+mit
#
chmod u+rwx file : Tulajdonosnak (U) r,w,x jog adása az adott file-ra.
chmod g+rx file : Csoport felhasználónak (G) r,x jog beáll.
chmod a-rwx      : Mindenkitől (A=all) elveszünk minden jogot.

umask
# A file és könyvtár jogok beállítása kapcsán érdemes megemlíteni az
umask-ot.
# Az umask meghatározza, hogy milyen jogosultságot kapjanak az újonnan
létrehozott file-ok, mappák.
# Értéke alapértelmezés szerint 022.
# Jelentése.: File-ok 644-et, Mappák 755 jogokat kapnak.
# File-ok esetén 666-ból,
# Mappák esetében pedig 777-ből kell levonni a 022-t, így kapjuk meg a
jogokat.
```

A Midnight Commander

A Midnight Commander egy karakteres felületen használható program, amelynek segítségével állomány- és könyvtárkezelő műveleteket lehet végezni. Ezenkívül alkalmas programok indítására, a beépített szövegszerkesztője révén szövegszerkesztésre, valamint FTP-kliensként is használható.



Ahogy a képen látható, a program – amely egyszerűen az `mc` parancs kiadásával indítható – két oszlopban képes egy-egy könyvtár tartalmának megmutatására. A két oldal között a Tab billentyűvel (néhány billentyűzeten csak két ellentétes irányú, vízszintes nyíl látható rajta) lehet váltani. Az aktív oldalt többek között abból lehet látni, hogy az egyik könyvtárbejegyzés sora zöld háttérben jelenik

meg – feltéve, hogy az eredeti, a képen is látható színösszeállítás van érvényben: a szóban forgó jelölést nevezzük *inverz megjelenésnek*.

Ez a sor jelzi, hogy éppen hol járunk az adott könyvtárban. Mindkét oszlop tetején, a menü alatt megjelenik az adott könyvtár elérési útvonala. Mivel a képen ez a home-könyvtár, így ennek speciális jele, a ~ jelenik meg.

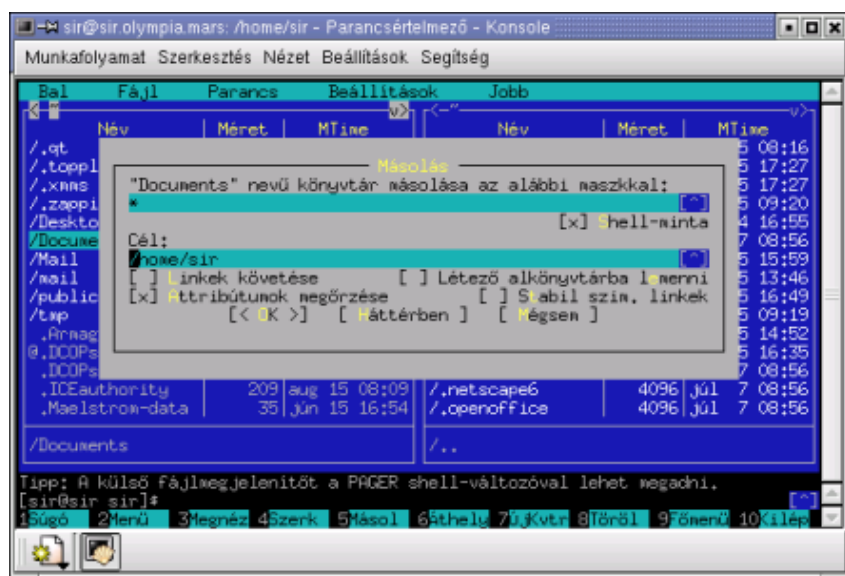
Legalul látható egy parancssor, ahol a szokásos parancsok bármelyikét kiadhatjuk, és az a szokásos módon végrehajtódik. Amennyiben a teljes parancsoros képernyőt látni szeretnénk – mert például egy parancs válasz üzeneteit szeretnénk látni, akkor a Ctrl+ O billentyűvel ezt megtehetjük, és vissza is ugyanígy juthatunk.

Az egyes könyvtárbejegyzésekről a képen látható helyzetben a nevéen kívül még a méretét és a legutóbbi módosítás időpontját lehet látni. Ez azonban csak az egyik lehetséges nézet. Ha a képernyő (vagy ablak) legalsó sorában látható billentyű-menü segítségével, vagy az egerrel a felső menübe jutunk, akkor a jobboldali oszlop beállítását a *Jobb*, a baloldali értelemszerűen a *Bal* menüben módosíthatjuk, az oszlop kinézetét a *Listázási nézet* nevű menüpont hatására előhívható párbeszédablakban.

A kérdés már csupán az, hogy az alsó billentyű-menüt hogyan kell értelmezni. Az egyes funkciók megnevezése előtt található szám egy funkcióbillentyűre utal. Eszerint a kiválasztott állomány tartalmának megtekintése például az F3 billentyűvel (megnéz) lehetséges. Ha egy könyvtáron állunk, akkor ez a funkció az adott könyvtárba történő átlépést jelenti. Eszerint a felső Főmenübe az F9 billentyűvel juthatunk, ahol a kurzorgombokkal lehet közlekedni.

Egy oszlopon belül a kurzor le és kurzor fel gombokkal lehet közlekedni. A kiválasztott könyvtárbejegyzéshez tartozó alapértelmezett cselekvést az Enter billentyűvel lehet elérni. Ez könyvtárnál a belépés, futtatható állománynál a futtatás, tömörített állománynál annak tartalmát mint könyvtárt jeleníti meg. A nem alapértelmezett műveleteket, amelyek az adott állománnyal vagy könyvtárral végezhetők, a F2 billentyűvel elérhető menüből lehet kiválasztani.

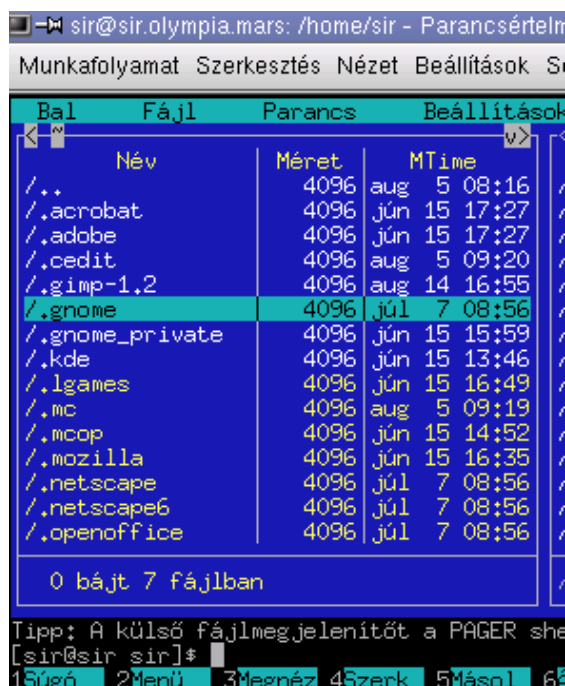
Az alsó billentyű-menün más műveletek is elérhetők, nemcsak az éppen láthatók: ha valamelyik funkciómódosító billentyűt lenyomva tartjuk (Alt, Ctrl), és a program ennek megfelelően van beállítva, akkor az azzal a billentyűvel elérhető szolgáltatások jelennek meg.



Miért jó, hogy két könyvtár látható egyszerre? Azért, mert a másolás és mozgatás műveletek alapértelmezett célkönyvtára a nem aktív oldalon látható könyvtár. Igaz ugyan, hogy a műveletek kezdeményezésekor megjelenő párbeszédablakban ez felülbírálnak, de ha előbb megkeressük a célkönyvtárt az egyik oldalon, majd kiválasztjuk a másik oldalon, akkor a művelet gyorsabban és kényelmesebben elvégezhető.

Természetesen még kényelmesebbé tehető a másolás, ha egyszerre több bejegyzést is lehet másolni. Ennek semmi akadálya, mivel a bejegyzéseket ki lehet jelölni. Ennek két módja van:

1. Egérrel: a kijelölendő vagy a kijelölésből eltávolítandó bejegyzésre az egér jobb gombjával kell kattintani.
2. Billentyűzetről: álljunk rá az inverz sorral, majd nyomjuk meg az Insert gombot. Ekkor a kiválasztó sor egy bejegyzéssel lejjebb kerül (ha van ilyen), és a kijelölt sor más színű lesz.



A következő képkönn a sárga színű bejegyzések vannak kijelölve.

A továbbiakban – a teljesség igénye nélkül – a Midnight Commander legfontosabb funkciót említjük meg. Aki ezekről, valamint a nem ismertett funkciókról többet szeretne tudni, az olvassa el az F1 billentyűvel elérhető *Help*et, vagy a `man mc` paranccsal elérhető manuál oldalt.

Másolás, mozgatás

Ahogy már szó volt róla, ezek a műveletek a nem aktív oldalon látható könyvtárt tekintik alapértelmezett célkönyvtárnak, de ez módosítható minden alkalommal. A mozgatás helyett átnevezés érhető el, ha a célkönyvtár megadását egy állománynévre módosítjuk. Az F5 (másolás) és az F6 (mozgatás/átnevezés) billentyűk tartoznak hozzá.

Törlés

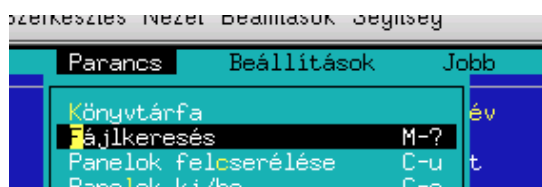
A kiválasztott/kijelölt bejegyzések törlése (végérvényesen!) az F8 billentyűvel lehetséges. Amennyiben könyvtárt akarunk törölni, akkor a program a biztonság kedvéért külön rákérdez, hogy

a nem üres könyvtárt tényleg törölni akarjuk-e. Egyébként a művelet végrehajtása előtt – hacsak valaki ki nem kapcsolja – mindig megerősítést kér a műveletről annak veszélyessége miatt.

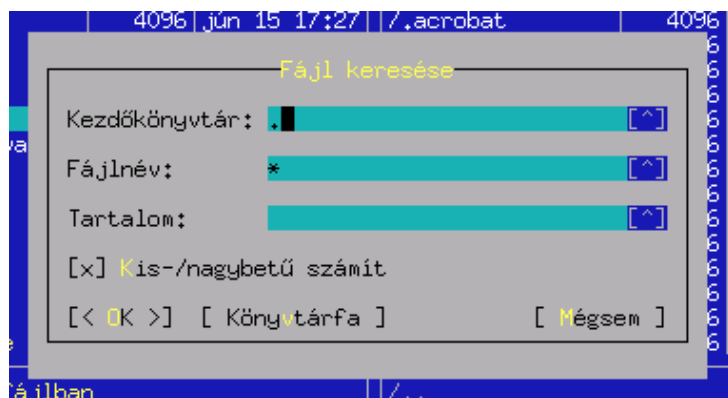
Új könyvtár létrehozása

Az aktív oldalon látható könyvtárba egy új könyvtárt lehet létrehozni az F7 billentyű segítségével. Egy párbeszédablakban a program megkérdezi a létrehozandó könyvtár nevét, és az Enter lenyomására az létre is jön.

Állomány keresése



A Főmenü *Parancs* menüjén belül érhető el a *Filekeresés* funkció. Kiválasztásakor egy párbeszédablakot kapunk, amelyben nem csak azt lehet megadni, hogy milyen nevű állományt keresünk, és hogy hol keressük, hanem egy részletet is az állomány tartalmából.



Ez utóbbit azonban csak akkor van értelme megadni, ha szöveges állományt keresünk, különben a program nem biztos, hogy megtalálja azt a szövegrészt. Ráadásul ha tartalomra is keresünk, azzal jelentősen megnövelhetjük a művelet végrehajtási idejét, hiszen a programnak végig kell olvasnia azokat az állományokat, amelyek a megadott névvel egyeznek.

Név megadásakor a * és ? karakterek használhatók a név maszkolására. A kezdő könyvtárnak a szerepe abban van, hogy az ott megadott könyvtárt és alkönyvtárait vizsgálja végig a program. Ha itt a . látható (aktuális könyvtár), akkor abban a könyvtárban és alkönyvtáraiban történik a keresés, amely a parancs kiadásakor aktív oldalon volt látható. Ha a / szerepel itt, akkor a teljes rendszert végigvizsgálja a program. Ez kerülendő, mivel rendkívül sokáig tarthat, és ha több gép össze van kapcsolva, akkor rettenetesen lefoglalhatja a rendszert, mert a más gépeken levő, de felcsatolt állományrendszereket is végigvizsgálja, ami a hálózatot is terheli!

FTP-kliens

A Midnight Commander nagyon egyszerűen használható FTP-re is. Virtuálisan fel lehet csatolni alkönyvtárként egy FTP-szerveret a rendszerbe valamelyik oldalra. A feljelentkezés történhet anonymous-ként, vagy jelszavas felhasználóként is. Mindkét esetben az adott oldal menüjében az *FTP-kapcsolat* menüpontot kell választani. A megjelenő párbeszédablakban kell megadni az

elérendő gép nevét (az azonosítót nem anonim-us-kapcsolat esetén egy @ jellel elválasztva a gép neve elé kell írni). A kapcsolat létrejöttékor megjelenik a szerver tartalma, mint helyi könyvtár, és ennek megfelelően is kezelhető – természetesen a megfelelő jogosultságok figyelembe vételével.

A kapcsolat bontása nagyon egyszerű: az FTP-szerver főkönyvtárában a legfelső bejegyzés, a . . kivezet a szerverről, és saját gépünk főkönyvtárába visz. Bármely más módon az adott oldalon visszatérve a saját gépünk valamelyik könyvtárába, a kapcsolat szintén automatikusan felbomlik.

Kicsomagolás

Tömörített, vagy tömörítetlen [tar](#) archívumot, és néhány egyéb archívumot – amelyek kitömörítéséhez van megfelelő, elérhető program a rendszeren – a Midnight Commanderrel nagyon egyszerűen ki lehet csomagolni: Mintha könyvtár lenne, lépünk bele az archívumba. Válasszuk ki a kicsomagolandó állományokat illetve könyvtárakat, majd a másolás F5 gombjával a szokásos módon kimásolhatjuk a másik oldali könyvtárba. Mozgatás, törlés, és egyéb, az archívumot módosító művelet általában nem végezhető.