

Az Ubuntu Linux telepítése és beállítása (3. rész)

A Linux kezdetben a hackerek, számítógépguruk világához tartozott. Fokozatosan hódította meg a kiszolgálórendszereket, az üzleti, irodai szférát, és betört az otthonokba is. Az Ubuntu éppen ezt a célt tűzte ki maga elé: otthoni felhasználásra, munkahelyen, szervereken, oktatásban egyaránt könnyen telepíthető, jól használható Linux rendszereket tudnak használni szerte a világon.

A Linux szinte automatikusan „Öntudatos Felhasználókat” hoz létre, akik tudatosan választják ezt az alternatívát, akik minden látszólagos és valódi nehézség ellenére is kitartanak mellette, sőt ember és szoftver között intim kapcsolat is szövődik. Ebben a részben *Öntudatos Felhasználóként* megismerkedünk az *Ubuntu* és így a *Debian* rendszerek működésének alapjaival.

Indítás

Az *Ubuntu* telepítése után láthattuk, hogy számítógépünkre egy boot managert is telepít, ahol más operációs rendszereket is kiválaszthatunk vagy magát az *Ubuntu* tudjuk másféle módban indítani. Ez a *GRUB (GRand Unified Bootloader)*. Ha *Windows*unk is volt egy másik partíción, akkor azt is láthatjuk itt, kiválasztásakor pedig egyszerűen a *Windows* saját boot managerének adja át a betöltés folyamatát. Ha meggondolatlan módon, már a *Linux* rendszer telepítése után próbálunk *Windowst* telepíteni, az szó nélkül felülírja a *GRUB*-ot, így kizárólag a *Windowst* fogjuk tudni elindítani. És bár ezt a hibát ki lehet javítani, mégis a békeesség kedvéért a sorrend mindig az legyen, hogy először a *Windowst*, és csak utána telepítsünk bármilyen linux disztribúciót. Ha mégis fordítva történik ez meg, akkor egy *Ubuntu Live CD* segít a gondon. A *GRUB* menüjében láthatjuk, hogy kétféle módon indíthatjuk az *Ubuntu*: szokásos és helyreállítás (*recovery*) módban. A szokásos indítás neve kb.

így néz ki: „*Ubuntu, kernel 2.6.12-9-386*”, míg a másiknál zárójelben az áll, hogy: „*recovery mode*”. Ez utóbbi arra szolgál, hogy úgynevezett „*single-user*” (továbbiakban: *single*) módban bejelentkezhessünk a gépre, amit általában hibajavításra használunk. Ilyenkor részletesen látjuk a bootolás folyamatát, nem jelentkezik be a grafikus rendszer és csupán egy felhasználó jelentkezhet be. Más disztribúciók egyéb szigorításokat is alkalmaznak, például a hálózatot sem használhatjuk *single* módban és a gyökér fájlrendszer csak olvasható módon fűzik be.

Az *Ubuntunál* ezek a korlátozások nincsenek. Ha először próbáljuk, bizonyára meglepve tapasztaljuk majd, hogy jelszó nélkül egyből root felhasználóként bejelentkezve találjuk magunkat a bootolási folyamat végén. Ez nagyon kényelmes akkor, amikor elfelejtjük a root jelszót (bár alapértelmezésben ilyen nincs is az *Ubuntuban*), viszont potenciális veszélyforrás, ha munkahelyen vagy kíváncsi gyerekekkel megáldott otthonban használjuk gépünket. Számomra mindkét dolog adva van, így az első dolgom ezt megváltoztatni úgy, hogy *single* módban is normálisan be kelljen jelentkezni. Ehhez a */etc/inittab* állományt kell szerkeszteni, természetesen root jogosultságokkal. Keressük meg a következő sorokat:

```
# what to do in single-user
# mode.
~~:S:wait:/sbin/sulogin
```

Cseréljük ki a második sort a következőre:

```
~~:S:wait:/bin/login
```

A *single* módot kiválóan alkalmazhatjuk, ha gond van a megjelenítéssel, a merevlemezzel, vagy bármilyen más tesztelést, javítást szeretnénk elvégezni.

A *GRUB* menüt a */boot/grub/menu.lst* fájlban található meg, ezt tanulmányozhatjuk, miként is kell ilyet szerkeszteni.

A szokásos bootolási folyamat során egy grafikát láthatunk egy *Ubuntu* logóval, folyamatjelzővel és bizonyos folyamatok visszajelzésével. Ez a *splash screen*. Nem tévesztendő össze az egyes ablakkezelők (*Gnome*, *KDE*, *Xfce*, stb) betöltésekor látott képpel, amit szintén splash-nek nevezünk. A *Kubuntunak* és az *Edubuntunak* is eltér a splash képe, sőt létezik egy *Xubuntu* névre hallgató is (a *Xubuntu* az *Xfce* ablakkezelőt alkalmazza a *Gnome* desktop helyett).

Kernel

A „*Linux*” név maga tulajdonképpen a kernelt, vagy más néven a rendszermagot takarja. Az egyes *Linux* terjesztések a kernel köré építkezve jönnek létre. A kernel maga a számítógépben található eszközök működéséért, összehangolásáért felelős, a memóriától kezdve a merevlemezen át a képernyőig. Azt is mondhatjuk, az operációs rendszer lelke a kernel. A *Linuxnál* ez biztosan így van.

Kernelfordítás Ubuntuhoz

Előkészületként installálnunk kell a következő csomagokat: *build-essential*, *kernel-package*, *gcc*, *g++*, *libcurses5*, *libcurses5-dev*, *libqt3-mt-dev*. Legelőször azonban adjuk ki az `apt-get update`

parancsot, hogy a legfrissebb változatokat töltsük le.

Először is szükségünk van a kernel forrásra. A patchelt *Ubuntu* kernel forrást is letölthetjük, ehhez adjuk ki a

```
sudo apt-get install linux-source-2.6.12
```

parancsot (a kernel verziószáma helyére természetesen az éppen aktuálisat írjuk).

Ha vanilla kernelt fordítunk, akkor szükségünk lesz a *linux-tree* csomagra, amely az *Ubuntu* patcheket tartalmazza.

A kernel forrás a */usr/src* könyvtárba kerül, ahol ki is kell csomagolnunk:

```
cd /usr/src
```

```
sudo tar -xjvf linux-2.6.12.tar.bz2
```

Hozzuk létre a szokásos szimbolikus linket is:

```
sudo ln -s usr/src/linux-2.6.12 /usr/src/linux
```

Ezután jöhet a konfigurálás:

```
sudo make xconfig vagy
```

```
sudo make menuconfig
```

Végül a fordítás:

```
sudo make-kpkg clean
```

```
sudo make-kpkg --append-to-version=--sajat kernel_image
```

```
↳ modules_image
```

A „sajat” helyére bármit írhatunk, ezt a karaktersorozatot hozzáfűzi majd a kernel nevéhez, így tehetjük egyedivé, könnyen megkülönböztethetővé.

A létrejött *.deb* csomagot telepítsük:

```
sudo dpkg -i kernel-image-2.6.12-sajat_10.00.Custom_i386.deb
```

Láthatjuk hogy a kernel sorozatszámát egyet ugrott (10-es). Az installálás során minden a helyére kerül, végül a *GRUB* menübe is belekerül, választható lesz a legközelebbi indításkor az új kernel.

A *Linux* kernelt *Linus Torvalds* kezdte el fejleszteni még 1991-ben (és természetesen magáról nevezet el), azóta jelentős fejlődésen ment keresztül.

A <http://kernel.org> címen található meg a legújabb és a régebbi változatokat. A kernel fő verziószáma a 2-esnél tart, az alverziószám pedig a hatosnál. Itt mindig a páros alverziószám a használható, a páratlanokat fejlesztőknek és kalandoroknak szánják.

A második alverziószám már gyorsan változik, ez 1-től megy felfelé és nem érdekes, hogy páros-e vagy páratlan. A *GRUB* menüjében az *ubuntus* sorokban szerepel, melyik milyen kernellel fog betöltődni: ez az a számsorozat, ami a „kernel” szó után szerepel. Például

a „2.6.12-9-386” a 2.6.12-es verziószámú kernelt jelenti, a 9-es egy sorozatszám, végül a 386 azt mutatja meg, milyen processzorra optimalizálták a fordítás során. Ugyanis az *Ubuntu* kernel nem az eredeti úgynevezett „vanilla” kernelt használja, hanem annak egy, a fejlesztők által foltozott (patchelt), azaz néhány dologgal kiegészített változatát. A „9”-es szám itt ezt a folyamatot mutatja meg: ez a kilencedik ilyen változata ennek a kernelnek az *Ubuntuban*. Ez nem jelenti azt, hogy mindegyik megtalálható a csomaglistában, csupán a két utolsó változat (tehát a 8-as és a 9-es). Ha valamiért nem jól működne a legutolsó kernel, egyszerűen áttérhetünk az egyel korábbi változatra is.

A haladó felhasználók maguk is fordíthatnak rendszerükhöz kernelt, így még „testhezállobbá” téve azt. A keretes írásban megadtam ennek a módját is.

Az *Ubuntu* a *Debian*ból született és mindkettő építkezik is egymásra.

A *Debian* 1993 augusztusában *Ian Murdock* indította el azzal a céllal, hogy egy olyan disztribúciót alkosson, amelyet a *Linux* és a *GNU* szellemében készítenek. A *GNU* kiáltványt *Richard Stallman*, a szabad szoftverek lelkes propagálója készítette ugyanebben az évben. A *GNU* lényege az önzetlenség, a szabadság a szoftverek készítésében és használatában.

A *Debian* ezeknek az elveknek kíván megfelelni, ezért *GNU* operációs rendszernek is hívjuk. Amennyiben *Linux* kernel az alapja, akkor *GNU/Linux* a neve. Ugyanis a *GNU* nem kizárólag a *Linuxon* alapszik, bármely más, elviek megfelelő kernel, operációs rendszer lehet az alapja. Az *Ubuntu* rendkívül szorosan kötődik a *Debianhoz*, ezért szintén ezeket az elveket tartja magára nézve kötelezőnek (kiegészülve speciális gondolatokkal, például az „emberség másokkal szemben”). Az *Ubuntu* kizárólag *Linux* terjesztés, tehát *Linux* kernelt használ.

Csomagok, tárolók, telepítés

A *Linux* operációs rendszer szoftvereit, komponenseit csomagokban (angolul: packages) tároljuk. Ezek nemcsak a lefordított alkalmazásokat, hanem a telepítéshez szükséges információkat is tartalmazzák. A nyílt szoftverek forráskódja természetesen rendelkezésre áll egyszerű, általában *tar.gz* formátumban is. A *Debiannál* (és így az *Ubuntunál* is) a források is csomagban vannak.

A *Debian* híres csomagkezelésének egyszerűségéről és használhatóságáról. amit az *Ubuntu* teljes mértékben átvett, így az egyes csomagok *.deb* kiterjesztésű fájlokban található. Ezek lehetnek egy helyi adathordozón (cd-n, merevlemezen) vagy egy távoli gépen, amit *HTTP*-vel vagy *FTP*-vel érhetünk el. Ezeket a tárolókat (*repository*), azok elérhetőségét meg kell adnunk a */etc/apt/sources.list* fájlban, amit terminálban, a fájl közvetlen szerkesztésével vagy grafikus segédeszközzel is megtehetünk. Természetesen a telepítés során a legfontosabb tárolók beállításra kerülnek. Használ-

hatóak a *Debian* tárolói is, ám ez nem ajánlott. Az *Ubuntu* csomagok eltérhetnek a *Debian* csomagoktól, így a legcélszerűbb a hivatalos *Ubuntu* tárolókat használnunk.

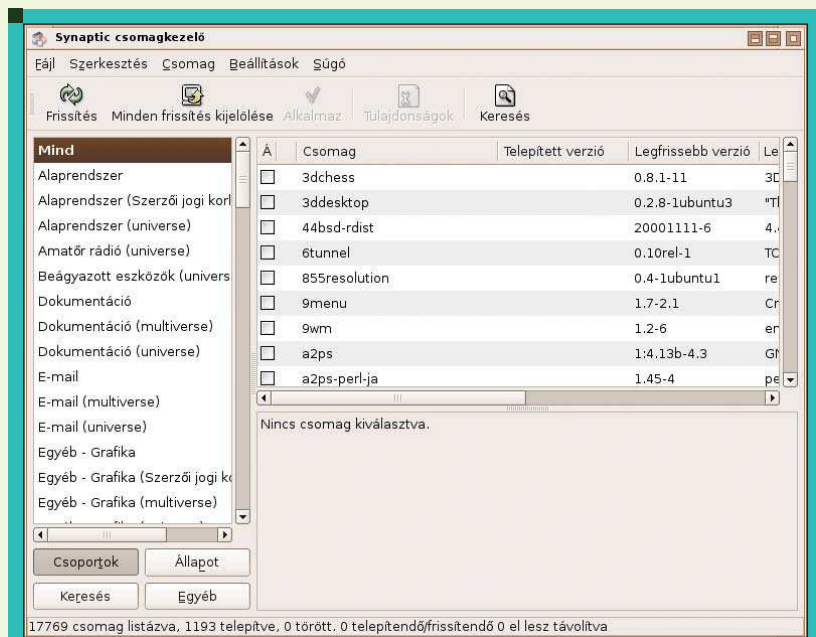
A csomagok csoportokba (*components*) vannak rendezve. Négyféle csoport van:

- A „*main*” csoport azokat a csomagokat tartalmazza, amelyek licenz szempontjából szabadok és az *Ubuntu* is teljes mértékben támogatja őket.
- A „*restricted*” csoportban lévő csomagokat szintén támogatja az *Ubuntu*, mivel széles körben használhatóság, de ezek licenz szempontból nem teljes mértékben szabad szoftverek.
- Az „*universe*” csoportban az olyan szabad szoftverek találhatóak, amik a *Linux* világához tartoznak ugyan, de az *Ubuntu*-ban már nem támogatottak. Mondhatjuk azt, hogy „*felhasználás csak saját felelősségre*”.
- A „*multiverse*” csoportban a nem támogatott és nem szabad, ám szabadon letölthető csomagok vannak.

Az „*universe*” csoportból fokozatosan építenek be újabb csomagokat a *main* csoportba, így bővül az *Ubuntu* támogatott csomagválasztéka. Telepítéskor a *main* és „*restricted*” csoportok automatikusan beállításra kerülnek. Az interneten elérhető tárolók tartalmazhatnak normál csomagokat és frissítéseket. A frissítések is két részre bonthatók: rendes és biztonsági frissítésekre. Így tehát egy tároló háromféle lehet. A *sources.list* fájlban ezt világosan megkülönböztetjük. Nézzünk egy ilyen bejegyzést:

```
deb http://archive.ubuntu.com/
↳ ubuntu breezy-updates main
↳ restricted
```

Az első kifejezés a tároló típusát (ez lehet *deb* és *deb-src* a forrásoknak) adja meg, ez után következik a tároló elérhetősége, majd a disztribúció azonosítója (*breezy*), amely ebben az esetben úgy módosul, hogy „*breezy-updates*”,



■ 1. ábra A Synaptic az Ubuntu grafikus csomagkezelője

ezzel jelezve hogy ez a normál frissítések tárolója. A „*main*” és „*restricted*” csoportokról már volt szó, ezekkel jeleztük, hogy e kétféle állapotú csomagokat szeretnénk innen installálni, amikor szükség lesz rá. A biztonsági frissítések elérési útja is más lesz:

```
deb http://security.ubuntu.com/
↳ ubuntu breezy-security main
↳ restricted
```

A normál *main* és *restricted* csomagoknál nyelvkódot is meg kell adnunk az elérési útban:

```
deb http://hu.archive.ubuntu.com/
↳ ubuntu breezy main restricted
```

Az *universe* és *multiverse* csoportoknál biztonsági frissítéseket megadhatunk a normál csomagok mellett, szokásos frissítést azonban nem, mert az az *Ubuntu* fejlesztőktől származna, de ezek nem támogatott csomagok, tehát a frissítésüket sem követik nyomon:

```
deb http://archive.ubuntu.com/
↳ ubuntu breezy universe
↳ multiverse
deb http://security.ubuntu.com/
↳ ubuntu breezy-security
↳ universe multiverse
```

Végül mindezek *deb-src* változatát is beilleszthetjük ha forráskódokat is felhasználnunk, például:

```
deb-src
↳ http://archive.ubuntu.com/
↳ ubuntu breezy-updates main
↳ restricted
```

Az *universe* csoportban lévő csomagokat gyakran használjuk, ám a *multiverse*-ben lévőket nagyobb fenntartással kezeljük. Érdemes ezt a csoportot csak ideiglenesen megadni, ha olyan dologra van szükségünk, ami máshonnan nem érhető el.

Nézzük át a *sources.list* fájlt: a legtöbb felsorolt tároló típus megtalálható benne, egyesek aktiválásához csupán a sorkezdő # jelet kell eltávolítani.

Az alaptelepítés természetesen a *universe* és *multiverse* tárolókat ilyen módon megjegyzésként illeszti be ebbe a fájlba, így azok alapértelmezésben nem elérhetőek.

Ha *CD*-t szeretnénk tárolónak megadni (a *main* csomagoknál ez is az alapértelmezés), akkor használhatjuk az *apt-cdrom* alkalmazást, amely automatikusan felderíti a meghajtóban lévő *CD*-t és a rajta lévő csomagokat. A *Debian*, és így az *Ubuntu* is a híres *apt* (*Advanced Package Tool*) csomag-

kezelőt használja, amellyel igen egyszerűen tudunk csomagokat telepíteni a tárolókból, telepített csomagokat eltávolítani, valamint keresni a csomaglistákban. A csomaglista frissítését a

```
sudo apt-get update
```

paranccsal tudjuk megtenni. Ez felderíti azt is, mely csomagok frissültek a tárolókon és eltávolítja ezeket. Automatikusan frissíteni a

```
sudo apt-get upgrade
```

paranccsal tudunk. Valamelyik csomag telepítése is igen egyszerű:

```
sudo apt-get install [csomagnév]
```

Az apt nagyszerűen kezeli a *függőségeket (dependency)* is. Egy csomag függőségei mindazokat a további csomagokat jelentik, amelyek feltétlenül szükségesek a csomag telepítése után annak működéséhez. Amikor azt apt-tal installálunk egy csomagot, automatikusan letölti és telepíti annak függőségeit is, így nem fordulhat elő, hogy egy telepített csomag működésképtelen legyen a függőségek nem megfelelő kezelése miatt. Természetesen kézi telepítéskor vagy hálózati hiba esetén előfordulhat, hogy egy csomag függőségei nem kerülnek teljesen feloldásra. Az ilyen csomagot *töröttnek* nevezzük.

Bár mindez nem túl bonyolult terminálban, mégis kényelmesebb és áttekinthetőbb kereséshez, telepítéshez és eltávolításhoz a grafikus felületet használni. Ez a *Synaptic*. A *Rendszer / Adminisztráció* menüben található meg. Az 1. ábrán látható, mennyire jól áttekinthetőek a csomagok: a bal oldali részben megtaláljuk a csomagokat logikusan szervezett csoportosításban, jobb oldalon pedig a kiválasztott csoport elemei, az egyes csomagok láthatóak. A 2. ábrán kiemeltet a jobb oldali részt, miközben rákattintottam az *openoffice.org2* csomagra. Amint látható, a telepített csomagok mellett kis zöld négyzet, a nem telepítettek mellett üres négyzet áll a bal oldalon. Az *Ubuntu* logó jelzi, hogy támogatott csomagról van szó, a „*telepített verzió*” és a „*legfrissebb verzió*” egyezése azt mutatja, hogy a legfrissebb

változatot használom. A csomag rövid leírás vele egy sorban is megtekinthető a részletes leírás pedig az alsó részben.

Telepítéshez vagy a már telepített csomagot eltávolításhoz megjelölni egyszerűen dupla kattintással kell, majd az *Alkalmaz* gombra kattintva elindul a letöltés, majd a telepítés. A *Synaptic* az *apt* csomagkezelőt alkalmazza, tulajdonképpen annak grafikus kezelőfelülete. Telepítésnél ha kívánjuk, még egy terminált is láthatunk, ahol az *apt* telepíti a kiválasztott csomagokat. Így természetesen ugyanolyan egyszerűséggel tudunk frissíteni is: a *Frissítés* gombra kattintva letölti az aktuális csomaglistát a tárolókból, majd az „*Összes frissítés kijelölése*” kiválasztása és végül az *Alkalmaz* gombra kattintással elindul a frissítés.

Az *Ubuntu*ban a frissítések kezeléséhez még rendelkezésünkre áll a *Frissítéskezelő* is, amely rendszeresen átnézi a tárolókat anélkül, hogy a *Synaptic*ot vagy az *apt-get*-et kézi futtatnunk kellene és figyelmeztet, ha új frissítések érhetőek el. Ezek itt is megtekinthetőek, ki-be kapcsolható az egyes csomagok frissítésének engedélyezése, valamint elindítható az automatikus frissítési procedúra.

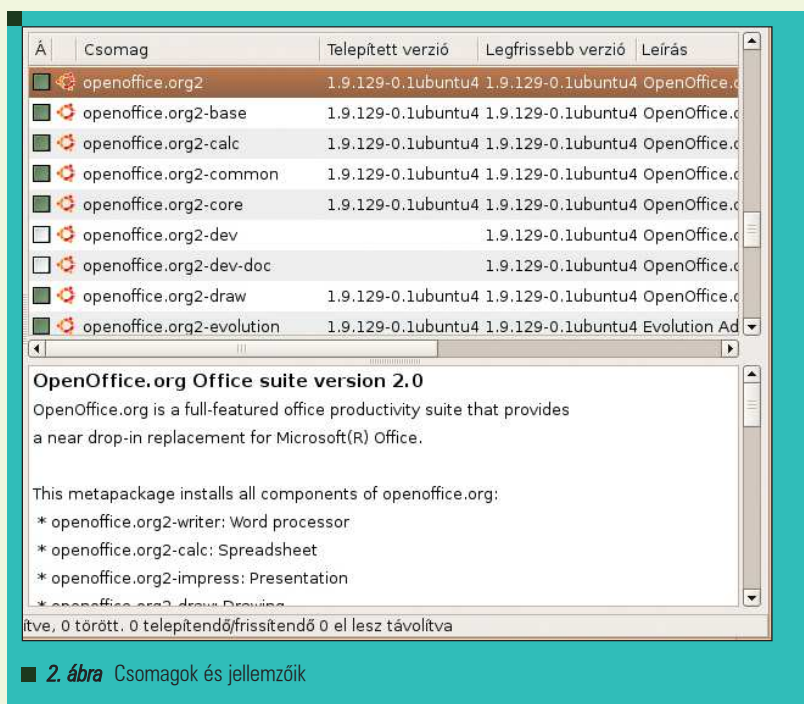
A *Synaptic*kal egyszerűen beállíthatjuk a használandó tárolókat is a *Beál-*

lítások / Tárolók menüpontban.

A megjelenő ablakban már „*Szoftverforrások*” megnevezése van a tárolók listájának. Minden egyes tétel egy sorra felel meg a *sources.list*-ben. Az „*Ubuntu*” név a hivatalos *Ubuntu* tárolókat jelzi, ami után a disztribúció verziószáma következik (most ez 5.10), majd normál tárolóknál a kiadás neve (*Breezy Badger*), frissítéseknél és biztonsági frissítéseknél pedig a „*frissítés*” vagy „*biztonsági frissítés*”. Nem *Ubuntu* tárolónál egyszerűen a tároló elérési útja jelenik meg itt. A név mellett zárójelben megjelenik, hogy bináris vagy forrás csomagokról van-e szó. A név alatt áll(nak) a megadott típus(ok).

A „*main*” megfelelője a „*Hivatalosan támogatott*”, a „*restricted*”-é a „*Szerzői jogi korlátozás alatt*”, az „*univers*”-é a „*Közösségi karbantartású*”, a „*multivers*”-é pedig a „*Nem szabad*”. A *Hozzáadás* gombbal egyszerűen tudunk új tárolókat felvenni: meg kell adnunk a tároló fajtáját és az összetevőket (típust). Egyéni tárolók elérési útját és egyéb adatait kézzel kell megadnunk. A *Synaptic*knak ezen kívül még jó pár hasznos funkciója van, amelyekre most nem térek ki: *CR-ROM* hozzáadása, törött csomagok javítása, csomagok újratelepítése, keresési szűrők megadása.

© Kiskapu Kft. Minden jog fenntartva



■ 2. ábra Csomagok és jellemzőik



■ 3. ábra A Lemezzkezelővel könnyedén tudunk partícionálni, formázni, partíciókat beállítani

Partíciók, fájlrendszer, könyvtárak

A partíciókkal feloszthatjuk merevlemezünket egymástól független részekre, amelyek eltérő fájlrendszereket, operációs rendszereket tartalmazhatnak. Ezek információit egy speciális helyen, a partíciós táblában tároljuk, a merevlemez elején. Egy partíció lehet *elsődleges*

(*primary*) vagy *logikai* (*logical*).

A PC-k történelmi okokból összesen négy partíciót engednek meg, ennyi fér a partíciós táblába (ezek az elsődleges partíciók), de ez nyilvánvalóan kevés, így azt az áthidaló megoldást választották (ahelyett hogy kibővítették volna a partíciók számát, ahogy azt a normális ember elvárná), hogy az egyik elsődleges partíciót kijelöljük kiterjesztett partíciónak (angolul: *extended partition*), amit már tetszőleges szeletre vagdoshatunk, mivel az információkat az extended partíción tároljuk.

Köszönjük, sikerült elbonyolítani a partíciókat, ha ez így nem logikus, nem az olvasó készülékében van a hiba. A *Windows* sokáig nem igazán törődött a partíciókkal (persze, így mindjárt nincs bonyolultság sem), így a felhasználók megszokták, hogy ők sem foglalkoznak ezzel. Szemben a *Linuxszal*, ahol ez fontos kérdés volt minden telepítés alkalmával. Az *Ubuntu* telepítő is felajánlja, hogy ő maga partícionál, de ekkor is három saját partíciót hoz létre (*/*, *swap*, */home*).

Hogy több partíción tároljuk dolgainkat, annak több oka lehet. Mindenekelőtt célszerű elkülöníteni magát az operációs rendszert és a saját adatainkat. Ha ezeket külön partíciókra rakjuk, egy esetleges új telepítéskor alkalmazott formázás csupán a rendszerpartíciót fogja érinteni, adatainkat nem. De azt is megtehetjük, hogy meghagyjuk a régi rendszert, az újat egy másik partícióra telepítjük, majd az új partíción lévő rendszert adjuk meg gyökérnek, miközben adataink továbbra is érintetlenek maradnak. Biztonsági, szervezési megfontolások is okolják a partícionálást. Különböző fájlrendszereket is csak így tudunk alkalmazni. Az elsődleges és logikai felosztás mellett a partícióknak egyedi típusú is rendelkeznek, amit hexadecimális számrendszerben 00-tól FF-ig számozunk. PC-ken a leggyakoribb partíció-típus a 05 vagy 0F (extended) mellett a 0C, azaz a *FAT32 LBA*. De például külön partíció-típusa van a *FreeBSD*-nek és az *OpenBSD*-nek is. Az egyes partíciókat úgy tudjuk tárolásra használni, ha valamilyen fájlrendszert teszünk rájuk (kivéve az extended partíciót, mert az további partíciókat tartalmaz). Az egyes operációs rendszerek eltérő fájlrendszer megoldásokat alkalmaznak. Mind-egyiknek megvannak a maga hívei és ellenzői is. *Windows* alatt *FAT* és *NTFS* fájlrendszerek alkalmazhatók, *Linux* alatt már sokkal változatosabb

a kép, de a leggyakoribb linuxos fájlrendszerek az *ext2* (*extended 2*), *ext3* (*extended 3*), *reiserfs*, a *swapsfs* és *tmpfs*. Míg a két utóbbi speciális fájlrendszer, addig az első három bármire alkalmazható. Az *ext2* bár elavult, kompatibilitási okokból sok helyen alkalmazták még. Az *Ubuntu* alapértelmezett fájlrendszere az *ext3*.

Egy fájlrendszer létrehozásához az illető partíciót formázni kell. Ezt a telepítés során megtette számunkra a telepítő. A 3. ábrán láthatjuk a Rendszer / Adminisztráció menüben belül található „Lemezek” alkalmazást, ahol grafikus felületen kezelhetjük a lemezeket, azok partícióit és a rajtuk lévő fájlrendszert.

Láthatunk itt a partíciók listájában egy speciális elnevezést: *cserepartíció*. Ezt angolul *swap*-nek nevezzük. Ezt használja az operációs rendszer segítségként, amikor a fizikai memória nem elég a futó szoftvereknek, így a nem használt memóriarészek közül néhányat a merevlemezre pakol, mégpedig egy *swapsfs* fájlrendszerrel formázott cserepartícióra.

A partíció tulajdonságainál megtalálhatjuk, melyik eszközön érhető el (például */dev/hda1*), milyen fájlrendszerrel formázott, mi az elérési útvonala, mekkora a mérete és ebből mennyi foglalt, valamint az állapotát (elérhető-e vagy sem). Mint bizonyára hallottunk róla, a *Linux* rendszereken minden eszköz fájlként jelenik meg. Az eszközöket a */dev* könyvtárban tároljuk. Az *IDE* merevlemezek elnevezése *hda*, *hdb*, *hdc*, stb. A *SCSI* merevlemezeké *sda*, *sdb*, *sdc*, és így tovább. Az egyes partíciók számozása egytől kezdődik, így a második *IDE* merevlemez első partíciója a */dev/hdb1* eszközön keresztül érhető el.

Az elérési út annak a könyvtárnak a helye, ahová felmountoltuk az illető fájlrendszert (partíciót). A *mountolás* egy speciális kifejezés. Ezzel tulajdonképpen hozzacsatoljuk az illető fájlrendszert a megadott könyvtárhoz. Tehát a könyvtárat megnyitva, mintegy belépünk a felcsatolt partíció fájlrendszerébe is. Általában elvárjuk, hogy az egyes partíciók már a bootfolyamat közben csatlakozzanak. Ezek megadását az */etc/fstab* fájlban tehetjük meg. Természetesen a telepítéskor megadott partíciók automatikusan bekerülnek ide.

A *Debian* és így az *Ubuntu* is a *Filesystem Hierarchy Standard (FHS)* szabványt követi a fájlrendszer felépítésében. Ez a szabvány megadja, hogyan is kell egy jólnevelt *UNIX* rendszernek kinéznie. A keretes írásban olvashatunk is az *Ubuntu* könyvtár-szerkezetéről. Akik nem szoktak hozzá a *UNIX* típusú rendszerek szigorú rendezettségéhez, talán soknak találják ezt a mennyiségű információt, de mindez nem felesleges. Előbb vagy utóbb, de az *Öntudatos Felhasználó* nem elégszik meg az egyszeres és dupla kattintás ismeretével. Operációs rendszerünk egyre közelibbé válik, míg végül kénytelenek leszünk újra fellapozni ezeket az információkat, hogy „beszélgetni” is tudjunk vele. A Lemezkezelővel bármely, nem használatban lévő partíciót, lemezt tudunk particionálni, egy partíciót formázni. Ez persze feleslegessé teszi, hogy megismerkedjünk a terminálban véggezhető felettebb veszélyes műveletekkel, de a bátrabbak és a haladók mindenképpen nézzenek utána az olyan parancsoknak, mint `mount` és `umount`, `fdisk`, `cfdisk`, `parted`, `fsck`, `mkfs`, `mkswap`, `swapon` és `swapoff`. E havi *Ubuntu* adagunk most egy kicsit talán szárazabbra sikeredett, de remélem, nem tartják olvasóim haszontalannak ezeket a dolgokat. Ígyunk talán hozzá egy kis édes francia bort, de közben építsük be tudástárunkba, hogy az *Öntudatos Felhasználó* megtisztelő címet viselhessük és büszkén mutogassuk jól működő, stabil rendszerünket ismerőseinknek, barátainknak, akik a *Nagy Kékhálós Operációs Rendszer* hívei. Na és persze az sem utolsó dolog, hogy sokrétűen tudjuk használni, amit birtokolunk. Végül, ne feledjük el, mi is az *Ubuntu*: „*Linux for Human Beings*”, azaz „*Linux emberi lények számára*”.



Molnár Norbert

(molnar.norbert@gmail.com)
35 éves, rendszergazdaként dolgozik, 5 éve foglalkozik Linuxszal.
Főként a szabad szoftverek és a számítógépes biztonság érdeklí. Budapesten él feleségével és 2 éves kisfiával. Hobbija a csillagászat és a filozófia – lehetőleg jó vörösbor mellett.

Az egyes *FHS* szerint fixen meghatározott könyvtárak a következők (a gyökérfájltól nyílnak):

- `/` Ez a gyökérfájl
- `bin` A legfontosabb parancsok
- `boot` A betöltéshez szükséges fájlok
- `dev` Eszközfájlok
- `etc` A rendszer konfigurációs fájlljai
- `home` A felhasználói könyvtárak
- `lib` Alapvető megosztott programkönyvtárak és kernel modulok
- `media` Cserélhető eszközök csatlakoztatási pontja
- `mnt` Fájlrendszerek ideiglenes csatlakoztatási pontja
- `opt` Kiegészítő szoftvercsomagok helye
- `root` A rendszer adminisztrátor könyvtára
- `sbin` A legfontosabb adminisztrátori parancsok
- `srv` Szerver szolgáltatások adatainak helye
- `tmp` Ideiglenes fájlok
- `usr` Másodszintű fájlrendszer könyvtárak
- `var` Változó adatok helye

A `/bin`-ben olyan futtatható állományok vannak, mint a `login`, `ls`, `cat`, `echo`, `mount`, stb. Ezek tehát a legalapvetőbb parancsokat takarják.

Az `/sbin` könyvtárban már olyan speciális eszközök állnak rendelkezésünkre, mint a fájlrendszer ellenőrzést szolgáló `fsck`, a hálózati útválasztó táblához szükséges `route`, vagy a formázáshoz szükséges `mkfs`.

A `/boot` könyvtárban tároljuk a kernel képeket, az ezekhez tartozó kiegészítő állományokat, valamint a boot manager állományait.

A `/home` könyvtárat bizonyára ismerjük: személyes beállításaink, saját dokumentumaink helye.

Az `/usr` könyvtárban újabb alkönyvtárakat találunk, amik egy másodszintű fájlrendszert képeznek az előbb felsorolt rendszeren belül. Itt is van `bin`, `lib`, `sbin` és olyan különleges könyvtárak, mint `X11R6`, `games`, `include`, `info`, `local`, `share`, `src`, `doc`.

Nagyon gazdag futtatható állományokban az `/usr/bin` és `/usr/sbin`. Az `/usr/src`-ben a különböző programfordításokhoz szükséges forrásokat tároljuk. Az `/usr/local`-ban a saját szoftvereinket, amik például nem léteznek az *Ubuntu* hivatalos tárolóiban. Az `/usr/X11R6` a grafikus alrendszer (*X11*) fájlrendszerét takarja. Az `/usr/doc` nevéhez hűen a dokumentációkat tárolja.

A `/var` könyvtárban olyan változó információkat találunk, mint az egyes folyamatok napló (`log`) fájlljai, a letöltött és installált csomagok tárolója, a futó alkalmazások azonosítói, a nyomtatási feladatok fájlljai, a levelezőrendszer változó adatai és így tovább.

Érdekes dolog a `/proc` könyvtár, ami tulajdonképpen az operációs rendszer folyamatainak kijelzője. Itt mindent megtudhatunk hardvereink állapotáról, a futó folyamatokról, a memóriában tárolt információkról.