

## Rendszergazda-képzés Linuxszal

Egy főiskolai gépteremben rendhagyó képzés zajlik.

**A** rendszergazdai teendők ellátása minden operációs rendszer esetében nagyon fontos, azonban a legtöbb egyetemen és főiskolán nem foglalkoznak rendszergazdák képzésével. Vajon akkor hogyan tanulhatják meg a diákok a szükséges fogásokat? A válasz egyszerű és közhelyszerű: maguktól. A következő ésszerű kérdés: mi teszi lehetővé azt, hogy valakiből rendszergazda lehessen? A válasz: egy rendszergazdának tisztában kell lennie az operációs rendszerek és a hálózatok alapvető működésével. Egy hagyományos, gyakorlati programozást is tanító oktatási rendszerrel ellentétben, a rendszergazdák képzésénél inkább az elméleti összefüggésekre kell helyezni a hangsúlyt. Például egy merevlemez gyorstár beállításához nem kell ismerni a lap-táblázatok kiosztását.

A Grand Valley Állami Egyetemen indítottunk egy tantárgyat, ennek keretében a diákok az operációs rendszerekről és a hálózatkezelés alapjairól tanulhatnak, és az egész oktatást a rendszergazdai teendőkre alapoztuk. Hallgatóink végzős informatikusok, akik egyébként sehol nem tanulnának az operációs rendszerek és a hálózatkezelés szabályairól és elveiről. Az órákon a legkülönbözőbb témák kerülnek terítékre: a felhasználók és csoportok kezelése, a fájlmegosztás és a folyamatok. A hálózatkezelés területéről pedig az alkalmazás-réteg-protokollok, az átviteli réteg és a hálózati eszközök beállítása került a tantervbe.

A tantárgy két fő részből áll: az elméleti oktatásból, melynek során a diákok az operációs rendszerek és a hálózatkezelés elméletéről hallgathatnak, és a gyakorlatokból, ahol a tanult elvekkel a gyakorlatban szabadon kísérletezhetnek. Ebben a cikkben egy rövid ismertetést adunk arról, hogy az oktatást miként segíti a Linux.

### Az EOS Linux gépterem

Az operációs rendszerekkel való kísérletezésre felállított EOS gépterem huszonegy Pentium III-as gépből áll, ezekben egyenként 128 MB memória, 10 GB merevlemez, egy hajlékony- és egy ziplemez-meghajtó van, és mindegyiken RedHat 6.2 fut. A gépterem nemcsak kísérleti terep, hanem mindennapi munkakörnyezet is: a gépeket a végzős informatikusok és a programozók használják, de más szakok hallgatói is idejárnak kipróbálni az operációs rendszereket. Tehát a gépterem nem tisztán kutatóegység, hiszen egészen hétköznapi teendőkre is használják a gépeket. Éppen ezért fel merült az, hogy félévenként huszonegy diáknak adjunk rendszergazdai jogosultsága a gépekhez. Így viszont azzal a nehézséggel szembesültünk, hogy a hallgatóknak a legalapvetőbb teendők elvégzéséhez is rendszergazdaként kellene belépniük az operációs rendszerbe.

A megoldás a zipmeghajtók használata lett, ennek köszönhetően minden hallgató saját Linux-környezetében dolgozhat. Mindenki készít egy indítólemezt és egy ziplemezt, mely az alapfájlrészt tartalmazza. Az óra kezdetén a diákok leállítják a gépüket, behelyezik az indítólemezt és az alaplemezt, majd újraindítják a rend-



szereket. Innentől kezdve mindenki saját Linux-változatával dolgozik, ezt természetesen rendszergazdaként használhatja, és kényelmesen elvégezheti rajta az aznapi feladatokat. A nap végén egyszerűen leállítják a gépeket, kiveszik a lemezeket, és az utánuk érkező felhasználók már a merevlemezről dolgozhatnak, a szokásos RedHat 6.2 környezetben.

### Az indítólemez elkészítése

Jelenleg a 2.2.13-as rendszermagot használjuk, ez ráfér egy hajlékonylemeze és semmiféle módosításra nincsen szükség. A rendszermag egyedi beállítására azonban szükség van, két okból is. Az első a SCSI-emuláció beállítása, ehhez a CONFIG\_CHR\_DEV\_SG és a CONFIG\_SCSI értékét igazra állítjuk. A zipmeghajtók ugyanis IDE csatolóak, és ezeket SCSI-emulációval kell működtetnünk, mert tapasztalataink szerint az IDE-meghajtó nem kezeli jól a nagy fájlokat. A rendszermag beállításának másik fontos oka a merevlemez letiltása. Említettük, hogy a laboratórium gépeinek merevlemezein RedHat 6.2 található. Ha nem tiltanánk le a merevlemez elérését, a hallgatók a zipről történő rendszerindítás után a mount paranccsal egyszerűen befűzhetnék a helyi merevlemez, és innentől kezdve teljhatalmat élveznének felette (akár a rendszergazda jelszavát is megváltoztathatnák). A merevlemez letiltását két változó (a CONFIG\_BLK\_DEV\_IDE és a CONFIG\_BLK\_DEV\_HD\_IDE) hamisra állításával oldottuk meg.

A rendszermag további beállításai lehetővé teszik a hálózati eszközök, például a SysV init stb. engedélyezését. Miután a rendszermagot beállítottuk, egyszerűen lefordítjuk azt. A hajlékonylemezen a mke2fs programmal először létrehozzuk az ext2 fájlrendszert, majd a lefordított rendszermagot rámásoljuk. A lemezen egy indítórészt is létre kell hoznunk (cp /boot/boot.b /mnt/floppy) és a LILO-t az alábbiak szerint kell beállítani:

```
boot=/dev/fd0 map=/mnt/floppy/map
install=/mnt/floppy/boot.b
prompt
compact
```

```
timeout=50
image=/mnt/floppy/vmlinuz
label=linux
root=/dev/sda1
read-only
```

Ez a LILO-beállítás indíthatóvá teszi a lemezt, és a /dev/sda1-et állítja be gyökérlemezként. Mivel SCSI-emulációt használunk, ezért a /dev/sda1 (az első SCSI lemez) a ziplemez. Ezt követően a /sbin/lilo -C /mnt/floppy/lilo.conf paranccsal telepítjük az új LILO-fájlt.

### Az alapelem elkészítése

Az alapelem a Slackware 7.0-s Linuxon alapul. Azért választottuk a Slackware-t, mert pontosan szabályozható, hogy mely csomagok kerüljenek telepítésre, és így a rendelkezésre álló 100 megabájtos helyet tökéletesen kihasználhatjuk. Az óra anyagából kiindulva az a, ap és n csomagokat telepítettük, az alábbi parancsok segítségével:

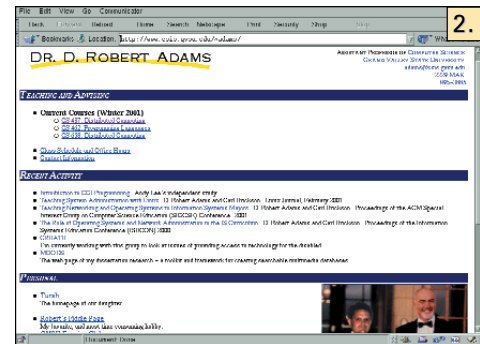
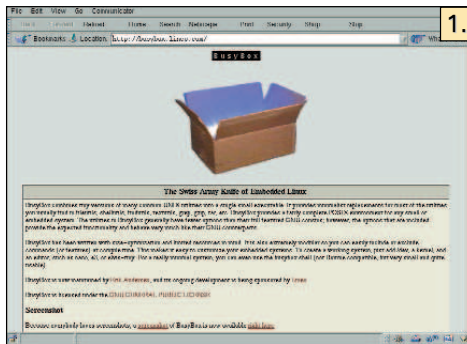
```
ext2 lemezrészét hozunk létre a ziplemezen
# fdisk /dev/sda1
fájlrendszer létrehozása a ziplemezen
# mke2fs /dev/sda1
befűzzük a ziplemezt
# mount /dev/sda1 /mnt/zip
# cd /mnt/zip
# tar -zxvf /tmp/slackware/a1/aaa_base.tgz
# sh install/doinst.sh ; rm -rf install
```

Az utolsó három lépést minden szükséges csomag esetében meg kell ismételnünk. Sajnos, a helyszűkéből adódóan sok hasznos csomagot ki kellett hagynunk. Például a C/C++-t magában foglaló d, és a rendszermag forrását tartalmazó k csomagokat is, hogy csak a legfőbb hiányságokat említsük. Ezen a helyzeten természetesen változtatni szeretnénk: azt tervezzük, hogy egy nagyobb cserélhető lemezt alkalmazunk

a feladatra. A Lineo nevű, beagyazott Linuxszal foglalkozó vállalat Linux-változata kis terjedelmű, és minden szokásos linuxos eszközt rendelkezésünkre bocsát. A BusyBox pedig egy nyílt forrású kezdeményezés, melynek célja, hogy a hagyományos linuxos eszközöket egyetlen, kisméretű futtatható fájlba sűrítse (lásd Kapcsolódó címek).

### Önműködő folyamat

Hallgatóink a félév első óráján elkészítik indító- és alapelemeiket. Természetesen az első óra alkalmával nem mindegyikük tudása áll olyan szinten, hogy erre önmaguktól képesek legyenek. Ezért kitaláltunk egy önműködő rendszert, ennek segítségével néhány parancs



kiadásával elkészíthetők a lemezek. Először is a lemezek működő változatát fájlba mentjük az alábbi parancsokkal:

```
dd if=/dev/fd0 of=floppyimage
dd if=/dev/sda1 of=zipimage
```

A hallgatóknak tehát csak a dd parancsot kell használniuk a fájlok kiírásához:

```
dd if=floppyimage of=/dev/fd0
dd if=zipimage of=/dev/sda1
```

### Összefoglalás

A Linux nálunk nagyszerűen bevált a rendszergazdák képzésében. A ziplemezek használatával minden diákunk függetlenül dolgozhat saját kis Linuxával, anélkül, hogy a gép merevlemezén található rendszernek bármilyen baja esne. Bár egy ziplemezen csupán 100 MB fér el, a nyílt forráskódnak köszönhetően a Linux-változatot sikerült egyetlen ziplemezre sűríteniünk, sőt, a gép merevlemezének módosítását is egyszerűen megakadályozhattuk. A gépeket sokan távolról is szeretnék elérni, és az alkalmazott módszernek egyetlen hátránya az, hogy a ziplemezről való újraindításakor ez lehetetlen. Úgy hirdeltük át a nehézséget, hogy a nem ziplemezt használó gépeket megkülönböztető névvel láttuk el. A felhasználókat pedig figyelmeztettük arra, hogy a ziplemezes gépek nem érhetőek el folyamatosan a hálózaton keresztül. A további részletek iránt érdeklődők az órajegyzeteket, a hallgatók feladatait és az egyéb programokat a képzés honlapján megtalálhatják.

#### D. Robert Adams

a Grand Valley State University segédprofesszora. Kutatási és érdeklődési körébe tartozik az objektumközpontú programozás, a Palm számítógépek programozása és a különféle programnyelvek.

#### Carl Erickson

a GVSU segédprofesszora. Ma jelenleg a XiphNet, Inc. programszerkezeteket kutató csapatában dolgozik. Szakterülete az elosztott és az objektumközpontú programozás.

**Kapcsolódó címek**

**The Role of Operating Systems and Network Administration in the IS Curriculum.** Az Information Systems Education Conference (ISECON) anyagából, 2000, D. Robert Adams és Carl Erickson.

**Teaching Operating Systems and Networking to Information Systems Majors.** A Special Interest Group of Computer Science Education (SIGCSE) Technical Symposium anyagából, 2000, D. Robert Adams és Carl Erickson.

**A CS437 honlapja:** (2. kép)  
 ➔ <http://www.csis.gvsu.edu/~adams/CS437>

**BusyBox:** (1. kép)  
 ➔ <http://busybox.lineo.com/>  
 Busy Box: The Swiss Army Knife of Embedded Linux, Linux Journal, 2000. október, szerző: Nicholas Wells.