

## A Linux NTFS fájlrendszerrel kapcsolatos eszközei

Elképzeltető, hogy a legjobb Windows segédprogramok közül néhány már ott lapul a Linux partíciónkon...

**L**inux alól több tucat különféle fájlrendszerhez férhetünk hozzá, legyenek azok akár a helyi gépünkön, akár egy hálózati kiszolgálón. Hogy rögtön a lényegre térjék, *Linux* alatt számos olyan eszköz létezik, amelyekkel *Windows* fájlrendszereken végezhetünk műveleteket, vagy akár meg is javíthatjuk azokat, kiegészítve ezzel a bajból a *Windows*-t.

Ezeknek a segédeszközöknek egy jelentős része a *Linux-NTFS* projekt terméke. Vannak köztük olyanok, amelyek képességei már-már csodálatosnak nevezhetők. Van például olyan, amelyikkel átméretezhetünk egy *NTFS* partíciót. Másokkal egyedi fájlokat kezelhetünk, míg megint másokkal klónozhatunk egy *NTFS*-re formázott lemezrészét. *Linux* alól megoldható például teljes *Windows* telepítések biztonsági mentése, de készíthetünk olyan lemezképeket is, amelyekről hálózaton keresztül klónozhatók a windowsos munkaállomások. És mivel valamennyi ilyen eszköz *Linux* alatt fut, öröklí mindazt a hatékonyságot, amivel a *UNIX* rendszer csak rendelkezhet. Használhatjuk ezt az eszközkészletet akár egyetlen olyan gépen is, amelyre két operációs rendszer van telepítve, de igazán hasznosnak azok érezhetik, akik egy nagyobb hálózatot tartanak karban. Mivel *Linux* alatt van átirányítás, csövek, és persze írhatunk héjprogramokat is, könnyedén automatizálhatunk számos olyan adminisztratív feladatot, amelyek végrehajtása *Windows* alatt igen fáradságos és persze unalmas lenne.

### Telepítés

Az említett segédprogramok számos helyen megtalálhatók, és a támogatásuk is egészen kiváló. Gyakorlatilag valamennyi olyan *Linux* terjesztésre létezik belőlük megfelelő csomag, amely tartalmaz valamilyen csomagkezelőt, sőt, még a *Knoppix CD*-n is megtalálhatók. Ugyanakkor a legtöbb terjesztés telepítője alapértelmezésként úgy állítja be a jogosultságokat, hogy ezeket az eszközöket csak a rendszergazda legyen képes futtatni. Aki nem tudja, hogy a kérdéses csomag fönt van-e már a gépén, próbaképpen adja ki a `man ntfsprogs` parancsot. Ebből egyértelműen kiderül, hogy legalább a dokumentációja föltelepült-e.

Még ha a próba nem is járt sikerrel, a fentiek miatt minden bizonnyal egészen egyszerű lesz a megfelelő csomag telepítése. *SUSE*, *Debian*, *Ubuntu* és *Gentoo* alatt a csomag neve a korábban említett *ntfsprogs*. Erre kell rákeresnünk a telepítési adathordozón, és föltelepíteni, ha megtaláltuk. Egyes rendszereken a csomagban benne van az összes, az *NTFS* fájlrendszerek kezelésére használható eszköz, míg másokon bizonyos dolgok hiányoznak. A *Debian Etch* például tartalmazza az `ntfsmount` eszközt, míg a *Sarge*-ben található csomag nem. A *Red Hat/Fedora* terjesztés licenzelési problémák miatt egyáltalán nem tartalmazza magát a csomagot sem, viszont az interneten számos helyen megtalálhatjuk annak kifejezetten ezekhez a rendszerekhez idomított változatát, olyannyira, hogy az még a *Linux-NTFS* projekt hivatalos webhelyéről is

elérhető. Aki a legfrissebb dokumentációt vagy forráskódot szeretné letölteni, annak természetesen szintén a hivatalos weblapot (☞ <http://www.linux-ntfs.org>) tudjuk ajánlani. A kapcsolódó dokumentáció minden olyan követelményt és megoldást is tartalmaz, ami a programok forrásból való lefordításához szükséges lehet. Utóbbi módszer természetesen a *Linux* bármely változatán működik, sőt, ha valaki ki szeretné próbálni a legújabb szolgáltatásokat is, annak feltehetőleg nem is lesz más választása, mint hogy forráskódból kezdjen el dolgozni. Egyes rendszereken persze gondot okozhat, hogy ilyenkor óhatatlanul kikerüljünk a csomagkezelő adminisztrációs rendszerét. Egy fontos megjegyzés mindazoknak, akik a fenti figyelmeztetés ellenére mégis a forráskód mellett döntöttek: a csomag lefordításához szükség van a *FUSE* könyvtárra, aminek a forrását a ☞ [fuse.sourceforge.net](http://fuse.sourceforge.net) webhelyről szerezhetjük be. A *Linux*nak van ugyan egy beépített *NTFS* meghajtója, az *NTFS* segédeszközök azonban egy másikra támaszkodnak, ami szintén benne van a csomagban. A „nem natív” meghajtó az, ami a *FUSE* könyvtárra támaszkodik, és amely számos extra szolgáltatást nyújt a „kincstári” `ntfsmount`-hoz képest. Igaz ugyan, hogy valamivel lassabban működik, mint a kernel részét képező *NTFS* modul, de ez talán nem olyan nagy probléma. Ugyanakkor fontos megjegyezni, hogy szükségünk lesz a *FUSE* kernelmodul betöltésére is. Aki tehát telepíteni szeretné a *FUSE* könyvtárat, az először is töltsse le

a legfrissebb forráskódot egy megfelelő könyvtárba, amely akár azzal is azonos lehet, amely az *ntfsprogs* csomagot tartalmazza. Ami a fordítás folyamatát illeti, abban nincs semmi különleges. Egyszerűen csak ki kell adnunk a `configure`, `make`, `make install` parancsokat, utóbbit persze root-ként bejelentkezve (a módszer amúgy mára gyakorlatilag szabványosnak is nevezhető, a verziószám pedig természetesen változhat, hiszen a rendszert folyamatosan fejlesztik):

```
tar -xvzf fuse-2.5.3.tar.gz
cd fuse-2.5.3.tar.gz
./configure
make
make install
```

A *FUSE* könyvtár és a kernelmodul telepítése amúgy nem föltétlen szükséges abban az esetben, ha csupán olvasási jogot szeretnénk az *NTFS* partíciókhoz. Ennek az az oka, hogy – amint az korábban is említettem – a *Linux* kernel eleve tartalmaz egy az *NTFS* fájlrendszerek olvasására való modult, ami együttműködik a közönséges `mount` paranccsal, ráadásul még gyorsabb is. Ugyanakkor nem rendelkezik mindazokkal a kiegészítő szolgáltatásokkal, amelyek az `ntfsmount` igen.

A következő lépés az *ntfsprogs* forráskódjának letöltése. Ezt is helyezzük el egy megfelelő könyvtárban, majd – rootként bejelentkezve – fordítsuk ugyanazzal a módszerrel, amivel az imént a *FUSE* csomagot (a verziószám természetesen ismét eltérhet az itt megadottól):

```
tar -xvzf ntfsprogs-
↳ 1.13.1.tar.gz
cd ntfsprogs-1.13.1
./configure
make
make install
```

Ha rendszerünk nem tartalmazza a *FUSE* könyvtárat (még ha a *FUSE* modul fut is) a `configure` parancs futtatása közben a következő figyelmeztetést fogjuk kapni:

```
checking for FUSE_MODULE...
↳ configure: WARNING: \
ntfsmount requires FUSE
↳ version >= 2.3.0
```

Ez persze nem végzetes, az *NTFS*-t kezelő eszközöket ettől még ugyanúgy lefordíthatjuk, az `ntfsmount` programot azonban nem. Aki *Red Hat/Fedora* rendszert használ, annak a gépén még a megfelelő kernelmodul sem lesz fönn. Ilyenkor két választásunk van: vagy átállunk egy olyan speciális kernelre, amely tartalmazza az *NTFS* modult, vagy mi magunk lefordítjuk a *FUSE* könyvtárat a bemutatott módon.

### Lássuk a programokat

Most tehát ott tartunk, hogy az Olvasó vagy sikeresen föltelepített gépére az *ntfsprogs* csomagot a fent bemutatott módszerek valamelyikével, vagy egyszerűen csak megtalálta azt a az lapértelmezésként telepített csomagok listájában.

Az *ntfsprogs* súgóoldala tartalmazza a rendelkezésünkre álló segédprogramok teljes listáját. Lássuk közülük most csak a legfontosabbakat:

- `mkntfs(8)` – Egy partíció megformázása *NTFS* fájlrendszerrel.
- `ntfscat(8)` – Egy *NTFS* fájlrendszeren található fájl tartalmának kiküldése a szabványos kimenetre.
- `ntfsc1one(8)` – Egy *NTFS* partíció tartalmának hatékony klónozása, létrehozása, visszaállítása vagy mentése lemezképe/ből.
- `ntfsc1uster(8)` – Felderíti az *NTFS* partíció egy adott szektorának vagy klaszterének tulajdonosát.
- `ntfscp(8)` – Egy *NTFS* partícióon található fájl felülírása.
- `ntfsfix(8)` – Egyes gyakori hibák ellenőrzése és javítása, a *LogFile* törlése, illetve a *Windows* utasítása arra, hogy a következő bootolásnál végezzen átfogó ellenőrzést az adott fájlrendszeren.
- `ntfsinfo(8)` – Információkat jelenít meg egy teljes *NTFS* partícióról, vagy egy azon található könyvtárról vagy fájlról.
- `ntfslabel(8)` – Az *NTFS* partíció címkéjének megjelenítése, vagy beállítása.
- `ntfs1s(8)` – Egy *NTFS* lemezrészén található könyvtár tartalmának listázása.
- `ntfsmount(8)` – *FUSE* *NTFS* modul.

- `ntfsresize(8)` – *NTFS* partíció átméretezése adatvesztés nélkül.
- `ntfsundelete(8)` – Törölt fájlok visszaállítása egy *NTFS* fájlrendszeren.

A listában szereplő segédprogramok közül némelyiknek kétségtelenül olyanok a funkciói, hogy azok inkább a fejlesztőknek, semmint a hálózati rendszergazdáknak, vagy a saját, két operációs rendszert tartalmazó gépüket karbantartó otthoni felhasználóknak lehetnek hasznosak. Ugyanakkor sok olyan is van, ami valóságos életmentő. Ez utóbbi kategória talán legfontosabb képviselője az `ntfsc1one`.

### Az *NTFS* eszközök használata

Ahhoz, hogy kipróbálhassuk az `ntfsc1one` működését, először is szükségünk lesz egy *NTFS* fájlrendszerre, meg persze egy másikra, ami elég nagy ahhoz, hogy elférjen rajta az előbbi lenyomata. Utóbbira első közelítésben egy *ext2*, *ext3*, *xf*s vagy *ReiserFS* fájlrendszert javasolnék. Az `ntfsc1one` dokumentációja ugyan fölhívja a figyelmet arra, hogy a *ReiserFS* kifejezetten lassú tud lenni, ha szétszórtan elhelyezkedő fájlokat kell kezelni, én azonban ezt nem igazán tapasztaltam az újabb változatokkal kapcsolatban. Használhatunk természetesen külső meghajtót is, ha van rajta elég hely. Arra azonban ügyeljünk, hogy bizonyos műveletek végrehajtásához akkora szabad hely szükséges, mint amekkora az egész *Windows* partíciónk. Amennyiben a külső meghajtó *FAT32* fájlrendszert tartalmaz, szintén kellemetlen meglepetésben lehet részünk, hiszen itt van egy meglehetősen szűk felső korlátja a legnagyobb tárolható fájlnak. A *FAT32* általánosságban tehát nem igazán alkalmas erre a célra. Végezetül – bár ez meglehetősen triviális – a gépünkön *Linux*nak kell futnia, tehát na nincs ilyen telepített rendszerünk, akkor használjunk valamilyen Live terjesztést, például egy *Knoppix CD*-t.

Az `ntfsc1one` fenti leírásában azt állítják, hogy a program hatékonyan teszi a dolgát, és ez nem csak valami hangzatos reklámszöveg. Olyannyira nem, hogy az újabb gépeken egy *Windows XP*-t futtató munkaállomás teljes

klónozása nem tart tovább pár perc-nél. Ha a klónozni kívánt *NTFS* fájlrendszer az első *IDE* merevlemez sorrendben első partícióján található, a mentéshez használt *Linux* pedig ugyanazon a gépen fut, akkor a következő parancs egyetlen fájlba „klónozza le” a teljes *Windows XP*-t:

```
ntfsclone /dev/hda1 -o /usr/
↳ local/backup/ntfs.img
```

A parancsban szereplő `-o` kapcsoló arra utasítja a programot, hogy írja felül a képfájlt, de ha az eredetileg nem létezik, természetesen automatikusan létre is jön. Ha ilyen módon futtatjuk a programot, akkor az a képfájlt nem tömöríti, olyannyira nem, hogy azt később akár be is csatolhatjuk a visszacsatoló (*loopback*) eszközön keresztül. Ez utóbbihoz először is hozzuk létre magát a csatolási pontot:

```
mkdir /usr/local/backup/mtpt
```

Ha ez megvan, akkor az *ntfsmount* segítségével pontosan ugyanazt a szintaxist használva fűzhetjük be a virtuális *NTFS* fájlrendszerünket, mint ahogy azt bármely más képfájllal tennénk:

```
ntfsmount -o loop /usr/local/
↳ backup/ntfs.img \
  /usr/local/backup/mtpt/
```

Az *ntfsmount* alapértelmezésként írható és olvasható állapotban fűzi be a megadott fájlrendszert, ami azt jelenti, hogy könnyedén másolhatjuk vagy törölhetjük is a rajta levő fájlokat. Persze ezen a ponton meg kell majd küzdenünk mindazokkal az „apró” problémákkal, amelyek mindig előjönnek, valahányszor különböző operációs rendszerek között próbálunk mozogni. Ha például konfigurációs fájlokat kezdünk szerkeszteni, mindenképpen ügyeljünk rá, hogy az adott rendszeren pontosan milyen karakterkombináció is jelzi a sor végét. Mellesleg pedig meggyűlhet a bajunk az eltérő karakterkódolásokkal is, de ez manapság szinte természetes. Ha a becsatoláshoz a natív *NTFS* meghajtót akarjuk használni, természetesen a hagyományos *mount* parancsot kell futtatnunk, de a szintaxis gyakorlatilag azonos:

```
mount -o rw,loop,nls=utf8 -t
↳ ntfs \
  /usr/local/backup/ntfs.img \
  /usr/local/backup/mtpt/
```

Figyeljük meg, hogy itt már a becsatoláskor megadtuk a *Windows* számára természetes karakterkódolást is.

Persze ez még nem véd meg bennünket mindentől. Itt is kell némi szakértelem, bölcs tapasztalat, meg némi éberség, ha teszem azt elkezdjük

*Emacs*-ból átszerkeszteni a *boot.ini* fájlt. Az az igazság, hogy éberség jobbra, gyakorlat balra, az ilyesmit még mindig biztonságosabb *Windows* alól elvégezni, esetleg a *Notepad* segítségével *Wine* alatt.

Ha írási és olvasási jogra is szükségünk van az *NTFS* fájlrendszeren, akkor a becsatolásra másodikként bemutatott módszerrel vagy szerencsénk lesz, vagy nem, attól függően, hogy pontosan melyik kernelverziót használjuk a gépünkön. Ezzel kapcsolatban csak annyit tudok mondani, hogy a natív *NTFS* meghajtó kezelése és egész működése némiképp trükkös. Ha valami nem tetszik neki, általában küld egy figyelmeztetést, és ezzel párhuzamosan visszakapcsol csak olvasható módba. Ami pedig a régebbi változatait illeti, nos azok némelyike egyenesen életveszélyes, ha engedélyezzük neki az írást is.

A fájlrendszer lecsatolását mindkét esetben ugyanúgy végezhetjük:

```
umount mtpt/
```

A most létrehozott *ntfs.img* nevű fájl ugyanúgy mozgatható és másolható, mint bármely más – bár kisése talán túlméretes – fájl. Ha szükséges tömöríthetjük, tárolhatjuk biztonsági másolatként, vagy feltölthetjük egy távoli gépre. A másolatot a bemutatott módszerre akár át is szerkeszthetjük, és szükség esetén felülírhatjuk vele az eredeti fájlrendszert. A lementett képfájl visszairása – feltéve, hogy abban a könyvtárban vagyunk, amelyben található – a következő paranccsal történhet:

```
ntfsclone ntfs.img -o
↳ /dev/hda1
```

Néha a kisebb a jobb, így aztán az *ntfsclone* parancsnak is vannak

olyan kapcsolói, amelyekkel hatékonyan tömöríthetjük a képfájlt. Talán meglepő, de ezekkel a jelzőkkel a klónozás még gyorsabb is lesz, akár a helyi gépen végezzük, akár hálózatról. Íme egy példa, amikor a képfájlt majdnem pontosan ugyanúgy mentjük, mint az imént:

```
ntfsclone --save-image /dev/
↳ hda1 -o \
  /usr/local/backup/ntfs.img
```

Persze azért semmi sem lehet tökéletes: ezt a képfájlt nem csatolhatjuk be a bemutatott módon, hacsak vissza nem írjuk egy merevlemezre, vagy egy másik, tömörítetlen fájlba. Az eredeti partícióra való visszaállítás a következőképpen fest:

```
ntfsclone --restore-image
↳ --overwrite /dev/hda1 \
  /usr/local/backup/ntfs.img
```

Figyeljük meg, hogy ebben a példában a nem túl beszédes `-o` kapcsolót a szkriptekben inkább szokásos `--overwrite` formára cseréltük. Persze a hatás pontosan ugyanaz lesz, csak a forma más. Általában is igaz, hogy a legtöbb jelző vagy kapcsoló értékét megadhatjuk hosszú de beszédes, vagy egybetűs formában is. És akkor most jön a dolog legjobb része. Az *ntfsclone* program az adatokat alapértelmezésként a szabványos kimenetére küldi. Ez egyben azt is jelenti, hogy az így kinyert adatokat könnyedén átadhatjuk a legkülönbözőbb tömörítőprogramoknak, vagy átküldhetjük őket közvetlenül a hálózaton is. Alapvetően bármely olyan program fogadhatja ezt az adatfolyamot, amely képes a szabványos bemenetről olvasni. Nézzünk néhány efféle alkalmazási példát is. Ha már a mentés során tömöríteni szeretnénk a keletkező képfájlt, használjuk a következő feldolgozási sort:

```
ntfsclone --save-image --output
↳ - /dev/hda1 | gzip \
  -c >ntfs.img.gz
```

Az *ntfsclone* a képfájl adatait a szabványos kimenetére küldi, amit úgy állítottunk be, hogy a `--output` jelző után csupán egy minuszjelet adtunk meg. A következő lépésen a *gzip*

azonnal tömöríti az átáramló adatfolyamot, amely végül az ennek a kimeneteként megadott `ntfs.img.gz` nevű fájlba kerül.

Ha azt szeretnénk, hogy a képfájl egy távoli gépen keletkezzen, a következő formát használhatjuk:

```
ntfsc1one --save-image -o -
↳ /dev/hda1 | ssh \
backups@storage.mydomain.org \
"dd of/home/backups/windows/
↳ images/ntfs.img"
```

Itt a `--output` jelzőt felváltotta az egybetűs megfelelő (`-o`), a kimeneti csatorna azonban továbbra is a szabványos kimenetre irányul. Az adatfolyam ugyanakkor most az `ssh` bemenetére kerül, amely átküldi azt a hálózaton a megadott nevű gépre. Itt végül egy `dd` parancs fogja majd elhelyezni a megfelelő könyvtárban a képfájlt. Az egész folyamat, valamint az `ssh`-val való bejelentkezés is a `backups` nevű felhasználó nevében, az ő jogosultságaival zajlik. Íme egy újabb példa:

```
wget ftp://storage.mydomain.org/
↳ home/backups/
windows/images/ntfs.img.gz \
-o - | gunzip | tee /usr/
↳ local/backup/ntfs.img | \
ntfsc1one --restore-image --
↳ overwrite /dev/hda1 -
```

Ezt a részletet akár közvetlenül egy klónozószkriptből is vehettük volna, hiszen láthatólag se jelszót nem kér, se bármilyen más felhasználói bemenetre nincs szüksége. A tömörített képfájlt a `wget` segítségével tölti le, majd a `gunzip` paranccsal kibontja. Ezután egy `tee` parancs következik, amivel elágaztatjuk az adatfolyamot. Az egyik ág a Linux partícióra kerül, amely a klón képfájl egy biztonsági másolatát továbbra is hordozni fogja. A másik ág a `/dev/hda1` partícióra íródik, ami természetesen nem más, mint a visszaállítani kívánt *NTFS* partíció. A dolog működéséhez természetesen elengedhetetlen, hogy a `storage.mydomain.org` nevű kiszolgálón működjön egy anonim *FTP* szolgáltatás. A „jelszómentes” letöltés persze máshogy is megoldható. Használhatjuk például a *wget* és *Apache* megfelelő kombinációját, vagy készít-

hetünk titkosító kulcspárt is a két gépen, amelyek segítségével az *SSH* kapcsolatot automatikusan hitelesíteni tudjuk. Erre a problémára is igaz, hogy a megoldások számát szinte csak a fantázia korlátozza, hiszen a megfelelő eszközök rendelkezésre állnak. Az *ntfsprogs* csomag egy másik igen hasznos tagja az *ntfsresize*. Ez pontosan azt teszi, amit a neve ígér: úgy lehet vele átméretezni *NTFS* fájlrendszereket, hogy közben az adataink is megmaradnak. A program működik valódi partíciókon, de használhatjuk az `ntfsc1one` segítségével készített klónfájlokra is.

Érdeemes megjegyezni, hogy az *ntfsresize* nem módosítja a partíciós táblát, csupán a partíción belül található *NTFS* fájlrendszert nyomja össze. A partíciós táblát az új helyzetnek megfelelően az `fdisk` vagy az `sfdisk` segítségével módosíthatjuk. Ebben a cikkben nem tudom részletesen leírni, hogyan is kell egy lemezt particionálni. Ami azt illeti, annak a részletes és körültekintő leírása, miként kell átméretezni egy olyan partíciót, amit eredetileg teljes egészében egy *NTFS* fájlrendszer foglal el körülbelül olyan hosszú lenne, mint az egész cikk. Maga a művelet persze nem hosszadalmas, csak kissé veszélyes. Aki figyelmetlen, vagy egyszerűen csak nincs szerencséje, könnyen olyan helyzetbe kerülhet, hogy többé nem tudja bebootolni a gépét. Ezt figyelembe véve, na meg azt, hogy egy merevlemez manapság körülbelül annyiba kerül, mint egy tank benzin, a továbbiakban egyszerűen feltételezem, hogy a particionálás már kész. Tegyük fel ugyanakkor, hogy az *NTFS* partíció egyszerűen túl kicsi ahhoz, hogy befogadja a ráírni kívánt *NTFS* fájlrendszert. Ilyen helyzetbe akkor szaladhatunk bele, ha nem vettük figyelembe, hogy a legtöbb particionálásra használt segédprogram lefelé kerekíti a szektorszámot, vagy ha egy hibás merevlemez cseréltünk ki egy elvileg ugyanakörára (méretként ugyanaz a szám van ráírva), csak – az ismert okok miatt – nem jött be az elv, vagy más a geometria.

Az `ntfsc1one` kiválóan működik a „túl nagy” partíciókon, de semmit nem tud kezdeni a kicsikkel. Ha akár csak egyetlen olyan bit van a képfájl-

ban, ami nem írható ki a rendelkezésre álló szabad helyre, a program egyszerűen el se indul.

Ilyenkor az *ntfsresize* lehet a mentőangyalunk. Ahhoz, hogy kiderítsük, vajon mennyire lehet összenyomni a képfájl tartalmát, futtassuk a következő parancsot (abból a könyvtárból, ahol az `ntfs.img` található):

```
ntfsresize --info ntfs.img
```

A program egy ilyesféle jelentést fog készíteni:

```
ntfsresize v1.11.2
Device name       : ntfs.img
NTFS volume version: 3.1
Cluster size     : 4096
↳ bytes
Current volume size:
↳ 90009203200 bytes (90010 MB)
Current device size:
↳ 90009203200 bytes (90010 MB)
Checking filesystem consistency
↳ ...
100.00 percent completed
Accounting clusters ...
Space in use      : 6508 MB
↳ (7.2%)
Collecting resizing constraints
↳ ...
You might resize at 6507421696
↳ bytes or 6508 MB
(freeing 83502 MB). Please make
↳ a test run using both
the -n and -s options before
↳ real resizing!
```

A fenti diagnózis szerint pánikra semmi ok, a fájlrendszer tartalmát akár 6508 MB-ra is összenyomhatjuk. Persze annak sincs értelme, hogy a lehető legkisebbre összenyomjunk egy fájlrendszert, olyannyira, hogy ha ezt a rendszerpartíció tartalmával tesszük meg, akkor a *Windows* valószínűleg el se tud indulni a kialakult helyhiány miatt. Okos ember hagy némi helyet a jövőbeni „növekedésre”. Maga a program ugyan azt javasolja, hogy ha nem vagyunk biztosak a dolgunkban, akkor futtassuk először teszt üzemmódban a `-n` és `-s` kapcsolók egyidejű megadásával, ha azonban tényleg biztosra akarunk menni, akkor legyen egy biztonsági másolatunk az eredeti képfájlról. Persze mindkét módszert is használhatjuk egyszerre.

Ha a fájlrendszert 10000 MB méretűre akarjuk összenyomni, a következő parancsot kell kiadnunk:

```
ntfsresize --size=10000M
↳ ntfs.img
```

Működés közben az ntfsresize elég sok diagnosztikai információt ír ki, amely sok egyéb mellett a következőket is tartalmazza:

```
100.00 percent completed
Updating $BadClust file ...
Updating $Bitmap file ...
Updating Boot record ...
Syncing device ...
Successfully resized NTFS on
↳ device 'ntfs.img'.
```

Ezzel elvileg elő is állítottunk egy olyan NTFS fájlrendszert, ami már ráfér a véletlenül alulméretezett partícióra.

## Összefoglalás

Azt ugyan nem lehet állítani, hogy ha valaki egy biztonságos Windows munkaállomást szeretne magáénak tudni, akkor föltétlenül szüksége van az itt

bemutatott eszközökre, azt viszont nyugodtan, hogy ezek használatával jóval egyszerűbb lesz az élete.

Ha valakinek egyetlen számítógépe, és azon két operációs rendszere van, az NTFS-t kezelő linuxos segédprogramokkal teljes mentést készíthet az NTFS partícióról egy másik fájlrendszerre, vagy akár egy olyan kivethető merevlemezre, amelyen megfelelő nagyságú szabad hely áll rendelkezésre. Könnyen elképzelhető persze, hogy ez a lehetőség nem mindenkinek éri meg a fáradságot. Ugyanakkor ha már eleve adott a gépen a két különböző operációs rendszer, telepíthetjük a csomagból azokat az egyéb eszközöket is, amelyekkel fokozott biztonság érhető el, s ez már talán meggyőző érv lehet mindenki számára.

Ami a hálózati rendszergazdákat illeti, ha számos Windowst futtató gépet kell karbantartaniuk, az itt bemutatott eszközök igen nagyot lendíthetnek munkájuk hatékonyságán. A két operációs rendszert futtató gépeken létre lehet hozni megosztott partíciókat (a témáról bővebben lásd *Kevin Farham* cikkét a *Linux Journal 2006 júliusi számában*). Ha mellel a GRUB és néhány

egyéb menüfájl is telepítve van egy ilyen megosztott partícióra, akkor írhatunk olyan szkripteket, amelyek egy olyan speciális futási szinten indítják újra a gépet, amelyen működve az automatikusan visszaállítja a megadott Windows lemezképet, frissíti azt, és így tovább.

A Windows és a Linux számos területen versenytársaknak tekinthetők. Ugyanakkor a Linux egyik legnagyobb erőssége a nyitottsága, valamint parancssori eszközeinek hajlékonysága. A Linux-NTFS eszközök egyrészt lehetővé teszik a kommunikációt az NTFS fájlrendszerekkel, másrészt mivel ez a kommunikáció alapvetően egyirányú, egyben természeténél fogva ideálisan biztonságos is.

*Linux Journal 2007., 154. szám*

**Steven Mathes** először valamikor 1995-ben telepített Linuxot a számítógépére. Ekkor a Windows partíciók tartalmáról még csak a tar paranccsal lehetett biztonsági másolatot készíteni. Ha valakinek kérdése van, Stevent a smathes@tiac.net címen érheti el.



## Free Software Foundation Hungary

Alapítvány a Szabad Szoftverek Magyarországi Népszerűsítéséért és Honosításáért

### Jelenlegi tevékenységeink:

- FSF.hu Hírlevél – <http://www.fsf.hu/index.php/FSFhu-hirlevel>
- Szabad szoftveres kirándulások szervezése – <http://www.fsf.hu/index.php/Kirandulas>
- Szabad szoftveres roadshow – <http://www.fsf.hu/index.php/Roadshow>
- Magyar OpenOffice.org – <http://office.fsf.hu/>
- Magyar Mozilla – <http://mozilla.fsf.hu/>
- Magyar Linux Dokumentációs Projekt – <http://tldp.fsf.hu/>
- Fordítási útmutató a szabad szoftverekhez – <http://forditas.fsf.hu/>
- A www.gnu.org weblap anyagainak fordítása – <http://www.gnu.org/home.hu.html>
- A szoftverszabadalmak elleni mozgalomban való részvétel
- Segítség a licenck helyes alkalmazásával kapcsolatban

**Fedezd fel a szabad szoftverek világát! [www.fsf.hu](http://www.fsf.hu)**