

Átállás (2. rész)

Folytatódjék hát a kezdő és a leendő Linux-felhasználóknak szánt sorozatunk!

Sorozatunk első részben főként a Linux és a hozzá hasonló szabad felhasználású és ingyenes Unix-rendszerek előnyeit és hátrányait taglaltuk, mostani cikkünkben pedig a Windows és a Linux közötti legfontosabb különbségekre térünk ki. Ahogy már előző számunkban is tisztáztuk, a Linux Unix-alapú operációs rendszer. A két rendszer használatában ez a legfontosabb különbség, tehát ez az írásunk lényegében a Unix- és a Windows-rendszerek közötti eltérésekkel foglalkozik. Akinek már akadt dolga valamilyen Unix-rendszerrel, annak rengeteg dolog ismerős lesz a Linuxszal való ismerkedés során. A most leírtak tehát inkább azoknak szólnak, akik ezeket a rendszereket csupán hírből ismerik.

Az első, amit meg kell jegyeznünk, hogy a Linux egy valódi többfeladatos 32-bites operációs rendszer. A már őskövületnek számító Windows 3.1 például jóindulattal sem nevezhető operációs rendszernek, mivel tulajdonképpen nem más, mint grafikus héj a szöveges DOS-környezeten. Ugyan lehetővé tette, hogy „egyszerre” több alkalmazást is futtathassunk, de minden futó folyamat (process) maga dönthette el, hogy meddig kíván futni, és mikor adja át a futási lehetőséget a többi folyamat számára. Ez nem valami hatékony megoldás, mert ha egy alkalmazás lefagy, az egész rendszert magával rántja.

A Windows 95 megjelenésével javult valamit a helyzet, a folyamatok ütemezését már maga a rendszer végezte (habár messze elmaradt a Windows NT-től), és büszkén hirdette magáról 32-bites mivoltát. Ám ez utóbbi nagyon csalóka, mivel a régi alkalmazásokkal való megfelelés (compatibility) következtében rengeteg 16-bites kódrészt hagytak a rendszerben, sőt a rendszer a 32-bites védett módból a 16-bites valós módba ide-oda kapcsolgatott. Ez pedig a rendszer üzembiztonságára nézve nagy veszélyt jelentett. A Windows 98/ME forgalomba kerülésével további fejlődést tapasztalhatunk: amit lehetett, átírtak 32-bites környezetbe, továbbá egyéb biztonsági

szolgáltatásokkal is kiegészítették.

A Windows NT/2000/XP azonban teljesen más, ezek – a Linuxhoz hasonlóan – valódi többfeladatos és 32-bites operációs rendszerek. Minden alkalmazás egymástól teljesen elkülönített „térben” fut, és az egyik nem tud a másik dolgába „belerondítani”.

A Linux magas szintű üzembiztonságát nem annak köszönheti, hogy kevesebb hibát tartalmaz, mint például a Windows. Az igazat megvallva, tele van hibával, hiszen ezt is emberek alkották. A különbség csak annyi, hogy egy alkalmazás lefagyása vagy helytelen működése korántsem okozhat akkora galibát, mint egy Windows 95/98/ME rendszerben.

Ez annak köszönhető, hogy egy folyamat nem férhet hozzá közvetlenül gépünk erőforrásaihoz, például rendszerünk memóriájához. Csak abba a tartományba írhat, ahová a rendszermag számára engedélyezi. Hiba esetén pedig a rendszermagnak jogában áll az alkalmazás futásának beszüntetése és a memóriából való „kitakarítása”.

Tehát Linux alatt is szép számmal fagnak az alkalmazások, de ez csak a legritkább esetben okoz fennakadást a rendszer és a többi folyamat működésében. A Linux csak akkor dőlhet össze, ha vagy a rendszermagban vagy a gép szintjén történik lefagyás (ezek az esetek szerencsére elég ritkák), viszont egy alkalmazás egyedül nem képes erre. Most térjünk a tárgyra, azaz a Linux- és a Windows-alapú rendszerek különbségeinek bemutatására!

A Linux-rendszerek magja modularizált. Ez azt jelenti, hogy a rendszermag bizonyos részei különálló egységekben (modulokban) találhatóak, amelyeket kedvünkre ki- és betölthetünk. Modulokban leggyakrabban a különböző eszközök támogatását szokás elhelyezni. Az ilyen modularizált rendszermagok előnye, hogy ezek a modulok csak akkor töltődnek be, amikor szükség van rájuk, egyébként nem foglalnak helyet a memóriában.

Azt hogy a rendszermag mely részei kerüljenek modulba és melyek ne, a rend-

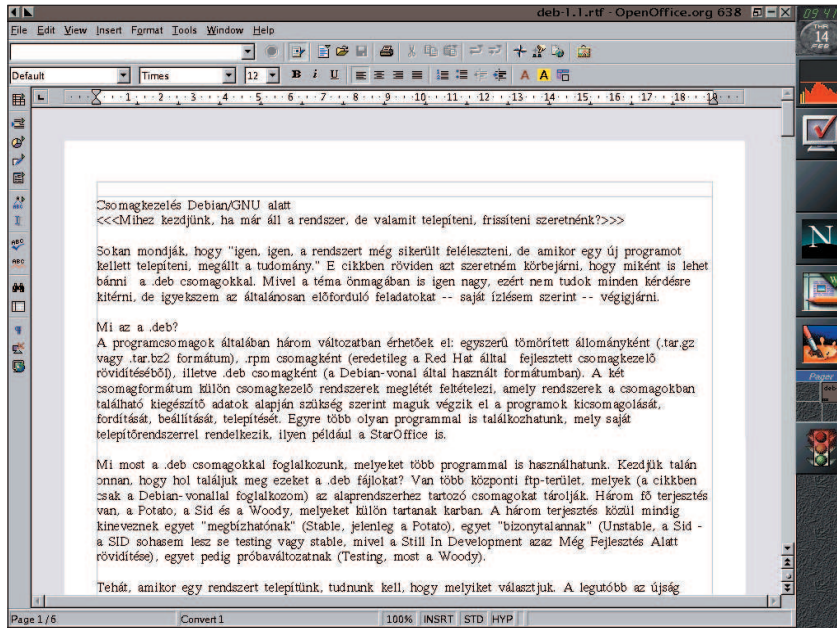


szermag újrafordításakor mi is meghatározhatjuk.

Előző számunkban említettük, hogy a rendszermag fordítása a Linux világában hétköznapi műveletnek számít, ugyanis számtalan előnye van, ha saját magunk készítette rendszermaggal dolgozunk. Egyrészt a mi processzorunkra lesz testreszabva (ez nagyobb futási sebességet eredményezhet), másrészt csak azok az elemek kerülnek közvetlenül a rendszermagba, illetve modulba, amelyekre szükségünk van (ezáltal a rendszermag mérete csökken).

A Windowshoz szokott felhasználók ezt első látásra furcsa és bonyolult műveletnek tarthatják, pedig semmiféle programozási előképzettség nem szükséges hozzá.

Előző számunkban szintén megemlítettük, hogy a Unixok elsősorban hálózati rendszerek. Az ilyeneket általában tehát nem egy, hanem több felhasználó használja, ezért bizonyos biztonsági szolgáltatásokra van szükség. A rendszernek például nem szabad engednie, hogy bármely felhasználó ész nélkül törölgethessen benne, írthasson bele stb. Ezért a Windows NT-khez hasonlóan, a munka megkezdése előtt itt is egy felhasználónév-jelszó párossal kell jelentkezni. A root-felhasználó, vagyis a rendszergazda bír a legtöbb jogosultsággal. Csak ő képes például a különböző rendszerbeállításokat megváltoztatni, további felhasználókat létrehozni, illetve törölni stb. A rendszergazdára (kevés kivétellel) semmilyen korlátozás nem vonatkozik, bármit letörölhet – szabadon „garázdálkodhat” a rendszerben. Ezért a Unix világában nagyon fontos, hogy csak akkor lépünk be rendszergazdaként, ha valami olyasmit szeretnénk elvégezni, amelyhez ezek a jogosultságok szükségesek. Egyébként hozunk létre magunknak egy saját felhasználót, amivel a „hétköznapi” munkákat végezzük. A felhasználókat egyébként csoportokba is szervezhetjük. Ez azért hasznos, mert az adott csoportra meghatározott jogosultságok a csoportban lévő összes



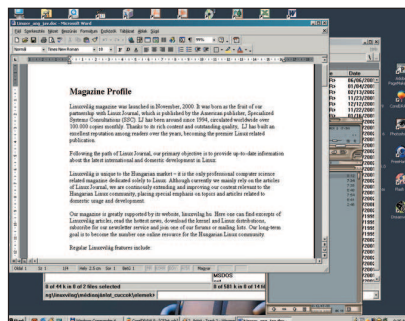
Linuxos képernyő

felhasználóra vonatkoznak.

A Linux fájlrendszere az ext2. Itt minden állományhoz egy úgynevezett fájlleíró (i-node) tartozik, amely az adott fájl tulajdonságait (például méret, tulajdonos, lemezblokkok címei stb.) tartalmazza. Egyes Linux-terjesztések egyébként azt is lehetővé teszik, hogy a rendszert akár FAT lemezterületre is telepíthessük. Az újabb változatokban már az ext2 fájlrendszert annak továbbfejlesztett változatára, az ext3 fájlrendszerre cserélhetjük le. Ez már egy úgynevezett naplózó (journaling) fájlrendszer. A naplózó fájlrendszerek lényege, hogy a merevlemezünkön lévő adat-szerkezetből naplót szerveznek, és az összes függőben lévő írásműveletet a napló végéhez csatolják. Így egy szegmensben fájlleírókat és adatblokkokat egyenesen találhatóak. A naplózó fájlrendszerek működése ennél azért sokkal összetettebb. A lényege az, hogy sok kisebb írásművelet esetén a naplózó fájlrendszerek használata gyorsabb működést eredményez, és például kevésbé érzékeny az áramszünet okozta fennakadásokra.

A linuxos fájlrendszereken kívül a rendszer mag egyéb fájlrendszereket is támogat, például a FAT16-ot és a FAT32-öt, továbbá az OS/2 HPFS-ét és a Windows NT NTFS-ét.

A Unix-rendszerekben a különböző lemezegységek fájlrendszerei nincsenek különválasztva egymástól, mint a Windows-rendszerekben, hanem mindegyikük egységes, összefüggő könyvtárszer-



Windowsos képernyő

kezetet alkot. Ha például a gépünkbe helyezett CD-n található állományokhoz szeretnénk hozzáférni, akkor azt előtte be kell fűznünk ehhez a fájlrendszerhez. Ha megtörtént, CD-nk tartalmát egy megadott könyvtár alatt (általában a `/cdrom`) találjuk. Ezt a műveletet befűzésnek (mount) hívjuk, sorozatunk későbbi számaiban e témára részletesen ki fogunk térni.

A Unix-rendszerek másik sajátossága, hogy szinte az összes rendszerbeállítás szöveges állományok tömkelegében tárolják. Ezek az állományok a `/etc` könyvtárban találhatóak, és általában csak a rendszergazda módosíthatja őket. Azzal, hogy a különböző beállítások mely állományokban vannak, és azokat miként módosítjuk, sorozatunk további részeiben foglalkozunk. A rendszer bináris állományai (azaz maguk a programok) a `/bin`, illetve a `/sbin` könyvtár alatt lelhető fel. Ezekben a könyvtárakban csak a legalapvetőbb

felhasználói és rendszerfelügyeleti programok találhatóak, a többiek a `/usr/bin`, illetve a `/usr/sbin` könyvtárban helyezkednek el.

A `/home`-ban a felhasználók úgynevezett saját könyvtárait találhatjuk. A saját könyvtár szintén fontos fogalom a Unix világában. Ez nem más, mint a felhasználó egyéni könyvtára, ahol a magánjellegű dolgait tárolhatja. A saját könyvtár neve általában a felhasználó azonosító-jával egyezik meg.

Mivel a Unix többfelhasználós rendszer, alapvető fontosságú meghatároznunk, hogy ki milyen állományhoz férhet hozzá. Minden fájlnak van egy tulajdonosa és egy csoportja. Az állományra vonatkozó jogokat a tulajdonos, illetve a rendszergazda határozhatja meg. Megmondhatja, hogy milyen jogok vonatkoznak a tulajdonosra, a csoportra és mindenki másra. A Unixban alapvetően háromféle jogosultság létezik: olvasás (read), írás (write) és végrehajtás (execute). Az utóbbi értelemszerűen csak a futtatható állományoknál érdekes. Ezenkívül létezik még néhány úgynevezett módosító bit is, ezekről a későbbiek folyamán fogunk szót ejteni. Egy másik unixos „szokás”, hogy az ilyen rendszerek alatti különböző eszközöket úgynevezett eszközállományok (*device files*) segítségével érhetjük el, amelyek a `/dev` könyvtár alatt találhatóak. Ha például a második soros kapunkra akarunk hivatkozni, ami a DOS-ban a COM2 névre hallgat, ezt Unix alatt a megfelelő eszközfájl nevének megadásával tehetjük meg (jelen esetben `/dev/ttyS1`). A merevlemezeknek és a gépünkben lévő többi meghajtónak (drive) is megvan a maga megfelelő eszközfájla. Például az elsődleges ATAPI merevlemeznek a `/dev/hda`, az azon található első lemezterületnek pedig `/dev/hda1` a neve. Sorozatunk következő részében már a gyakorlati oldalról vizsgáljuk meg a Linuxot: megismerkedünk néhány alapvető utasítással, és bemutatunk néhány egyszerű rendszerfelügyeleti műveletet.

Garzó András

(garzoand@interware.hu)

Körülbelül három éve foglalkozik Linux- és más Unix-rendszerekkel. Legjobban az operációs rendszerek lelkivilága érdekli, de nyitott egyéniség. Kedvenc étele a palacsinta, és van egy Richard nevű macskája. Minden észrevétel, megjegyzés, levelet szívesen fogad.