

## GRUB rendszertöltő – az alapoktól kezdve (2. rész)

Telepítés, testreszabás, finomhangolás.

**E**lőző cikkünkben megismerkedtünk a **GRUB** rendszertöltő felépítésével, filozófiájával, alapjaival. Eljött az ideje annak, hogy a tettek mezejére lépjünk, s kipróbáljuk a gyakorlatban is, mire képes a program. Ehhez először megpróbáljuk telepíteni a gépünkre a **GRUB**-ot, s ha ez sikeresen megtörtént, igényeink szerint testreszabjuk a rendszert. Egyszóval ebben a cikkben átállunk **LILO**-ról **GRUB**-ra. Látni fogjuk, hogy a dolog egyszerűbb, mintha **LILO**-t konfigurálnánk, s cserébe mégis egy jóval kényelmesebb eszközt kapunk.

### Amire szükségünk lesz

Mielőtt mindehhez hozzákezdénénk, nem árt tisztázni, hogy milyen környezetre van szükségünk. Az alább bemutatott fogások ugyanis egy olyan gépet igényelnek, amelyen egynél több operációs rendszert használunk. Természetesen egyetlen Linux is bőven elég a feladat elvégzéséhez, de ekkor nem fogjuk látni, hogy milyen előnyökkel jár a **GRUB** használata. (Megjegyzem: egyetlen operációs rendszer használata esetén is előnyösebb a **GRUB**, mint a **LILO**.) Ha tehát szép színesben szeretnénk látni a dolgokat, nem árt, ha van egymás mellett legalább egy Windows és egy Linux operációs rendszer, és mindkettő működőképes. Ezen túl szükségünk van a **GRUB** csomagjaira, amelyek ma már a legtöbb terjesztésben helyet kapnak, így nemhogy fordítanunk nem kell, még csak utánajárni sem kell a szükséges elemeknek. Egyszerűen rendszerünk csomagkezelőjével telepíthetjük a kívánt összetevőket. (Ezek neve Debian, ill. SuSE alatt **grub** és **grub-doc**.) A csomagok telepítése nem jár együtt a rendszertöltő telepítésével, ami azt jelenti, hogy pusztán a telepítéstől nem törődik a régi MBR (Master Boot Record). Nem kell tehát attól félni, hogy a gépünk esetleg nem indul el. A régi rendszertöltő egészen addig a helyén marad, amíg az újat be nem állítjuk. A telepítést megelőzően igencsak ajánlatos biztonsági indítólemezt készítenünk, amivel az esetleges „katasztrófák” esetén újraindíthatjuk a rendszert – akár a régi rendszertöltő visszaállításának, akár egy új telepítési kísérlet céljából.

### Nosza, telepítsünk!

Mint már említettem, a rendszer csomagkezelőjének segítségével telepítsük a gépünkre a **GRUB** programfájljait. Ha valamilyen oknál fogva ez nem megoldható, vagy újabbat szeretnénk használni, mint ami rendelkezésre áll, akkor látogassunk

el a **GRUB** honlapjára (☞ <http://www.gnu.org/software/grub/>) és a letöltés után fordítsuk le, majd telepítsük. Ha ezzel megvagyunk, be kell írunk a **GRUB** első szakaszát a merevlemez fő rendszerindító területére (Master Boot Record – MBR), illetve azt a bizonyos „másfeledik” szakaszt az azt követő területre, ám ez utóbbi már automatikusan zajlik. Kétféle módon telepíthetjük a **GRUB**-ot az MBR-be, ezek közül azonban számunkra most csak az egyik érdekes. Létezik egy parancsállomány (script), amely **grub-install** névre hallgat. Ezt kell a megfelelő paraméterekkel meghívni, s a művelettel gyakorlatilag készen is vagyunk. A parancsállomány átmásolja a lenyomatok könyvtárából (image directory) a megfelelő állományokat az indítókönyvtárba (boot directory), amely alapértelmezésként a **/boot/grub**. Ezután meghívja a **GRUB**-ot magát, hogy a kapcsolóknak megfelelően írja be az egyes alkotórészeket a merevlemez megfelelő területeire. Lássunk erre egy példát. Minden valószínűség szerint a BIOS szerinti legelső lemezre szeretnénk telepíteni a **GRUB**-ot, és ajánlatos úgy eljárni, hogy ez valóban – IDE eszköz esetén – az elsődleges mester (primary master) merevlemez legyen. Ha ez nem így van, és másik merevlemezre telepítjük a **GRUB**-ot, akkor láncolt betöltéssel (chainloading – lásd Linuxvilág magazin 43. szám, 62-63. oldal) kell az elsődleges lemezen található rendszertöltővel elérni a **GRUB**-ot. Megjegyzem, erre általános használat esetén igen kicsi az esély, ezért maradjunk most az előző az eshetőségnél. Nincs más dolgunk, mint kiadni az alábbi parancsot: `grub-install /dev/hda`

Ez természetesen független attól, hogy az aktuális Linux partíciók hol van – ezt a telepítő parancsállomány majd elintézi. A telepítő alapértelmezetten a **/boot/grub** könyvtárban keresi a rendszerállományokat, ám ha a gyökérben is találna egyet, akkor azt használná. Erre csak különleges esetben kell vigyáznunk, akkor, ha valamiért a gyökérbe másoltuk a **GRUB** szakaszait képező lenyomatokat. Ha ez így volna, akkor a `-root-directory=<elérési út>` kapcsoló megadásával segíthetünk a problémán, amely kapcsoló természetesen akkor is hasznos lehet, ha más könyvtárban található a **GRUB** állományai – de mint említettem, ez általános esetben nem fordulhat elő.

A parancs kiadását követően pár másodperces ellenőrzés és telepítés következik. Ha ez sikeresen megtörténik, akkor elvileg minden rendben, készen vagyunk. Előfordulhat azon-

ban, hogy figyelmeztetést kapunk. Amíg ez valóban csak figyelmeztetés, addig nem valószínű, hogy gondunk lesz belőle. Én is rendre kapok ilyeneket, pedig mindössze egyetlen merevlemez van a gépemben. Ennek ellenére még mindig remekül működött a dolog.

### Hogyan tovább?

Remek! A rendszertöltő most már a helyén van, tenné is a dolgát, ha most újraindítanánk a gépet. Ezt most inkább ne tegyük, előbb készítsünk egy menüt, hogy kényelmesebben választhassunk a telepített rendszerek között. A menüt a **GRUB** alapértelmezetten a `/boot/grub/menu.lst` nevű állomány alapján állítja elő, tehát itt lehet megadni az egyes menüpontokat, s ezen kívül még a menü színét, az esetleges háttérképet illetve grafikus beállításokat. Ugyanakkor ne felejtjük el, hogy ezeknek a lehetőségeknek a megvalósítása még erősen gyerekcipőben jár. Jómagam még színeket sem használok, teljesen jól működik az egyszerű, fekete alapon fehér betűs változat is.

Hogy ne kelljen sokat gyötrődnünk a menüvel, a **GRUB** fejlesztői, illetve a leírás készítői mellékeltek egy példaállományt, melyet a csomagok dokumentációi között találunk. Ez a legelterjedtebb terjesztésekben a `/usr/share/doc` könyvtár, s ezen belül keressük a `grub/examples/menu.lst` fájlt, amit másoljunk át valahová, ahol kedvünkre alakíthatjuk.

### A menu.lst fájl

Ha szerkeszteni kezdjük a fájlt, észrevehetjük, hogy a beállítások két részre oszthatók: vannak általános beállításokra, illetve magukra a menüpontokra. Ezekon kívül természetesen még számos egyéb, haladó szintű előzetes beállítás is létezik.

Az általános beállítások a menü automatikus kezelésére vonatkoznak. Nézzük sorban a lehetőségeket.

`timeout <másodperc>`: Meghatározza, hogy mennyi ideig várakozik a rendszertöltő a felhasználói beavatkozásra, mielőtt az alapértelmezett operációs rendszer automatikusan betöltődne. Ha akármelyik billentyűt lenyomjuk, akkor ez a várakozási idő érvényét veszti. Ez azt jelenti, hogy ezután addig nem indul el semmi, amíg magunk nem választunk ki egyet a lehetőségek közül. Az alapértelmezett rendszer indításáig hátralévő időt a gép indításakor a rendszertöltő menü jobb alsó sarkában olvashatjuk.

`default <menü száma>`: Ez határozza meg, hogy a timeout pontban megadott idő eltelte után melyik menüpontot töltsse be automatikusan a **GRUB**. A menüpontoknak nincs külön száma, a menüfájlban elhelyezett sorrendjük szerint sorszámozódnak, de nullától kezdődően.

`fallback <menü száma>`: Abban az esetben, ha az alapértelmezett menüpont nem működne, az itt megadott

számú menüpontra ugrik a **GRUB**, s ezt tölti be. E menü számára is ugyanazok a szabályok érvényesek.  
**híddenmenu**: Ha ezt beírjuk a menüfájlba, rendszertöltéskor a menü nem jelenik meg, s a lejárati idő (timeout) eltelte után az alapértelmezett (default) menüpont aktiválódik. Ha ezen alkalomadtán változtatni szeretnénk, akkor az Esc billentyű megnyomásával előhozhatjuk a menüt. Ezen kívül már csak egy menüparancs van, ez pedig a `title`. Ezzel készíthetünk új menüpontot az alábbi módon:  
`title <menüpont látható neve amit majd kiválasztunk>`

Létrehozza a menüpontot. Ez alá kell sorban megadni azokat a parancsokat, amelyeket a menüpont kiválasztásakor szeretnénk végrehajtani.

Ezeknek nem kell feltétlenül a rendszertöltésre vonatkozniuk. Választhatunk színt, megadhatunk hálózati beállításokat, és így tovább.

A **GRUB** a menüpont aktiválásakor a `title` után a fájl végéig, vagy a következő `title` utasításig szereplő minden parancsot végrehajt. Ezek a bizonyos parancsok valójában a GRUB parancsértelmezőben kiadható utasítások, amelyek egy héjprogramhoz hasonlóan hajtódnak végre. A parancsok közül most csak a legfontosabbakat nézzük meg.

### Linux rendszer betöltése menüből

Ez az menübejegyzés az alábbi módon néz ki:

```
title Linux betöltése
root (hd0,1)
kernel /boot/2.6.7/bzImage root=/dev/hda2
```

A `root` parancs segítségével jelölhetjük ki a gyökérpartíciókat, amin azután a rendszermag lenyomatot keresni fogjuk. A lemez megadása két azonosítóval történik. Az első azonosító a lemez BIOS által ismert sorszámát mutatja. Ez a sorszámozás nullával kezdődik. A `hdd` lemez tehát `hd3`-mal azonosítható a GRUB ezen parancsa számára. A második szám a lemezrész sorszáma, szintén nullától kezdődően. Esetünkben a `hda2`-n van a Linux, amelyet az 1-es szám képvisel. Mindezek megállításához természetesen alaposan ismernünk kell a rendszerünket.

A `kernel` parancsall állíthatjuk be, hogy hol is található magának a rendszermagnak a lenyomata, majd ezt követően szóközzel elválasztva egymás után adhatjuk meg a rendszermag kapcsolóit. Ezek közül egy mindenképp kötelező: a `root` paraméter, amely nem azonos az előző sorban szereplő **GRUB** utasítással. A rendszermagnak ez azt jelzi, hogy melyik lemezrészre kell a Linux gyökerekének tekintenie. Néhány terjesztés használ még `initrd` állományt is. Ha a mi Linuxunk is ilyen, az állományt `initrd <elérési út>` paranccsal adhatjuk meg.



## Windows rendszer betöltése menüből

Ez az menübejegyzés az alábbi módon néz ki:

```
title Barmilyen windows inditasa
rootnoverify (hd0,0)
makeactive
chainloader +1
```

A `rootnoverify` parancs a `root` parancs egy különleges változata, amely a parancs kiadásakor nem ellenőrzi a lemezrész meglétét, állapotát, és egyéb paramétereit. Egyszerűen elhiszi nekünk, hogy ott van. Abban az esetben használjuk, amikor a `GRUB` számára ismeretlen típusú lemezrész, és ezzel együtt ismeretlen típusú operációs rendszer helyét adjuk meg. A kapcsolókat tekintve ugyanúgy kell használni, mint az egyszerű `root` utasítás.

A `makeactive` parancs a `DOS` szempontjából aktívnak jelöli meg az előzőleg megadott lemezrész. Mint tudjuk, a *Microsoft* termékei csak úgy hajlandók (az esetek jelentős többségében) elindulni, ha aktívnak kijelölt lemezrészről töltjük a rendszert. Ezt biztosítja ez a bizonyos parancs.

A `chainloader +1` parancs kiadásával a láncolt betöltést aktiváljuk, melynek hatására a betöltés vezérlése átadódik az adott lemezrész rendszertöltőjéhez. Ez utóbbi természetesen úgy érzi majd, mintha a BIOS ébresztette volna fel, s szó nélkül elindul a kiválasztott operációs rendszer. Ha készen vagyunk a menüvel, akkor másoljuk át a `/grub/boot` könyvtárba (`menu.lst` néven), s a következő indítás során már az új menübejegyzés szerinti rendszertöltő indul, próbáljuk ki hát!

## GRUB indítólemez készítése

Indítólemez készítésére ismét több lehetőségünk is van. Az első a `Linux dd` parancsa, amellyel átmásolhatjuk a `GRUB` szakaszait egy hajlékonylemezre. A másik módszerrel magát a `GRUB`-ot hívjuk meg, amely a merevlemez helyett a meghajtóban lévő hajlékonylemezre települ. Számunkra az első lehetőség az érdekes. Ehhez lépünk be a `GRUB` lenyomat-könyvtárba (`/usr/lib/grub/i386-pc`), majd adjuk ki a következő parancsokat, egymás után.

```
dd if=stage1 of=/dev/fd0 bs=512 count=1
dd if=stage2 of=/dev/fd0 bs=512 seek=1
```

## Rendszerindítás GRUB indítólemez használatával

Előfordulhat (nem csak most, a telepítés következtében, hanem más esetekben is), hogy valami miatt csak indítólemez használatával tudjuk életre lehelni rendszerünket. Ha például újratelepítjük Windowst, akkor az rossz esetben felülírta a `GRUB`-nak az MBR-be telepített részét. Ha ez után szeretnénk a *Linuxot* elindítani, akkor egy `GRUB` lemezzel kell újra életre kelteni pingvinünket. Igaz ugyan, hogy most még semmi bajunk nincs, de amolyan tűzoltási gyakorlatként azért próbáljuk ki a lemezes módszert. Ha floppy-ról indítjuk a gépet, azonnal a `GRUB` már emlegetett parancsértelmezőjét kapjuk a megszokott menü helyett. Ebben ugyanazokat a parancsokat használhatjuk, mint a menüfájlban. Próbaképp indítsuk el a *Linuxot* ebből a parancssorból. Ehhez adjuk ki egymás után ugyanazokat a parancsokat, melyeket a Linux rendszer betöltése során is használnánk.

Minden egyes parancs beírásakor TAB-bal kiegészíthetjük a beírandó parancsot, kapcsolót, illetve listát is kapunk a lehetséges értékekről, csakúgy, mintha a bash-ben tevékenykednénk. Még az egyes lemezrészek fájlrendszerét is látjuk – feltéve, hogy az megegyezik a telepített `GRUB` által ismert lemezrész formátumával. Ellenkező esetben csak a lemezrészben található fájlrendszer típusáról kaphatunk információt. A parancsok kiadása után nem történik semmi. Ahhoz, hogy elindíthassuk a megadott operációs rendszert, adjuk ki a `boot` parancsot, amely az eddig begépeltek alapján elindítja a rendszertöltést. Ha betöltődött a rendszer, máris számtalan lehetőségünk van a helyreállításra, telepítésre vagy amire éppen szükségünk van.

Most pedig nézzük meg, hogy sérült MBR esetén hogyan állíthatjuk helyre a `GRUB`-ot, hasonlóan, indítólemez segítségével. Ez egyébként nem más, mint a már említett második telepítési forma. Lássuk, mely utasításokat kell kiadnunk.

Először is a `root` parancs segítségével adjuk meg, melyik lemez hányadik lemezrészén található a `GRUB` programfájljainak telepített változata. Ez alapértelmezetten a linuxos lemezrészünk, tehát az, amit a példánk során a menüben is megadtunk (`hd0, 1`). Ezt követően adjuk ki a `setup (hd0)` parancsot, amely az elsődleges lemez fő lemeztöltő területére helyezi a `GRUB` első szakaszát, s gondoskodik az esetleges „másfeledik” szakaszról.

Ha nem vagyunk biztosak abban, hogy melyik lemezrészben van a `GRUB`, adjuk ki a `find /boot/grub/stage1` utasítást, mely megkeresi azt a lemezrész, melynek `/boot/grub` könyvtárában ott van a `GRUB` első szakasza, majd kiírja a lemezrész nevét. És ezzel készen is vagyunk, helyreállítottuk a `GRUB`-ot.

Bármely indítás során elérhetjük ezt a bizonyos – rendkívül hasznos – parancssort. Indításkor a menünél csak nyomjuk meg a C billentyűt, és máris ott találjuk magunkat. Ezek után egy `help` parancs segítségével kilistázhathatjuk, hogy milyen elérhető utasítások közül válogathatunk. Látni fogjuk, hogy a rendszertöltéssel kapcsolatban ezzel gyakorlatilag bármilyen fontos feladatot elvégezhetünk. Ha vissza szeretnénk térni a menühez, csak nyomjuk le az Esc billentyűt.

## Végezetül

Láthattuk, hogy a `GRUB` rendszertöltő esetében messze több lehetőségünk van a működés beállítására mint bármely hasonló program esetében. Ugyanakkor határozottan állíthatom, hogy ennek ellenére használata jóval egyszerűbb, mint a *LILO*-é. Tekintve, hogy a telepítésen túl számos egyéb, hasznos fogásra is kitértem, őszintén remélem, hogy azok számára is értékes információval szolgáltak ezek a cikkek, akik olyan terjesztést használnak, amely már eleve a `GRUB`-ot telepíti. Ilyen például a *SUSE*, de tudomásom szerint az összes friss *Linux* változat a `GRUB`-ot részesíti előnyben. Talán nem véletlenül.



**Komáromi Zoltán**

(komi@kiskapu.hu)

23 éves, a BME hallgatója, mellette

PHP-programozóként dolgozik.

Kedvenc területe a multimédia.