

## Irány az iroda!

**A**zt hihetnénk, hogy manapság már, mikor még az IBM is komoly összegeket költ a Linux támogatására (lásd a 9. oldalon), nem kérdés többé, hogy rendszerünk nagykorúvá vált. Vannak viszont olyan felhangok, hogy a Linux használhatatlan és ismeretlen rendszer. Történt, hogy napilap hozott le cikket, melyben a Linuxot komoly munkára használhatatlan rendszernek tünteti fel, sőt, a cikkíró felháborodását fejezi ki, hogy hivatalos helyen miért is ajánlják ezt a rendszert támogatott gépekhez a „mindenhol használt” Windows helyett stb. Felvetődik a kérdés, hogy mit lehet tenni? Vagy egyáltalán érdemes-e vitába szállni olyan valakivel, aki komolyan gondolja, hogy a hazai piacon a Microsoft termékein kívül szinte nem is létezik más operációs rendszer. Nem azt mondom, hogy nincs sok Windows-felhasználó, de azért erős túlzásnak vélem az elhanyagolható kategóriába rakni az összes Unix-rendszert, a Macintoshokat, vagy a különböző új rendszereket. A Google Web Directory alatt mintegy nyolcvan témakör található az „Operating System” alatt, ezek között a legtöbb létező rendszert takar.

De ha nagyon unatkozunk, ellátogathatunk a Grafikus Felületek Tárlatába (<http://pla-netx.com/linebackn/guis/>), ahol szintén hihetetlen mennyiségű operációs rendszer képével találkozhatunk. Ha van kedvünk, elnézhetünk a különböző rendszereket összehátraló oldalhoz is (<http://leb.net/hzo/ioscount/>, ez még 1999-ben történt gyűjtés alapján készült). Nocsak, a „www” kategória alatt a magyar oldalaknál az összes Windows együtt nem tesz ki húsz százalékot sem, míg a Linux harminc fölött van. Ej, no! Rendben, ez 1999-es adat, sajnos nem tudom, hogy van-e frissebb kimutatás. És egy újabb vonal is kezd bekúszni a Linux világába. A politika. Ennek örülök, mint a köszvénynek. Egyre-másra reppennek fel a hírek, hogy a Linuxszal takarózva így meg úgy kapott egy cég harmincmilliót (<http://www.nepszabadsag.hu/Body.asp?CollID=57&DocID=9589>). Az összeg egyébként egy tízzel több, és riportot is olvashatunk a cég vezetőjével a <http://euroastra.com/mlinux/> címen. Jó sok pénzről van szó! Itt nem is az a kérdés, hogy meglesz-e a magyar Linux-változat, hanem az, hogy ez a változat mit tud, amit a többi nem? A Contact sokat ígér, kíváncsi vagyok, mennyit tud majd megtartani. Találkozunk majd itt is olyan fordításokkal, amit a Mandrake vágott a fejemhez? („Ékeld be a(z) cdrom-t, melynek neve cdrom1.”).

De a Linuxellenes hangokra visszatérve, már-már azon gondolkodom, hogy felesleges felvenni a kesztyűt, a komoly szakemberek úgyis tudják, mi a sületlenség, és melyik hír tekinthető szakmailag megalapozottnak. A gond csak az, hogy a vállalati döntéshozók legtöbbször bizony nem informatikus, és ha egy nagynevű újságban olvassa, hogy a Linux megbízhatatlan, használhatatlan és egyáltalán.... akkor lehet, hogy eleve kizár olyan lehetőségeket, amelyek pedig mind a pénztárcája, mind az alkalmazottak számára előnyös. Akkor tehát ennyit megér: „Hé, Mindenki! A Linux nagyon jól használható, üzembiztos és gyorsan fejlődik!” (Zárójelben jegyzem meg, hogy remélem, a végén lesz egy helyesírás-ellenőrző rendszere is. Még mindig nem tetszett meg egyik egyetlennek sem a feladat? Még támogatás is volna rá!)

Tehát most már nem az a kérdés, hogy cégünknel előkerül-e a Linux, hanem hogy mikor. Ha még nem készültünk fel rá, akkor rohamtempóval kezdjünk bele! A 26-28. oldalon olvashatunk az önképzésről, valamint a hazai oktatási helyzetről. Az önképzésről szóló cikk

felhívja a figyelmet néhány, az életből vett hibára. Ezeket csak alátámasztani tudom. Ne várjunk csodát, de ha azt is várunk, legalább ne holnapra várjuk!

A rendszergazdák számára az egyik legfontosabb témakör a biztonság. E hónapban is sok érdekes cikk lát napvilágot e témában; megnézzük, hogy miként érdemes kiépíteni egy olyan hálózatot, mely kívülről is elérhető gépeket tartalmaz, folytatjuk *Mátó Péter* és *Borbély Zoltán* cikksorozatát, és megnézzük, mire is használhatjuk az nmap nevű pásztázót.

De a biztonság mellett azért kell szakítani egy kis időt a szórakozásra is! Foglalkozunk egy kicsit az MPEG szabvánnyal elméleti síkon (62–63. oldal), majd vizsgáljuk meg gyakorlati megvalósításait is (32–33. oldal). Ha valakinek nem tűnt volna fel, még Amerikában is igen jó vélemény alakult ki a magyar programozókról. Talán nem véletlen, hogy a cikk írója magyar fejlesztést ajánl első helyen?

Jó olvasást és jó linuxozást kívánok!



Szy György

a Linuxvilág főszerkesztője, a Kiskapu Kiadó vezetője. Mindenki véleményét és levelét örömmel várja az alábbi levélcímen: [Szy.Gyorgy@linuxvilag.hu](mailto:Szy.Gyorgy@linuxvilag.hu)

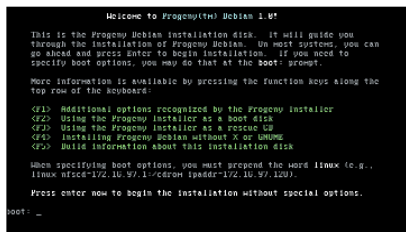
### A lap megjelenéséről

*Sajnos nagyon sokat csúsztunk. Ezt szégyellem is, hiszen a csúszások okai között dobogós helyezéssel kell említenem magamat. Viszont most komoly döntésre jutottunk. Gondolom, mindenki emlékszik, hogy a februári és márciusi számot összevonva jelentettük meg, egy árban, egy számként. Ezzel az összevont számmal valójában időt szertünk volna nyerni. Nos, nem jött össze. Ezt a trükköt kíséreljük meg még egyszer, most már az előző próbálkozás keserű tapasztalataival felvértezve, és a tervek szerint a #7-et ismét összevontként jelentetjük meg július közepén, június-július néven. Valójában ez az olvasó szempontjából annyit jelent, hogy a következő lapszámunk is emelt oldalszámmal, de azonos árban jelenik meg, és reményeink szerint augusztustól már a természetes rend uralkodik: mindig a hónap elején kerül a polcokra a legfrissebb szám. Az előfizetők esetében is – ugyanúgy, ahogy ez a február-márciusi számnál történt – a június-július egy lapszámmal minősül, az előfizetések önműködően egy hónappal tovább tolódnak, tehát a fél évre szóló előfizetés összesen hat lapszámot jelent. Remélem, hogy a nyár végére tisztázódnak a megjelenés körüli bonyodalmak és végre mindenki időben juthat hozzá kedvenc linuxos magazinjához!*

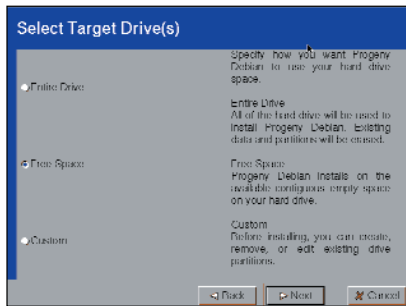
# Programvadászat

## Progeny 1.0 Newton telepítési segédlet

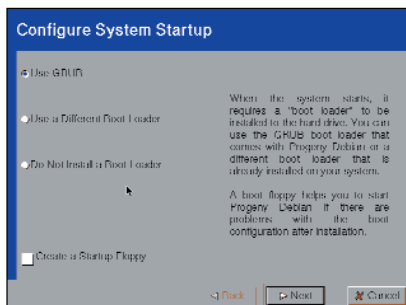
**A** Progeny Debian száz százaléig Debian/GNU Linux-megfelelő rendszer, fejlesztését a Debian alapítója *Ian Murdock* indította el. 2001 április 9-én jelent meg a végleges 1.0-s változat, melynek az első telepítő CD-je kapott helyet mellékletünkön. Ennek telepítéséhez szeretnénk egy kis segítséget nyújtani.



A telepítés kezdete



A lemezterület felosztása



Rendszerindítás beállítása

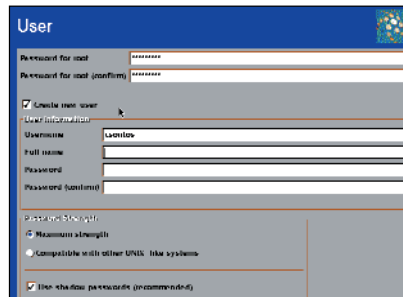
### Első lépések

A CD rendszerindításra alkalmas, így indítólemezek készítése nélkül is nekikezdhetünk a telepítésnek. Az első képernyőn a telepítésről több adatot is kaphatunk, vagy egyszerűen ENTER-t ütve továbbléphetünk a telepítőbe. A Debiannál ellentétben, ezt már grafikus felületen telepíthetjük, az eszközök

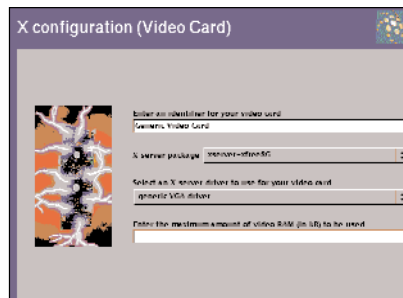
felismerése itt is önműködően zajlik. A lemezterület felosztásánál felhasználhatjuk az egész meghajtót, szabad területet vagy saját kiosztást készíthetünk. Ezután következik a rendszerindítás beállítása: a grub, a LILO vagy más indításvészérlő közül választhatunk. Ezután az elkészített lemezterületek formázása következik, egy alap rendszer kerül a lemezünkre. Vegyük ki a lemezt a hajlékonylemez-meghajtóból és indítsuk újra gépünk.

### A rendszer beállítása

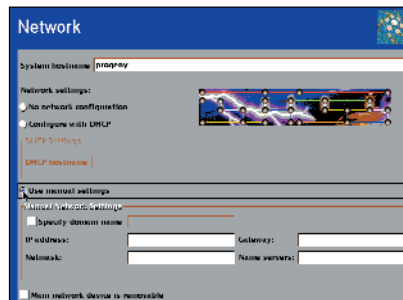
Válasszuk ki a felsorolásból, mit szeretnénk a rendszerünkben beállítani. Először



Felhasználó hozzáadása

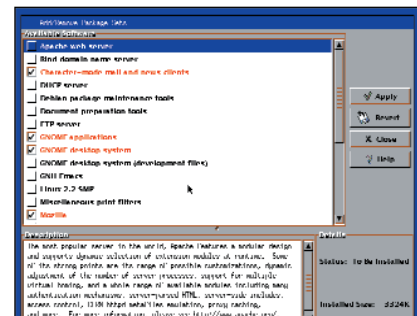


Grafikus felület beállítása

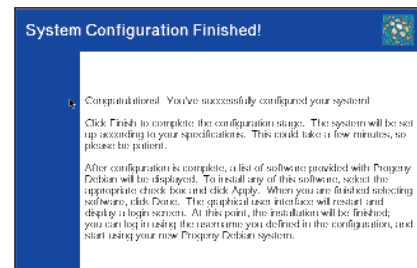


Hálózat beállítása

az időzónát és a földrajzi helyzetet kell megadni, majd a rendszergazda jelszavainak megadása és egy egyszerű felhasználó hozzáadása következik. Ugyanebben az ablakban beállíthatjuk a jelszavaink védelmi szintjét is. Majd az X Windows rendszer beállítása következik, melynek meg kell adnunk a következő adatokat:



Csomagok telepítése



A telepítés sikeresen befejeződött

- a X kiszolgáló típusa,
  - a VGA kártya típusa,
  - a monitor beállításai: monitorunk adatainak megadása három módban történhet, lehet egyszerű (Simple, itt csak a monitorunk méretét kell tudnunk), közepesen nehéz (Medium) és a profinak való (Advanced),
  - a felbontás és színmélység beállítása,
  - az egér csatolófelülete.
- Ezután a levelezőrendszerünket hangolhatjuk saját igényeinkhez. Választhatunk a szolgáltatótól az otthoni egyedüli gépig bármit. Ezt a nyomtató kiválasztása és a hálózat beállítása követi.
- A telepítendő csomagokat kényelmesen kezelhető listából választhatjuk ki. Miután a csomagok telepítésével is elkészültünk, a rendszerünk használatra kész. Kellemes linuxozást!

© Kiskapu Kft. Minden jog fenntartva

## Rendszermag

A legfrissebb Linux rendszermag, a 2.4.4-es megtalálható a CD *Rendszermag* könyvtárában. Javítottak az USB meghajtókon, a ReiserFs fájlrendszeren, valamint a hálózati meghajtókon is.

## Rendszerfrissítés

A mostani CD-mellékleten a Progeny Debian-hoz eddig megjelent összes frissítés megtalálható. Ezek telepítésével rendszerünk a legfrissebb csomagokat fogja tartalmazni és kevésbé lesz sérülékeny akár a külső, akár a belső támadókkal szemben. Ezekben a csomagokban ugyanis már kijavították az eddig megtalált összes hibát. Rendszerünk apt forrásához az apt-cdrom add paranccsal adhatjuk, ezután az apt-get update ; apt-get upgrade parancsokkal fríszíthetjük azt.

## Opera 5 Végleges

Nagy előrelépés a linuxos böngészőpiacon az Opera 5 végleges, megbízható változatának megjelenése. A CD *WEB/Opera* könyvtárában az i386-os gépekhez .deb, .rpm, .tar.gz, Power PC-hez pedig .rpm és .tar.gz csomagok találhatóak. Telepítésük Debian csomagkezelő esetén dpkg -i *csomagnév*, rpm-es rendszer esetén pedig az rpm -i *csomagnév* paranccsal lehetséges. Mindenképpen érdemes frissíteni a régebbi változatot, illetve



kipróbálni az új Opera tudását, mivel gyors és kevés memóriát igényel. Erről lásd még a szemközti oldalon.

## Mozilla 0.9

CD-mellékletünkre felkerült még a Mozilla 0.9 is. E kiadás újdonságai közé tartoznak:

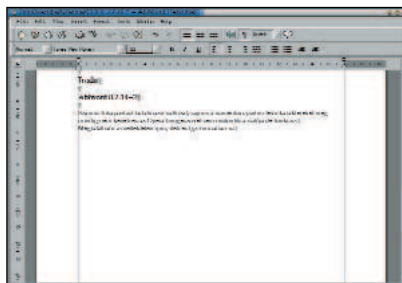
- az önműködő proxybeállítás,
- a növelt teljesítményű Personal Safety Manager 2.0,
- a MailNews, melynek felülete az áttevezés után jelentősen gyorsult,
- a Java teljesítményén is javítottak,
- új a súgó, egyelőre számos dolgot átvettek a Netscape 6-ból, de folyamatban van a Mozillára való átültetése.

A CD *WEB/mozilla* könyvtárában található a kész, lefordított csomagok, valamint a program forráskódja is.



## Sim City 3000 bemutatóváltozat

Kellemes városépítő játék, sok izgalommal! Kilenc különböző nyelven játszható Linuxon (angol, francia, német, dán, olasz, lengyel, portugál, spanyol és svéd).



Legkisebb rendszerkövetelmények: 2.2.x rendszermag és glibc-2.1 könyvtár, Pentium 233 MHz vagy gyorsabb processzor (300 MHz Pentium II processzor ajánlott), 4 megabájtos grafikus kártya, XFree86 3.3.5 vagy újabb változat, 16 bites színmélység, 8x CD-ROM-meghajtó, legalább 32 MB RAM (64 MB az ajánlott), 16 bites hangkártya és 230 MB lemezterület. A CD *Jatekok/Sim\_City\_3000* könyvtárából telepíthető az `sh sc3u-demo.run` parancs kiadásával. Jó játékot!

## Irodai alkalmazások

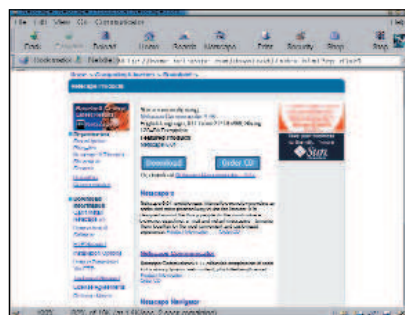
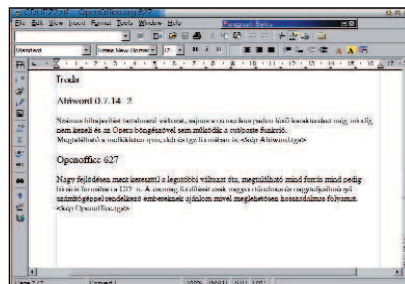
### Abiword 0.7.14-2

Számos hibajavítást tartalmazó változat, sajnos a számbillentyűzeten lévő karaktereket még mindig nem kezeli, és az Opera böngészővel sem működik a *cut*, illetve a *paste* szolgáltatás.

A mellékleten megtalálható .rpm, .deb és .tgz formában is.

## Openoffice 627

Óriási fejlődésen ment keresztül a legutóbbi változat megjelenése óta. A lemezen megtalálható mind forrás, mind pedig bináris formában. A csomag fordítását csak nagyon türelmes és nagyteljesítményű számítógéppel rendelkező embereknek ajánlom, mivel rendkívül hosszadalmas a folyamat.



## Netscape 4.77

A 4-es sorozatú Netscape böngésző legújabb változatát ismét kisebb-nagyobb hibajavításokkal látták el. Ezt a változatot azonban már nem annyira támogatják (ugyanis a RedHat ezentúl valószínűleg Mozillával fogja szállítani a rendszereit), pedig az erőforrásigénye korántsem akkora, mint az újabbaké. A könyvjelzők és a Java parancsállomány hibáit is kijavították. A frissítést mindenkinek ajánljuk!



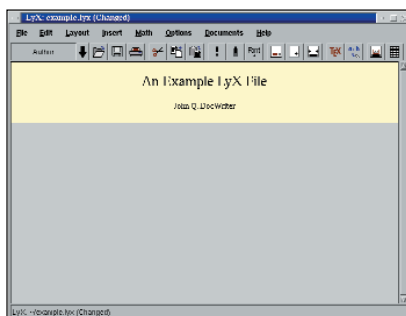
**Csontos Gyula**  
(Csontos.Gyula@linuxvilag.hu)  
a Linuxvilág hír- és CD-szerkesztője, valamint a [www.linuxvilag.hu](http://www.linuxvilag.hu) tartalomfelelőse.

### Kapcsolódó címek

- [www.openoffice.org](http://www.openoffice.org)
- [www.abisource.org](http://www.abisource.org)
- [www.opera.com](http://www.opera.com)
- [www.mozilla.org](http://www.mozilla.org)
- [www.lokigames.com](http://www.lokigames.com)
- [www.netscape.com](http://www.netscape.com)
- [www.kernel.org](http://www.kernel.org)

## LyX

Megjelent a LyX 1.1.6fix2. A LyX egy TEX–LATEX-alapú grafikus felületen futó szövegszerkesztő. Nagyon sok hibajavítást hajtottunk végre, ezek közül néhány:



- kijavították a kurzor hibáját a görög karaktereknél,
  - a helyi nyelv kiválasztását,
  - megszüntették az összeomlást *Edit->Reconfigure* futtatása után,
  - és igazították a nyomtatáson is.
- ➔ [www.lyx.org](http://www.lyx.org)  
 Letölthető  
 ➔ <ftp://ftp.lyx.org/pub/lyx/stable/lyx-1.1.6fix2.tar.gz>  
 ➔ <ftp://ftp.devel.lyx.org/pub/lyx/lyx-1.1.6fix2.tar.gz>

## Linux-tábor Szerencsen

2001. július 2–8. között Szerencsen Linux-tábor lesz. Az érdeklődők a nyaralás időtartama alatt Linuxot tanulhatnak, eközben családjuk is pihenhet. A táborról bővebb tájékoztatás és jelentkezési lap a ➔ [linuxabor.webhome.hu](http://linuxabor.webhome.hu) címen található.



## NetBSD tűzfal könnyedén

A ➔ [www.dubbele.com](http://www.dubbele.com) címen elérhető egy NetBSD-alapú tűzfal. A telepítéshez csak két hajlékonylemez és internet-kapcsolat szükséges.



Miután a két lemez anyagát letöltöttük és rámásoltuk a hajlékonylemezre elkezdhetjük a telepítést. Figyelem, ha nincs a rendszerben támogatott hálózati kártya, akkor a telepítő hibáüzenettel leáll! Ezzel a csomaggal bármilyen régi kiszolgált számítógépből készíthetünk tűzfalat, ehhez legalább 386-os processzor, 8 MB RAM, 40 MB merevlemez-hely és két hálózati kártya kell.

- ➔ [www.dubbele.com](http://www.dubbele.com)

## Mandrake Linux 8.0

A Mandrake Linux fejlesztői a könnyű telepíthetőséget és beállíthatóságot tartják szem előtt. Ennek szellemében fejlesztették a 8.0-s változatot is.

Főbb jellemzői:

- Egyszerű és gyors grafikus telepítés (körülbelül fél órát vesz igénybe).
- Önműködő alkatrészelismerés és -beállítás. Kezdek számára különösen eszményi, érdemes azonban megnézni a támogatott alkatrészek listáját. (A lista elég széles, de nem minden alkatrész támogatott.) A 2.4.3-as rendszermaggal pedig USB, Infravörös és FireWire eszközeinket is vezérelhetjük.
- Egyszerű rendszerkarbantartás a Mandrake Control Center segítségével.



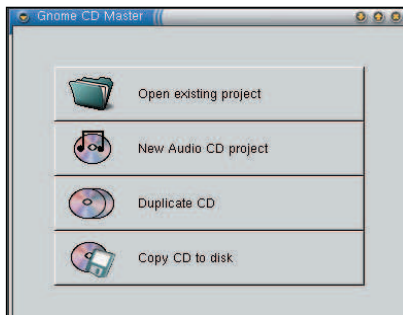
Felhasználói felületként választhatjuk a KDE 2.1.1-et, a GNOME 1.4-et, a WindowMakert és még sok mást. Az XFree86 4.0.3-nak köszönhetően teljes 3D-s támogatás érhető el, így már a játékos kedvűeknek sem kell sokat bajlódniuk a beállításokkal.

- ➔ [www.linux-mandrake.com](http://www.linux-mandrake.com)

## Gnome CD Master 1.1.5

Ez a program mindent tud, amire egy audio-CD másolásához szükségünk lehet.

A cdrdao programot használja, amivel CD-ket lehet egy az egyben kiírni a merevlemezűkre, mintegy lenyomatot készítve róla és így minden az eredetivel megegyező módon kerül át az új lemezre.

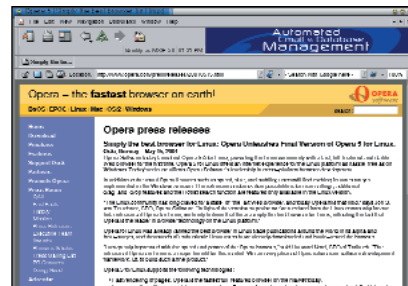


- ➔ <http://cdrdao.sourceforge.net/gcdmaster/>  
 ➔ <http://cdrdao.sourceforge.net/gcdmaster/download.html>

## Opera 5 végleges újdonságai



Megjelent az Opera 5 végleges változata Linuxhoz. Ez a böngésző méltán örvend nagy népszerűségnek a linuxos táborban, mivel kicsi, gyors és kevésbé érzékeny a hidegre (nehezebben fagy!). Egyelőre a Java támogatást még nem oldották meg,



de várhatóan az is hamarosan bekekerül. Javascript, StyleSheet támogatása is megfelelően fejlett. Használata közben ugyan még mindig előfordul egy-két kellemetlen dolog – ilyenek például a popup ablakok, amikor az összes ablak összeugrik –, de ennek ellenére is nagyon jó programot sikerült létrehozniuk a fejlesztőknek.

➔ [www.opera.com](http://www.opera.com)



© Kiskapu Kft. Minden jog fenntartva



### JBLinux 2.0

Megjelent a JBLinux 2.0-s változata. A JBLinux 1.1-es már szerepelt az 5. CD-mellékletünkön (február–március). Ezt a Linuxot a csomagok frissessége jellemzi, szinte mindentől a legújabb található benne. Egy kis ízelítő a változásokból:

- a telepítési folyamatot újraírták
- támogatja az SGI XFS fájlrendszerre történő telepítést
- a telepítési eljárást több típusra osztották: Munkaállomás, Kiszolgáló, Minimális, Teljes és Testreszabható.

A csomagok tartalmazzák a 2.4.4-es rendszermagot, a glibc 2.2.3-at, a gcc 2.95.3-at, az XFree86 4.0.3-at, a KDE 2.1.1-et és a Ximian GNOME 1.4-et.

Ebben a kiadásban megtalálható még a Mozilla 0.9 is, valamint a grafikus felület beállításához az Xconfigurator programot használja.

➔ <http://www.jblinux.net>

### PostgreSQL 7.1.1

Ez a változat – mint ahogy a száma is mutatja –, hibajavításokat tartalmazó kiadás. Néhány változás és hibajavítás az előzőhöz képest:

- pg\_dump javítás, a program most már 7.0 adatbázisokat is kezel
- JOIN, Unicode karakterkezelés, ODBC felület javítása
- Python-bővítés

Akit érdekel a teljes lista az a ChangeLog fájlban megtalálhatja. Ennél a változatnál nincs szükség a dump/restore műveletre, ha 7.1-esről frissítünk. A fejlesztők nagy erőfeszítéssel dolgoznak a 7.2-es változaton.

➔ [www.postgresql.org](http://www.postgresql.org)

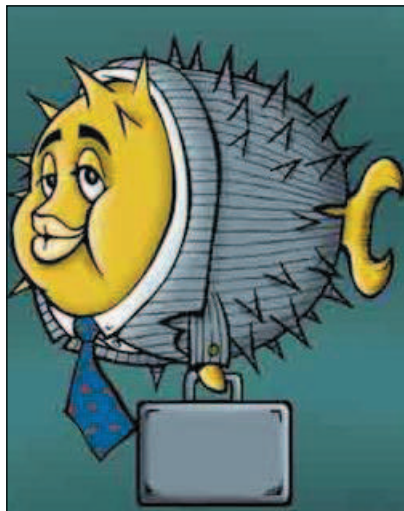
### Lin(Win)Modemek

Akinek programfüggő modemje van, valószínűleg tapasztalta, hogy az MS operációs rendszerekben kívül mással nem nagyon



tudja használni ezeket az eszközöket. Ennek az az oka, hogy a gyártók kispóroltak minden olyan alkatrészt, amit programmal lehet helyettesíteni, így szinte minden feladatot a számítógépünk proceszora végez el. Az ilyen modemektől féljünk, mint a jégveréstől, ha mégis ilyen eszközt kell életre keltenünk, az alábbi címek segítségünkre lehetnek:

- ➔ <http://www.linmodems.org>
- ➔ <http://linuxdoc.org/HOWTO/Winmodems-and-Linux-HOWTO.html>



### OpenBSD 2.9

Június elsején jelent meg az OpenBSD 2.9-es változata. Ebben már a legújabb 2.9-es OpenSSH változat van, ami már rendelkezik SFTP-vel (az FTP-t hivatott felváltani, ami az adatokat és a jelszavakat is kódolatlanul küldi el). Több mint ezer előre elkészített csomagból választhatnak a felhasználók, ezek közül megtalálhatók az alaprendszerben: az XFree86 4.0.3, a gcc 2.95.3, a Perl 5.6.0, az Apache 1.3.19 + mod\_ssl + OpenSSL 0.9.6 + DSO támogatás, a Sendmail 8.11.3, és az OpenSSH 2.9. Sokat javítottak az alkatrészek támogatásán, például az USB támogatás már alapbeállításként engedélyezett, és természetesen a biztonságon is sokat javítottak.

➔ [www.openbsd.org](http://www.openbsd.org)

### Debian

Akik frissebb programokra vágnak, mint amilyenek a megbízható Debian-változatban vannak, de nem akarnak áttérni a testing vagy unstable változatra, azoknak jöhet jól néhány nem hivatalos, de annál hasznosabb apt-forrás, amit az alábbi címen lehet elérni:

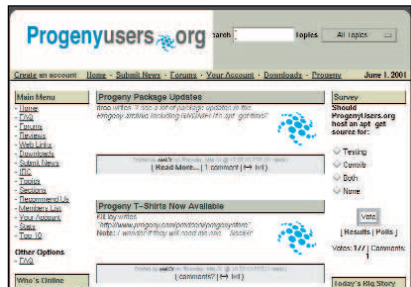
➔ <http://www.internatinf.org/bortzmeyer/debian/apt-sources/>

Ezekhez a csomagokhoz nincs hivatalos támogatás, ha valami nem működik jól a rendszerünkben, akkor segítséget elsősorban a csomag készítőjétől kérhetünk.

### ProgenyUsers

Ha az e havi CD-mellékletünkön lévő Progeny Debian telepítik, és ellátogatnak az alábbi címre, számos ötletet és segítséget kaphatnak. Hogyan telepítsék a rendszert, miként frissítsék a csomagokat, illetve mi módon állítsák be őket stb.

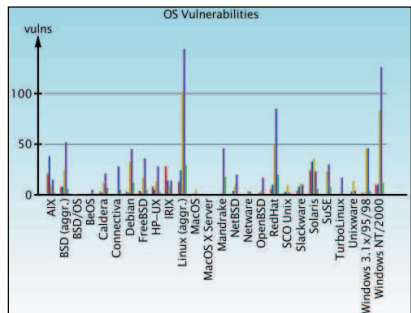
➔ <http://www.progenyusers.org/>



### Operációs rendszerek sebezhetősége

A <http://www.shopip.com/chart.htm> címen sok operációs rendszert hasonlítanak össze, „száraz” tények alapján. Az itt található hibalista élt 2000-ben a Microsoft Windows NT/2000 vezet, de találhatunk itt Linuxot, Solarist, BeOS-t, BSD-ket stb. Maga a ShopIP cég biztonságos e-bussines rendszerek építésével foglalkozik, ezekről is kaphatunk tájékoztatást a honlapon.

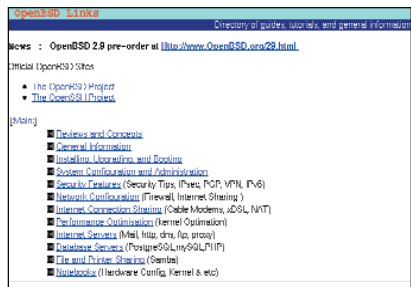
➔ <http://www.shopip.com>



### InfoBSD

Az InfoBSD nem egy újabb BSD változatot takar, hanem egy az OpenBSD-hez tartozó címtárat. Az innen elérhető dokumentumok és oldalak hasznosak lehetnek a rendszer beállításában és finomhangolásában jártas vagy járatlan felhasználóknak egyaránt.

➔ [www.infobsd.org](http://www.infobsd.org)



## Szerelmes levél a nagy Tuxtól a nagy Pénzeszsákhöz



A nagyfiúk közül senki sem indult be annyira a nagy Tuxra, mint a Kék óriás. Mindenki tud a milliárdokról, amit az IBM szeretne rá költeni. Hallottunk a kiszolgáló-üzletág átszervezéséről és a kutatásfejlesztés elkö-

telezettségéről. Láttuk, amint a LinuxWorld Expón babzsákszékeket görgettek ki a szakik számára. De ez még mind semmi. Hatalmas fellángolásról van szó: abból a fajtából, mely megköveteli az érzelmek felszabadítását. Mivel azonban a nagy cégek nem tudnak ölelni és csókolni (ez sohasem jutna túl a jogi osztályon), az IBM másképpen fejezi ki érzelmeit: hirdet. Az új reklámhadjárat jelmondata: „béke, szeretet és Linux”. Egy kérdező-felelek típusú Flash szerelmes levéllel indít a webhely. Így néz ki a nyitóoldal:

**„Mi a béke, a szeretet és a Linux?**  
Felhívás az erők egyesítésére.”

Ez az IBM érdeklődésének egyértelmű, lelkes összefoglalása a Linuxszal és a linuxos közösség támogatásával kapcsolatosan.

**„Miért támogatja az IBM a Linuxot?”**

Mert csodáljuk, hiszünk benne, szükségünk van rá és jó az ügyfeleinknek.

Az IBM jövőképe: segíteni kell a jövő technikai hátterének felépítésében – a haladás irányvonala a rugalmas e-kereskedelem és az egyesített, értelemmel felruházott gépek létrehozása. A látomás bonyolultsága, és igényei megdöbbentők és mindent elsöprők. Alaposan megvizsgáltuk, hogy mire lesz szükségünk ezek megvalósításához, és két dolog vált világossá: először is soha sem érhetjük el célkitűzéseinket a Linuxhoz hasonló, nyílt szabványok széles körű elfogadása nélkül. Másodszor a feladat annyira bonyolult, hogy az IBM egymaga

egyszerűen nem tud vele megbirkózni. Bár néhányan még mindig mást sugallnak, erre egyedül egyetlen cég sem képes. Csak a széles körű szakmai közösség együttes erőfeszítése teszi ezt lehetővé. A Linux-mozgalom összefoghatja és vezetheti az erre irányuló erőfeszítéseket.”

Ezt üzeni az IBM a természetükből adódóan hitlenkedőknek (olvasóink között is akadnak jó néhányan):

**„Ez az együttműködésed most komoly, vagy csak boldognak szeretnének látni bennünket?”**

Komoly. Véresen komoly. Aki azt hiszi, hogy csak a levegőbe beszélünk, elsiklik a lényeg felett.

„Az IBM évente ötmilliárd dollárt költ kutatásra és fejlesztésre, ebből egymilliárdot a Linuxra szánunk. De ez mind semmi ahhoz képest, amit a linuxos közösség magától létrehoz majd.

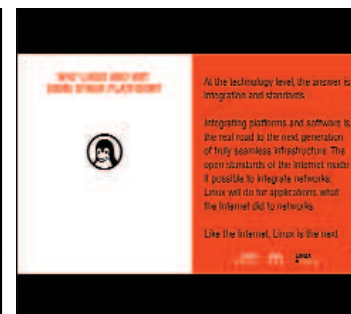
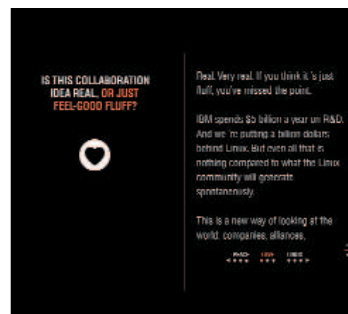
Ez egy új világszemlélet: cégek, szövetségek, társulások és egyének dolgoznak együtt, mindenki a saját szaktudását adja, hogy létrehozhassunk egy „erőteljes, párhuzamosan gondolkozó” erőt, ami mérhetetlenül hatalmasabb minden cégnél, az IBM-et is beleértve.”

Mint mindig, a virágokat nyújtó szerető most is elkövetett néhány apróbb hibát. Például úgy tűnik, hogy az IBM nem tudja, hogy a linuxos közösség érzékeny a GIF-ekre. Ahhoz,

hogy elérjük ezt a szerelmes levelet (☞ <http://www-1.ibm.com/servers/eserver/linux/passport.swf>), az IBM linuxos oldalát böngésző olvasónak át kell kattintania seregnyi GIF-en. Ha ezt minden ellenérzetünk dacára leküzdöttük, sokan közülünk akkor sem lesznek elégedettek, ha elérik az oldalt, mert nem tudnak Flasht lejátszani.

Néhány jó tanács az udvarlónak: a webes marketingre költött pénz töredékével támogathatná, mondjuk a Nyílt Forrású Kezdeményezést (☞ <http://www.opensource.org/>). Szükségük van rá, és segítségükre lenne számos olyan célkitűzés megvalósításában, amivel kérkedtek.

Doc Searls



**A POP forgalom titkosítása**  
Titkosíthatod a teljes POP forgalmat az ügyfélgép és a kiszolgáló között, ha a fetchmail ssh csatornán „bújtatod át”. Bővebb leírás a Linuxvilág 2000. novemberi számában, illetve az alábbi oldalon.  
<http://www.fetchmail.org/>

**Találás kérdés**  
Mit kapsz, ha keresztezel egy pentiumos gépet egy tudományos ösztöndíjjal? Egy örült tudóst.

## Az IBM termékbemutató körúton

„Az IBM szakemberei öt helyszínen előadások, esettanulmányok, valamint személyes tanácsadás keretében mutatták be a gyakorlatban az e-üzlet működését...”

Az IBM Magyarországi Kft. „Akcióban az e-business roadshow” címmel rendezvénysorozatot szervezett négy vidéki nagyvárosban (Debrecenben, Győrben, Pécsen, Szegeden) és Budapesten. A rendezvények célja, hogy az „e-business” fogalmát, illetve az IBM e-üzlethez kapcsolódó termékeit és szolgáltatásait széles körben népszerűsítsék. Az IBM szakemberei az öt helyszínen előadások, esettanulmányok, valamint személyes tanácsadás keretében mutatták be a gyakorlatban az e-üzlet működését, és igyekeztek átfogó képet nyújtani a legújabb irányzatokról, illetve a cég termékkínálatáról. A szakmai előadások mellett bemutatták az új, elektronikus üzleti célokra szánt xSeries kiszolgálóikat, igaz ezeken még Windows futott, de a szakemberek szerint a Linux sem okoz gondot. Az IBM a többi óriáshoz képest korábban nyitott a linuxos világ felé, ezt azzal is bizonyította, hogy a rendezvényeken lehetőséget teremtett a Linux, a Linux-felhasználók Magyarországi Egyesülete, valamint lapunk bemutatására is. Az IBM PC-k és a Linux jó együttműködési képességének érzékeltetésére a szép számú érdeklődő egy Mandrake Linux 7.2-vel felvértezett NetVista gépen ismerkedhetett „a másik világgal”. A számítógépet főleg otthon és irodai célokra használók között nagy sikert arattak a szabad programok sorait erősítő irodai alkalmazások: az OpenOffice, az AbiWord, illetve a KOffice. A feltett kérdések is főleg ezt a fonalat követték, de sokan érdeklődtek a linuxos tűzfalak, a levelező- és a fájlkiszolgálók beüzemelésével kapcsolatban is. Az utóbbi években egyre többen és többit beszéltek az e-üzlet térhódításáról. Magyarországon elsőként az IBM által 1998 őszén rendezett „e-business Akadémián” esett szó róla egy félnapos rendezvény keretében. Az azóta eltelt időben jelentése, tartalma ismertté vált a kormányzat és a gazdaság résztvevői számára, de a mindennapok legkülönbözőbb területein is. Mint ahogy azt a rendezvénysorozaton tapasztaltuk, még mindig rengeteg kérdés vár megválaszolásra a gazdaságot alapjaiban megváltoztató e-üzlettel. A körút beváltotta a hozzá fűzött reményeket, hiszen számos olyan érdeklődőt sikerült elérni, akik többit szerettek volna megtudni e témáról és az IBM kapcsolódó kínálatáról.

### Az IBM által bemutatott termékek

Az óriás a rendezvényen azokat a termékeket és szolgáltatásokat kínálta, amelyek egymással, vagy az IBM-éhez hasonló megoldásokkal, illetve IBM üzleti partnerek és más programgyártók kínálatával együtt segítik a cégeket az új terület meghódításában.

### e-marketing programok

A fontos e-marketing programok már a nyílt operációs rendszereken is használhatók. Az IBM bejelentette új, Linux-alapú programját, mellyel a vállalatok személyre szabott marketingüzeneteket juttathatnak el ügyfeleikhez az Interneten keresztül. Ez a lépés az IBM legújabb kezdeményezése azon cél eléréséhez, hogy a teljes e-üzlet program-, kiszolgáló- és szolgáltatási kínálatát kiterjessze Linuxra is.

Az IBM WebSphere Personalization programmal a vállalatok személyre szabott weblapokat készíthetnek, amelyekkel a webhely használatára lehet csábítani a látogatókat, és elég vonzó ahhoz, hogy meg is tartsák őket visszatérőkként. A termék segítségével megismerhetik, és előre megjósolhatják a felhasználói szokásokat. „Az IBM WebSphere Personalization segítségével gyorsan elkészíthetők, telepíthetők és frissíthetők a személyre szabott adatok különféle felületeken, többek között Linuxon, amely az iparág leggyorsabb ütemben terjedő operációs rendszere.” – idézték *Ed Harbourt*, az IBM WebSphere felületért felelős marketingigazgatóját. A WebSphere valójában egy teljes termékcsalád, amely a következő eszközöket tartalmazza:

- a WebSphere Studio webfejlesztő eszközöket,
- a WebSphere Personalization eszközt, mely üzleti szabályok alkalmazásával teszi lehetővé a webes tartalom látogatók érdeklődésének megfelelő testreszabását,
- a Macromedia LikeMinds együttműködési szűrési módszert,
- a WebSphere Site Analyzert, amellyel elemezhető, hogy mennyire vonzó a webhely tartalma, és így egyszerűsíthető a webhely testreszabása,
- valamint a WebSphere Application Server v3.5 rendszert, amellyel olyan webes környezet készíthető, mely a mennyiségbeli változásokat gyorsan, a fontos tranzakciós adatok elvesztése nélkül képes követni. A vállalkozás növekedésével a WebSphere lehetővé teszi az új, elektronikus kereskedelmi alkalmazások és a meglévő, örökölt üzleti rendszerek egyesítését is.

A WebSphere Application Serverek jelentik a WebSphere felület alapját. Ez olyan internetes háttérprogram, amely lehetővé teszi az e-üzleti alkalmazások kifejlesztését, üzemeltetését, futtatását és egyesítését. A WebSphere Application Server Standard és Advanced Edition v3.5 lehetővé teszi testreszabott e-üzleti alkalmazások telepítését Linuxra, a webes közzétételtől a vállalatközi elektronikus piacokra jellemző nagy tranzakciószámú alkalmazásokig.

A WebSphere Personalization előző, 3.5-ös változata egy termékben tartalmazta a szabályalapú személyre szabást és a javaslatkészítő motort, amely a Macromedia cég LikeMinds együttműködési szűrési módszerét alkalmazta. A WebSphere Personalization legújabb, 3.5.2-es változatában a javaslatkészítő motor már AIX felületen is futtatható, tovább szélesítve ezzel a használható operációs rendszerek körét.

A Personalization 3.5.2-es változatában a szabályalapú személyre szabási szolgáltatás is tovább bővült: a folyamat arra is alapozható, hogy mi történik, miközben egy látogató körbenéz egy webhelyen. Például megfigyelhető, hogy a látogató mit nézett meg a helyen, vagy mit tett a bevásárlókosarába. Így a rendszer jobban követheti az egyes látogatók érdeklődését, és pontosabban kiválasztható a látogatók számára hasznos és érdekes tartalom. Ez a szolgáltatás elősegíti a látogatók elcsábítását és megtartását.



A WebSphere felület átfogó megoldás a középső réteg (middleware) megvalósításához, mellyel a vállalatok következő nemzedékbeli e-üzleti alkalmazásokat fejleszhetnek, telepíthetnek és építhetnek, ideértve a vállalkozási (B2B) elektronikus piacokra jellemző nagyszámú tranzakciók kezelésére alkalmas programokat is. A WebSphere felület az üzleti alkalmazásokat támogatja, az egyszerű webes közzétételtől a nagyvállalati szintű tranzakció-feldolgozásig. A WebSphere segítségével átfogóan kezelhetjük cégünk üzleti kapcsolatait is, legyen szó a partnerekkel, az ügyfelekkel, vagy akár saját alkalmazottainkkal történő kapcsolattartást.

### Elérhetőség

A WebSphere Personalization for Multiplatforms 3.5.2-es változata 2001. január 26-tól vásárolható meg Linux, AIX, HP-UX, Solaris, Windows 2000 és Windows NT operációs rendszerekhez. A program szabály- és erőforrás-kezelő motorja Linux, AIX, HP-UX, Solaris, Windows 2000 és Windows NT operációs rendszeren is

futtatható. A javaslatkészítő motor AIX, Solaris, Windows 2000 és Windows NT operációs rendszeren használható. A WebSphere Personalization OS/400 környezetben futtatható változata – amely biztosítani fogja a javaslatkészítő motor használatát AIX felületen is – a tervek szerint a későbbiek folyamán fog megjelenni. A következő Linux-változatok használhatók: RedHat 6.2, Caldera 2.3 e-Server, TurboLinux 6.0 és SuSE 6.4. A WebSphere Personalization for Multiplatforms v3.5.2 támogatja a SuSE 6.4 használatát S/390 felületen. A programcsomaggal kapcsolatban bővebb felvilágosítást a [software@hu.ibm.com](mailto:software@hu.ibm.com) címen, illetve az IBM Magyarország központi telefonszámán (382-5500) lehet kérni.

További adatok a WebSphere programfelületről

➔ <http://www.ibm.com/websphere> és a

➔ <http://www.ibm.com/software/webervers/personalization> címen találhatók.

## Rövid hírek az LME életéből

### Újabb Pingvin-füzetek pályázat

Az egyesület új pályázatot hirdet azon Pingvin-füzetek megírására, melyekre nem érkezett érvényes pályamunka az előző fordulóban. A feltételek teljes egészében megegyeznek az eredeti pályázati kiírással. Az új leadási határidő 2001. június 18. (hétfő), az elbírálás határideje pedig 2001. június 22. A pályázati űrlapok letölthetők a [palyazat.lme.hu](http://palyazat.lme.hu) oldalról.

Pályázható témák:

- A számítástechnika alapjai
- Grafikus kezelőrendszerek (KDE)
- Grafikus kezelőrendszerek (Gnome)
- GNU/Linux hálózat beállítása
- Levelezőrendszerek
- Egyéb hálózati alkalmazások
- Szövegszerkesztő rendszerek I. – OpenOffice
- Szövegszerkesztő rendszerek I. – Abiword
- Szövegszerkesztő rendszerek I. – KOffice
- Szövegszerkesztő rendszerek II. – Openoffice
- Szövegszerkesztő rendszerek II. – Abiword
- Szövegszerkesztő rendszerek II. – Koffice
- Táblázatkezelő rendszerek I. – KOffice
- Táblázatkezelő rendszerek II. – Koffice
- Prezentációkészítés – OpenOffice
- Prezentációkészítés – MagicPoint
- Prezentációkészítés – KOffice
- Számítógépes vektorgrafika alapjai – OpenOffice
- Számítógépes vektorgrafika alapjai – XFig, Dia
- Számítógépes vektorgrafika alapjai – KOffice
- Számítógépes bitmapgrafika alapjai – GIMP
- Számítógépes bitmapgrafika alapjai – XPaint
- A GNU/Linux alapparancsai
- Segédprogramok használata

### III. GNU/Linux Szakmai Konferencia

A Linux-felhasználók Magyarországi Egyesülete 2001. szeptember 15-én (szombaton) tartja III. szakmai tanácskozását a Közép-Európai Egyetem (CEU) Konferencia Központjában, neves előadók részvételével. A rendezvény díszvendége *Richard M. Stallmann* az Egyesült Államokból, a szabad programok guruja, a GNU atyja, a Free Software Foundation alapítója.

A tanácskozás területén kiállítás és vásár is helyet kap.

A konferencia fő témája a szabad programok és a GNU/Linux lesz.

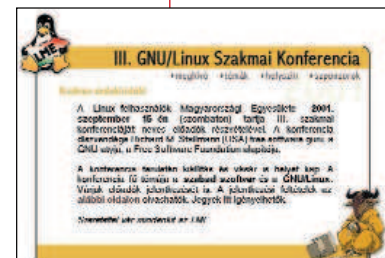
Az előadók jelentkezését a következő témákban várják:

- GNU/Linux rendszerfelügyelet,
- biztonság,
- programozás GNU/Linux-rendszereken,
- multimédia és grafika,
- magyartítás,
- GNU/Linux a cégeknél.

A jelentkezőktől A4-es terjedelmű kivonatokat várnak az [abstract@lme.linux.hu](mailto:abstract@lme.linux.hu) címre. A beérkező vázlatok alapján az előadókat bizottság választja ki.

Leadási határidő: 2001. június 4. (hétfő).

A bírálóbizottság döntéséről minden jelentkezőt levélben értesítenek, legkésőbb 2001. június 18-ig, így az előadónak az előadások elkészítésére két hónap áll rendelkezésükre. A tanácskozás előadásaiból idén is kiadvány készül, melyhez a kiválasztott előadóktól legkésőbb augusztus 10-ig várják előadásuk teljes anyagát, lehetőleg Latex formátumban. Erről bővebb tájékoztatás a <http://lists.linux.hu/Lists-Archives/lme/2000-August/002494.html> oldalon olvasható.





## Linux irányítja San Francisco közlekedését

A Bay Area Rapid Transit (BART) rendszer legfontosabb feladata, hogy a San Francisco-öböl környékének tömegközlekedését minél hatékonyabban működtesse. Központja Oakland szívében, tíz méterrel a föld alatt



### Néhány tény a Linuxról...

A Linux 2.4.0 CREDITS fájljában 375 név szerepel. Az operációs rendszer 13 számítógéprendszert támogat, valamint 1471 eszközmeghajtót tartalmaz.

(A Linux 2.4.0 forráskódja alapján)

található. Az irányítóterem falain harminc vetítő jeleníti meg a BART vonatirányítási térképeit és az elektromos hálózatot. A vetítőrendszert NCD X terminálokról vezérlik, melyek egy Tandem S4000-hez kapcsolódnak (ez utóbbi tárolja a rendszer állapotát). A vezérlők Sun munkaállomásokkal kezelik a rendszert és továbbítják az adatokat a Tandemnek. Mivel a BART-ot a San Francisco-i repülőtérrel is összekötötték, ezért a BART vezérlőtermének elektromos hálózatának adatait megjelenítő rendszert is frissíteni kellett, és ezt 2000. júliusáig végre kellett hajtaniuk. A hatékonyabb fejlesztés érdekében úgy döntöttünk, hogy a rendszervezérléssel kapcsolatos emberi tényezőknél is változtatunk. A rendszert eredetileg több, nagy felbontású Tektronix monitorhoz tervezték, így a tapasztalt kezelőszemélyzet egy szemvillanás alatt felmérhette a pillanatnyi állapotot. Amikor a Tektronix-kijelzőket Sun munkaállomások váltották fel, a szolgáltatások köre bővült ugyan, de a képet a kevés hely miatt görgetni kellett, hiszen az ablakok eltakarták egymást. Ezt a személyzet egyáltalán nem kedvelte, hiszen így nem láthatták át azonnal a rendszert.

A Sony nemrégiben hozta ki a 24 colos monitorait, melyek 16:9-es képaránya a BART-nál végzett munkához tökéletesen megfelelt. A Sun Solaris 2.4-es operációs rendszer sajnos nem támogatja e monitorok nagyfelbontású üzemmódjait, így más megoldás után kellett néznünk. Hamar rájöttünk, hogy a Linux lesz a megmentőnk. A monitorok Linux alatti használatához mindössze az XFree86 beállításfájljait kellett módosítanunk. Erre építve Linuxot futtató, hagyományos PC-rendszert helyeztünk

üzembe, így tökéletesen kihasználható a Sony monitorok nagyfelbontású üzemmódja.

A Qt vagy a GTK+ helyett a Motif elemkészletet használtuk, megbízhatósága miatt. Ezután a grafikus felület fejlesztéséhez használható programokat kutattuk fel. Végül a Loox Software (<http://www.loox.com/>) cég LOOX++ és az ICS vállalkozás (<http://www.ics.com/>) Builder Xcessory Pro nevű programja mellett döntöttünk (ez utóbbi a Linux Journal 1999. évi díjazottja).

A LOOX++ egyszerűsítette elektromos rendszerünk megjelenítését, a Builder Xcessory Pro programcsomaggal pedig áttekinthető grafikus felhasználói felületet hoztunk létre.

Az új elektromos program fejlesztését 1999. decemberében kezdtük, a munka három hónapot vett igénybe, így a San Francisco-i repülőtér-bővítés határidejére elkészültünk vele. A központban jelenleg minden munkaállomáson Linux fut.



[www.loox.com/home/index.html](http://www.loox.com/home/index.html)

Ha arra gondolunk, hogy akár a gépi környezet meghibásodása, akár egy programhiba milyen beláthatatlan következményű balesetet okozhat, érthető, hogy a Linux választása alapvető kérdéssé vált. Az új elektromos rendszerrel azonban komoly eredményeket mutattunk fel, így megkaptuk az engedélyt a második átépítési szakasz megkezdésére is, melynek során a vonatirányítást is Linuxra állítjuk át. A közönséges PC-alkatrészek használata, az operációs rendszer ingyenes volta és az eszközök folyamatos, gyors fejlődése 15–20 százalékos megtakarítást eredményezett. A linuxos környezet teljesítménye azóta is tökéletes, azzal együtt, hogy a rendszer folyamatosan üzemel.

Gary A. Messenbrink  
számítógépes rendszermérnök a BART-nál.

Frank Ruffa  
a BART fejlesztési részlegének szakmai vezetője.

Fényképek: Bay Area Rapid Transit

## A nagy durranás, avagy mit hozott a Linux az IBM háza tájára

Tájékoztatóanyag érkezett szerkesztőségünkbe az IBM Magyarországtól a webkiszolgálókkal kapcsolatos bejelentéseikről. Munkatársunk ennek járt utána. Beszélgetőtársai voltak **Auer Zsolt** termékmenedzser, az IBM Unix Systemstől és **Dávid Miklós** termékmenedzser, az iSeries részlegtől.

**Mészáros Ferenc:** Mit keres a Linux az IBM kínálatában, jelesül az RS/6000-eseken, hiszen a cég nem csak alkatrészgyártó?

**Auer Zsolt:** Valóban, az IBM nem csak alkatrészgyártó, már amennyire alkatrész kategóriába beleférhet egy számítógép is, de talán egy mondat erejéig érdemes volna kitérni arra, hogy az IBM mely felületekkel gazdagította a számítógépek piacát. Időrendi sorrendben a Mainframe volt a legelső 1964-ben, amivel gyakorlatilag az IBM megalapozta a kereskedelmi számítógépek piacát. A 80-as években ismerkedhetett meg a világ az IBM PC fogalmával, ami azt hiszem, hogy nem szorul bővebb magyarázatra. Ezt a követte megbízhatóságáról elhíresült AS/400-as gépcsalád, majd negyediként az RS/6000-est jelentettük be 1990-ben, mint az IBM unixos rendszerét, így ez a vonal is már tizenegy éves múltra tekinthet vissza. Azt hiszem a kérdés kissé kihívó volt, hiszen az említett gépcsaládok mindegyike saját, jól bevált operációs rendszerrel rendelkezik. Hogy kerül ide a Linux? Nos, a nyitottság irányába történő elmozdulásra a cég tulajdonát képező operációs rendszerek fejlesztési stratégiájában már évek óta szemmel látható jelek tapasztalhatók. Ami a Linuxot illeti, a linuxos piac térhódítása olyan mértékű, ami mellett nem mehet el még az IBM sem, így a Linuxot stratégiai irányként kezeljük, de hangsúlyoznám, hogy a meglévő felületek mellett és nem feltétlenül azok helyett. Az a célunk, hogy közeledjünk a Linux piacrésze felé, és a Linux-felhasználók közeledjenek az IBM felé. Ennek kézenfekvő megoldásaként látszott az az elgondolás, hogy a meglévő vasainkon tegyük lehetővé a Linux futtatását.

**M. F.:** Két új középosztályos IBM eServer Unix RS/6000-es modelljüket – a pSeries 620 Model 6F1-et és a pSeries 660 Model 6H1-et – akár hat 668 MHz-es processzorral is fel lehet szerelni. Miért pont a 668 MHz lett a határ?

**A. Zs.:** Ez fejlesztői és marketingdöntés volt. Nyilván nem ez a felső határ, hiszen órajel szempontjából már két éve felmutattunk gigahertzes tartományban működő processzorokkal üzemelő gépeket is. Ennek túllépése elméletben és gyakorlatban sem jelent gondot. A termékcsalád fejlesztésében ez a modell az egyik lépcsőfok, ezt jelentettük be, és a piacon nagyon versenyképes modellnek bizonyult. Felesleges az órajeleket túlfeszíteni, ha a piac nagyobb órajelű modellt – természetesen, ez drágább – nem tud megfizetni, és a kisebb órajellel is versenyképesek vagyunk.

**M. F.:** A Linux hogyan kapcsolódik az IBM-hez? Tétélezzük fel, hogy valaki megszerzi az IBM gépi környezet leírását és saját Linuxot telepít rá. Hogyan történik a valóságban?

**A. Zs.:** A dolog majdnem ilyen egyszerű. Mint már említettem, mind a négy kiszolgáló osztályunkon – S/390, AS/400, RS/6000, Netfinity – valamilyen módon támogatjuk a Linuxot, ami a Linux ismeretén túlmenően némi

rendszerismeretet is megkövetel azoktól, akik szeretnének megismerkedni az IBM felületei által nyújtott előnyökkel linuxos környezetben. Természetesen részletes leírással, használati útmutatókkal és programozói eszközökkel, segédletekkel szeretnénk megkönnyíteni az ismerkedést. Megjegyzem, a négy gépcsalád már nem a régi nevén szerepel a piacon, a tavalyi terméknevváltás kapcsán az egységes IBM eServer elnevezést alakítottuk ki. Ekkortól a zSeries, a pSeries, az iSeries, az xSeries névvel jelöljük négy gépcsaládunkat – az előbbi sorrendnek megfelelően –, amelyek mind alkalmasak valamilyen módon linuxos alkalmazások futtatására, vagy a Linux natív módon történő futtatására is. Ezzel próbáljuk a linuxos felhasználókat az IBM termékeinek megismerésére ösztönözni.

**M. F.:** Továbbá a gépek vásárlására?

**A. Zs.:** Amennyiben ez előnyt jelent a felhasználók számára is, miért ne? Bár itt nemcsak a vasakról lehet szó, hiszen az IBM nemcsak gépeket gyárt, hanem programokat is, valamint szolgáltatások széles körét kínálja. Mindazt, amit az információtechnológiában létrehozunk – több ezer termékre gondolok –, valamilyen úton-módon megpróbáljuk a linuxos világhoz kapcsolni. Önmagában a Linux sem – ahogy egyetlen operációs rendszer sem – életképes, ehhez kellene az alkalmazások, a köztes programok. Kevesen gondolták, hogy a Linux ilyen népszerű lesz, és nem csak az egyetemi környezetben, illetve Webfarmokon és az internetszolgáltatók körében. Ha hihetünk a piacelemző cégek jóslatainak, akkor pár éven belül már a rendkívül fontos üzleti alkalmazások „alatt” is megtalálhatjuk a Linuxot.

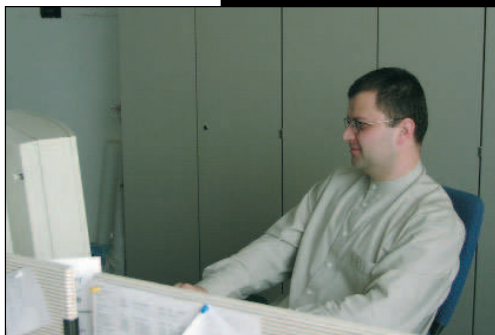
A felhasználástörténetet tekintve, ez a népszerűség annak is köszönhető, hogy a komolyabb alkalmazások és alkalmazásszállítók szintén elkötelezték magukat, hogy átültetik az alkalmazásokat Linuxra. Nyilván az IBM számos alkalmazást és köztes programot is gyárt, híres például a Lotus, a Tivoli termékcsalád vagy a DB2 és a Websphere. Ez kiemelten fontos az e-üzleti alkalmazások fejlesztésében, és a Java sem maradhat ki. Az említett alkalmazásokat, a gépcsaládokat és az ehhez társuló szolgáltatási hátteret – ami egyébként a korábbi, meglévő rendszerek üzemeltetése és támogatása kapcsán már létezik – természetesen igénybe vesszük a linuxos felhasználói tábor támogatása során is.

**M. F.:** Ha jól értelmezem a linuxos vásárló, ha ilyen kiszolgálót szeretne venni magának, akkor a kézen-közön megtalálható Linux operációs rendszert egyből föltelepítheti, vagy kap hozzá valamilyen segítséget, programcsomagot, ami átsegíti a kezdeti nehézségeken?

**A. Zs.:** Mindenképpen kap hozzá programcsomagot, illetve föl is telepítheti. Az IBM szorosabb kapcsolatban áll néhány főbb változattal: a Calderával, a RedHattal, a



Auer Zsolt  
termékmenedzser



Dávid Miklós  
termékmenedzser

SuSE-vel, illetve a Turbo-Linuxszal. Van azonban külön fejlesztői csapatunk is, akik a Linux rendszerrel kapcsolatos fejlesztéseket végzik, másrészt különböző anyagokkal segítik a linuxos tábornak. Ami nálunk használatos, és bevált a rendszereinkben,

azokat próbáljuk megvalósítani linuxos környezetben is.

**M. F.:** És ez mind a négy gépcsaládra igaz?

**A. Zs.:** Igen, az a cél, hogy mind a négy felületen egységesen támogassuk a Linuxot.

**M. F.:** Jelenleg hol tartunk ebben a fejlesztésben?

**A. Zs.:** Az inteles felületen korábban sem volt gond, hiszen a szerkezetet a Linux a kezdetek óta támogatja, az xSeries gépekkel tehát minden rendben van. A kereskedelmi forgalomban már sikereket ért el a Mainframe vagy újabb nevén zSeries. Ennél a típusnál létezik olyan változat, amely akár harmincezer (!!!) Linux rendszermagot képes egyszerre futtatni. Ilyenre hamarosan lesz példa Magyarországon is (jelenleg próbaüzem alatt áll néhány rendszer). Az iSeries, azaz az AS400-as esetében is rendelkezünk linuxos támogatással, amelyet majd munkatársam ismertet. Utoljára hagytam a pSeries azaz az RS/6000-es gépcsaládot, ahol a linuxos alkalmazások támogatásának két módját is biztosítjuk. Egyrészt natív módon támogatják az említett Linux-változatok a gépcsaládot – pillanatnyilag nem minden tagját, de folyamatosan bővül ez a géppark, ahogy az IBM és a változatok közösen fejlesztgetik a rendszermagot –, ez az egyik irányvonal. A másik pedig az AIX 5L (az IBM unixos operációs rendszer) 5.1-es változat, mely linuxos irányultsággal bír.

**M. F.:** Itt, gondolom, az L-betű jelöli ezt. Mit értsünk alatta?

**A. Zs.:** Természetesen adódik a kérdés mit is jelent ez a linuxos irányultság? Ez annyit tesz, hogy linuxos alkalmazásokat lehet futtatni AIX operációs rendszeren, csupán újrafordítás szükségeltetik. Nyilván ez mindenképpen szükséges, mert nem Intel-felület, hanem IBM POWER processzor található a pSeries gépekben. Az AIX 5L biztosítja a felületet, melyen a linuxos alkalmazások futnak a pSeries összes modelljén. Itt az újrafordítási környezetet ki kell alakítani, ezt tartalmazza az AIX Toolbox for Linux Applications néven terjesztett ingyenes csomag. A csomag letölthető a <http://www.ibm.com/linux> címről is, de az AIX 5L operációs rendszerhez is hozzátartozik. A Toolbox fejlesztői és fordítókörnyezetet takar, amivel hatékonyan, egyszerűen és gyorsan lehet a meglévő linuxos alkalmazásokat illeszteni az AIX operációs rendszerhez. Az AIX 5L linuxos alkalmazási környezet előnyei a következők:

- *nem jár teljesítménycsökkenéssel*, mert itt nem további rétegről, nem emulációról van szó, hanem csak egyszerűen az AIX operációs rendszert használjuk. A megfelelő könyvtárhívások nem az AIX library hívá-

sokat használják, hanem a beépített GNU könyvtárakat, vagy ami a linuxos alkalmazáshoz szükséges.

- A linuxos rendszerek méretezhetőségének felső határa inteles felületen pillanatnyilag négy processzor, illetve dolgoznak már a nyolcprocesszoros rendszermagokon. Ez azonban – mint tudjuk – nem a világ teteje. Az AIX 5L használatával a gépcsalád 24 processzoros gépén is futtatható lesz bármely linuxos alkalmazás, azon a 24 processzoros rendszeren (p680), mely pillanatnyilag a világ leggyorsabb Oracle-alapú tranzakció-feldolgozó rendszere (☞ [www.tpc.org](http://www.tpc.org)). Ha olyan teljesítményigény van, ahol ez indokolt, akkor itt a megoldás.
- A Power processzorcsaládnak hihetetlen erős a lebegőpontos teljesítménye. Egy olyan területen (például a High Performances Computing területén), ahol már a Linux elég fontos szerepet tölt be a világban, ott ez komoly követelmény. Szűk keresztmetszet egyébként, hogy egy négyprocesszoros Intel-felület után az egész programszerkezetet át kell építeni párhuzamos környezetre, ahol működőképesnek és egy rendszeren belül lekezelhetőnek kell lennie. Ha 24 erős processzorban tudunk gondolkodni, akkor ez igen komoly erőforrás-növekedést jelent.
- Logikai tartományok használatára még nincs lehetőség a gépcsaládon, ezt is az év folyamán vezetjük be, és további üzenetnek szánjuk a linuxos felhasználói körnek.

Kiemelném még azt is, hogy a fűrtözési módszer szerint pillanatnyilag nem erőssége a linuxos világnak, talán mert a kiindulási piac nem nagy fontosságú alkalmazásokat futtató piac. De ahogy a Linux az üzleti alkalmazások felé halad, egyre fontosabbá válik a felhasználók számára, hogy megbízható rendszert kapjanak, azaz amikor leáll a rendszergép, akkor legyen valamilyen önműködő mentési rendszer. A Linux-változatok fejlesztői csapataival olyan közös fejlesztések zajlanak, melynek eredményeképpen hamarosan megjelenik a piacon ez a fűrtözési módszer. Az, hogy elérhetővé tesszük a Linux-felhasználók számára, óriási előny, hiszen a megbízhatóságot, a rendelkezésre állást jelenti, ami eddig hiányzott.

**M. F.:** Köszönöm szépen, és hadd forduljak most Dávid Miklós úrhoz. Milyen új fejlesztésekkel állt elő az AS/400-as, vagyis az iSeries családban az IBM?

**Dávid Miklós:** Az IBM április 23-án jelentette be az iSeries családban az új modelleket, és az új operációs rendszert, az OS/400 V5R1 kiadást. Így lehetővé vált a Linux futtatása ebben a kiszolgálócsaládban is. A bejelentésnek három lényeges része van. Az elsőben beszámoltunk az új gyártási módszerről: az új processzorok a legújabb rézalapú, Silicon-On-Insulator módszerrel készülnek. Ezt most már nemcsak a csúcsgépekben vehetik igénybe az IBM ügyfelei, hanem az alacsony kategóriában, azaz a kiszolgálócsalád legkisebb gépeinél is. Az iSeries minden gépében azonos módszerrel készített processzor van, így minden ügyfél ki tudja használni ezt a teljesítményt. A bejelentés második része az új operációs rendszerről szólt, a harmadik pedig a többféle megoldás futtatására képes eServer iSeriesről. Ez lehetővé teszi a gépek logikai felosztását és a géptartományokban a Linux futtatását.



**M. F.:** Milyen linuxos elérhetőséget teremtetek az AS/400-asok?

**D. M.:** Az új operációs rendszer megjelenésével lehetőség nyílik mind a régi, mind az új modelleken logikai tartományok segítségével Linux-alapú alkalmazások futtatására. Ehhez az AS/400-on több logikai tartományt kell létrehozni. Lesz egy vezérlőtartomány, ez az első tartomány, mely minden esetben OS/400 (az IBM AS/400-okhoz írt operációs rendszere). Ebből következik, akinek van iSeries AS/400-as gépe, annak van legalább egy OS/400-as operációs rendszere is, amely ingyen jár a géphez. Az összes többi tartományból választhat az ügyfél, hogy mit szeretne rajta futtatni: OS/400-ast másik tartományban, másik változatú OS/400-ast vagy Linuxot.

**M. F.:** A másik tartományban futás eredményez-e sebességsökkenést, illetve milyen gyorsaságot tesz lehetővé?

**D. M.:** Az AS/400-as gépeken a teljesítményt CPW-ben mérjük (Commercial Processing Workload). Már a kezdetektől fogva arra törekedtünk, hogy az AS/400-ast, vagyis az iSerieset kiegyensúlyozott felépítésű géppé alakítsuk, tehát itt nem a meghaztertek számítanak. Ebben ugyanazok a 668 MHz-es processzorok futnak, mint a pSeries gépekben. Itt azonban az a legfontosabb, hogy a többi alrendszer képes-e ennek megfelelő sebességgel továbbítani az adatokat a processzor legnagyobb kihasználtságához. Mivel a Linux most jelent meg az iSeriesen a teljesítménybeli különbségeket még csak vizsgálják. Meg lehet határozni, hogy a processzorteljesítményt milyen arányban osztja szét a felhasználó a különböző logikai tartományok között, és itt egy százalékos ugrások lehetnek a processzorteljesítményben, tehát a rendszer dinamikusan, újraindítás nélkül, százalékonként tudja hangolni a tartományok között a teljesítményt. Az új gépeken lehetőség van egy processzoron több (legfeljebb négy) logikai tartomány indítására, míg a régebbi modelleken egy kinevezett processzor szükséges egy tartomány futtatására.

**M. F.:** Az elhangzottakon kívül milyen fontos híreket osztanának meg olvasóinkkal?

**D. M.:** A linuxos rendszerekkel kapcsolatban még hadd mondjam el, hogy az iSeries (azaz az AS/400-as) gépeken elsőként a RedHat támogatását tervezzük, ezt követi a SuSE és a TurboLinux. Érdeklődés Magyarországon is van, például jelenleg is vannak már megrendelőink (május 10-én), pedig április 23-án volt a bejelentés és a teljes elérhetőséget csak május 25-ére ígértük. Már most rengeteg leendő ügyfél érdeklődik, hogy iSeriesen Linuxot futtassanak. **Auer Zsolt:** Küldenek néhány üzenetet a magyar linuxos tábornak. Az alkalmazásfejlesztéshez kínálunk ingyenes IBM Applications Developers Kitet, ami a beszélgetésünk elején elhangzott IBM-es alkalmazáskörnyezeteket támogatja, és annak a fejlesztését segíti elő linuxos környezetben, WebSphere és DB2, Lotus, illetve egyéb jól ismert vonalnak a linuxos alkalmazásfejlesztését támogatja, így Javáét is. Az IBM Magyarországon belül működő úgynevezett Solution Porting Centert nyitott. Ez arra hivatott, hogy a programfejlesztő cégek alkalmazásait bevizsgálják IBM-felületeken. Ezt pillanatnyilag xSeries, tehát Intel-alapú gépekkel szereltük fel. Két éve működik sikeresen. Ezt az ingyenes segítségnyújtást, támogatást hamarosan ki fogjuk bővíteni pSeries, RS/6000-es gépcsaládunkkal is. Itt a programokat bárki kipróbálhatja, ehhez csak időpontot kell kérni, sőt akár távolról is kipróbálhatja natív módon a linuxos alkalmazásait. Ezenkívül lehetőség nyílik az AIX 5L nyújtotta előnyökkel megismerkedni, ha valaki arra is kíváncsi, hogy működik Linux a pSeries osztályban.

*Az első nagy durranás a 80-as évek elején az volt, amikor a Commodore-ok, Atarik és Sinclair Spectrumok világában (az otthoni számítógépek világában) szorítóba lépett a Big Blue, létrehozva az IBM PC-t. Ki használ ma már otthon Commodore-t vagy Atarit? Húsz év telt el, és most a Big Blue letette szavazatát a Linux mellett, mégpedig úgy, hogy a PC-től a nagygépekig saját operációs rendszerei mellett, sőt helyett használja. Tíz év sem kell és... Szerintem ez lesz az új évezred számítástechnikájának első nagy durranása.*

## Ők mondták

A programozást és a programelemzést a kis elsősöknek kéne tanítani. Nem, inkább még előbb. *(Bob Frankston)*

Először azt hittük, hogy a PC egy számológép. Aztán rájöttünk, hogyan lehet a számokból ASCII karaktereket varázsolni, és akkor azt hittük, hogy írógép. Aztán föltaláltuk a grafikát, és azt hittük, hogy a PC egy tévé. Aztán előjöttünk a Webbel, és kiderült, hogy az egész egy reklámújság. *(Douglas Adams)*

A gépeléssel még senki nem keresett pénzt. *(Wade Hennesey)*

A Hold felszínét csillagászati baromságok fedik. *onyxruby (Kuro5hin.org)*

A Linux olyan, mint egy indián sátor: nincsenek ablakok, nincsenek kapuk, és belül egy apacs ül. *(Albert Arendsen)*

Gondolkodó számítógép... hm... az valami olyasmi lenne, mint az úszó hajó? *(Albert Arendsen)*

Nem bízhatunk a szabálytagadó emberekben. Gyorsan szabályok közé kell szorítanunk őket. A szervezet nem bízhat az egyénben; az egyénnek viszont bíznia kell a szervezetben. *(Ray Kroc -, a McDonald's alapítója)*

Nem hiszünk a királyokban, az elnökökben és a szavazásban. Úgy-ahogy megegyezünk, aztán a végén csak fut a program. *(David Clark)*

A kétszáz kiállítóból mintegy húsz jelezte a kiállítás útmutatójában, hogy „beágyazott rendszerű” termékeket kínál...

A Lineo cég pavilonja szokás szerint pezsgett, számos bemutatót láthattak az érdeklődők...

## A linuxos világ beágyazott oldala

Háromnapos barangolás után, miközben bejártam a kiállítás mintegy tízezer négyzetméternyi területét, ahol 200 cég jelent meg, és amelyet talán kétszázézer „pingvinista” tekintett meg, kétségtelenül sok szemrevaló Linuxot láttam.

Néhány napot hagyva a „leülepedésre”, pár általános észrevételre jutottam. Először is az idén több volt az öltöny, több a csillogó kiállítás és kevesebb volt az apró (9 m<sup>2</sup>-es) stand. Másodsor, az idén sokkal több kiállító jelent meg – a kiállítási terület a tavalyi New York-i rendezvény területének a kétszerese volt. Az idei kiállítás azt is sugallja, hogy a Linux serdülőkorba lépett. Sokkal jobban érződött egy érlelődő ipar levegője, mint a tavalyi New York-i rendezvényen, hiszen olyan nagy cégek is kiállítottak, mint az IBM, az Intel, a Compaq, a Sun, az SGI és a HP. Végül, ezen a Linux vilákiállításon a beágyazott rendszerek valóban befutottak. Majdnem mindenütt közös téma volt a Linux növekvő jelentősége a készülékekben és a beágyazott rendszerekben. Megkockáztatom a jóslatot, hogy ez az év úgy vonul be a Linux történetébe, mint „a linuxos készülékek éve” – olyan termékekkel, mint az egyre nagyobb számban a piacra kerülő Linux-alapú PDA-k, mobiltelefonok, webpadok és set-top szórakoztató rendszerek. A kétszáz kiállítóból mintegy húszan jelezték a kiállítási útmutatóban, hogy „beágyazott rendszerű” termékeket kínálnak, de közel kétszer ennyien támogattak valamilyen fajta beágyazott terméket vagy szolgáltatást. Ha az olvasó elmulasztotta megnézni a kiállítást, íme egy rövid összefoglaló arról, mit találtam.

### Accelent Systems

Az Accelent Systems bemutatta egységes fejlesztői felületét, egy beágyazott Linux- és StrongARM-alapú referencial felületet, mely a vállalatoknak nyújt segítséget Linux-alapú internet-berendezések és más beágyazott készülékek fejlesztésében. ➔ <http://www.accelent.com/>

### Applied Data Systems

Az Applied Data Systems négy bemutatót tartott grafikus felhasználói felületű, beágyazott Linux- és StrongARM-alapú egy lapra szerelt („egykártyás”) gépeiről (single-board computer – SBC), annak ellenére, hogy a szállítás során eltűnt minden kiállítási berendezésük. Ezek között volt a Bitsy, 3x4 hüvelykes (7,62x10,16 cm) méretű SBC, mely beágyazott Linuxot futtat. A Tandem, a kétcárú StrongARM SBC, mely egy processzorról két független képernyőt hajt meg beágyazott Linux segítségével. A Graphics Client Plus, mely a Century Software linuxos asztali eszközkészletét futtatja, valamint a Graphics Master, mely szintén beágyazott Linuxot futtat. ➔ <http://www.applieddata.net/>

### Axiom

Az Axiom kiállított seregnyi beágyazott és ipari célra készült SBC-t a PC/104, EBX, 3,5 hüvelykes (8,89 cm) és felezett méretű kártyaformátumokban. Az Axiom termékek mostantól rendelkeznek linuxos támogatással. ➔ <http://www.axiomtek.com/>

### BSDi

A BSDi standján lévő felirat szerint „az eBSD – az internetszakember választása az internet-alkalmazásokba és berendezésekbe való beágyazásra”. Nem volt hozzá semmilyen további magyarázat, sem bárki a környéken, aki mondhatott volna valamit a cég beágyazott piaci stratégiájáról. ➔ <http://www.bsdi.com/>

### Century Software

A Century Software dobogóján a RedHat pavilonban a Century Software-féle Microwindows, a ViewML és a PDA programokat három bemutató keretében ismertették:

- A Microwindows alatt futó teljes értékű MPEG videólejátszó első nyilvános bemutatása, mely a Simple DirectMedia Layer (SDL) nevű gyors kerettárazó (framebuffer) megjelenítő módszeren alapul, ADS Graphics Master SBC-n futott.
- A Microwindowst, a legújabb ScreenTop PDA készletet, a 802.11 vezeték nélküli hálózati rendszert és a Doom játékot mind egy Compaq iPAQ gépen mutatták be. Greg Haerr, a Century Software képviselője a legfrissebb ScreenTop programokat „felhasználóbarátabb és Palmtop-szerűbb terméként” mutatta be.
- A harmadik bemutatóban egy SIS alaplapot láthatunk, mely mindössze hét másodperc alatt betöltött mindent a Microwindows felületének megjelenéséig – figyelemre méltó teljesítmény. A titok: kombinálja a LinuxBIOS, a Memory Technology meghajtó (mely egy DiskOnChip Flashlemez használ), egy kicsiny Linux mag és természetesen a Microwindows használatát. ➔ <http://embedded.centurysoftware.com/>

### Compaq

A Compaq magától értődően bemutatta a kiemelkedően népszerű StrongARM-alapú iPAQ PDA-t, mely napjaink legkedveltebb Linux PDA programfejlesztő felülete lett. Próbáljuk ki és vásároljunk egyet! ➔ <http://www.compaq.com/>

### Coventive

A Coventive zsúfolt, kör alakú pavilonban mutatkozott be a linuxos világnak, számos termék és partnercég kapott helyet. Az érdekesebb bemutatók az alábbiak voltak:

- Az Axis cég (Japán) canD nevű, könnyen telepíthető, távirányítós set-top gépe, mely grafikus felülettel rendelkezik (az egész Coventive XTiNux-alapú rendszer elfér egy 8 MB-os ROM-ban).
- A Legend cég által gyártott új kínai PDA. Ez a gép szintén XTiNuxot futtat StrongARM-on, és szolgáltatásai között találunk három



érdekességet is: teljes értékű böngészőt PIM/SYNC szolgáltatásokkal; kínai kézírásfelismerőt (egyszerűsített és hagyományos változathoz); és egy Linux-alapú, folyamatos (online) raktár- és kereskedelmi alkalmazást.



- Egy Linux-alapú Smart kártyaolvasót a Disonic (Tajvan) cégtől.
- Egy Gigabyte Server Appliance rendszert, mely segítségével „a nem annyira szakemberek is képesek hálózati környezeteket kiépíteni és felügyelni”.  
➔ <http://www.coventive.com/>

### DevelopOnline

A DevelopOnline öt bemutatóval szemléltette saját online képességeit és fontos partnereit:

- Bemutatták az STMicroelectronics STPC internet-közelítő lapkagépet (system-on-chip – egyetlen lapkára összeépített teljesértékű számítógép) mintatervét.
- A DevelopOnline egységes fejlesztőkörnyezet (IDE) bemutatópéldányát. Ez pontosan szemlélteti, hogyan használhatják a fejlesztők a szolgáltatást program betöltésére és kipróbálásra kiválasztott referenciakörnyezeteken, a számos online beágyazott Linux programfejlesztő készlet (SDK) valamelyikével.
- A RidgeRun, a DevelopOnline partnercége elmagyarázta és szemléltette a DSPLinuxot, mely beágyazott Linux OS és kifejezetten a TI kétmagú processzorok követelményeihez igazodik. Ezek a processzorok egyetlen lapkán tartalmaznak egy ARM és egy DSP processzort.
- Megismerhettük az Insignia Solutions-féle Jeode beágyazott virtuális gépet (EVM), ami RedHat Linux alatt futó ThinkPadon bemutatott Java-megfelelő beágyazott programkörnyezet.
- Valamint megtekinthettünk egy távoli bemutatót, melyet a DevelopOnline-féle módszer segítségével lehetett elérni egy iPAQ gépen, Hard Hat Linux alatt futott egy Jeode EVM és egy PointBase beágyazott adatbázis. ➔ <http://www.developonline.com/>

### HP

A Hewlett-Packard bemutatta új Chai programcsomagját, melyet internetes berendezések létrehozásához használhatunk.

➔ <http://www.hp.com/products1/embedded/index.html>

### IBM

IBM-ék bemutatták népszerű Linux karórájukat, amely teljes Linux-rendszer egy apró (56 mm x 48 mm) karórában. Az IBM kivitte a kiállításra a DB2 Everywhere-t is, a beágyazott rendszerekhez készült adatbázisrendszert, mely kevesebb, mint 150 KB-ba belefér. A termék hivatalos bejelentése is a kiállításon történt meg. Óriási pavilonja másik részén az IBM egy Linux-alapú kiskereskedői eladói rendszerről adott módszertani bemutatót.

➔ <http://www.ibm.com/linux/>

### Intel

Intelék számos beágyazott linuxos bemutatót tartottak. Megismerhettük az Assabet mintadarabját a StrongARM SA-1110-hez (ami a Compaq-féle iPAQ szerkezet alapja). Az Assabetet egy telepről táplálták és – az Intel szóvivője szerint – 10-12 óra hosszat működik kikapcsolt LCD háttérfénnyel, illetve hat órát, bekapcsolt háttérfénnyel. Az Intel-féle új XScale mikroszerkezetet, a StrongARM

utódát is bemutatták, mely beágyazott Linuxot futtatott. A MontaVista és a LynuxWorks standján láthattuk.

➔ <http://developer.intel.com/>

### Lightning Instrumentation

A Lightning Instrumentationt egy apró, beágyazott Linuxot futtató útválasztót, a MultiCom Ethernet II-t állította ki. A Lightning-Linux, a MultiCom útválasztóban használt beágyazott Linux, a készüléktől függetlenül is megvásárolható lesz. A Lightning-Linux sok összetevőjéhez hozzájuthatunk a GPL alatt. ➔ <http://www.lightning.ch/>



### linAXE Project

A linAXE Project egy apró standon valami vadonatúj mutatott be. A linAXE úgy indult, mint egy Linux-alapú valós idejű OS (RTOS) kifejlesztésére irányuló erőfeszítés, a népszerű „BattleBots” harci robotok irányításához. Azóta a linAXE alkalmazási területe szélesebb lett.

➔ <http://linaxe.sourceforge.net/>

### Lineo

A Lineo cég pavilonja szokás szerint pezsgett, számos bemutatót láthattak az érdeklődők:

- A Lineo-féle uCsimmm utóda az uCdimmm. Az új soDIMM formátumú (1,7x2,7 hüvelyk, 4,3x6,8 cm) SBC egy DragonBallVZ mikrokontrolleren alapul és, nem meglepően, uClinux alatt fut.
- Jelen volt a Trolltech cég (a Lineo partnere) és bemutatták a Qt/Embeddedet, illetve a Qt Palmtop Environmentet (QPE).
- A Rappore cég (szintén a Lineo partnere) beágyazott Linux-alapú Bluetooth rendszert állított ki, mely egy babaházból állt, világítással, garázzsal, ajtóval stb. Egy hordozható gép vezérelte, és minden összekötést Bluetooth vezeték nélküli adatcserével valósítottak meg.
- Bemutatták a Lineo SecureEdge VPN útválasztó felületet is.
- Egy Embedded Planet gyártmányú kis PowerPC-alapú PC/104 formátumú SBC-n mutatták be az M-Systems cég legújabb DiskOnChip beágyazott linuxos meghajtót.
- Kiállították az új Lineo Academic Student Kitet. E készlet tartalmazza az uCsimmet, az uClinux OS-t, egy kis kísérletező kártyát és részletes használati utasítást. Diákoknak szánták és 250 dollárért megvásárolható. Magában foglalja a tanfolyam- és a program CD-eket. Tizenhárom egyetemen már használják a készletet, a számítógép-tudományi tananyag részeként.  
➔ <http://www.lineo.com/>

### LynuxWorks

A LynuxWorks pavilonjában bemutatták a nyílt forrású BlueCat Linuxot és a cég által jegyzett LynxOS forrás idejű OS új változatát, mely a LynuxWorks és az ISDCorp együttes erejének bizonyítéka. Utóbbi céget tavaly nyáron vásárolta fel a LynuxWorks. A bemutatók közt szerepelt egy érdekes is, mely a LynxOS leendő változa-



tának előzetes kiadásában megtalálható bináris alkalmazási felület (Application Binary Interface – ABI) támogatását mutatta be. Hasonló kiépítésű gépek (Linux és LynxOS rendszer alatt) futtatták a Quake ugyazason bináris változatát, és egy hasonló erőforráséhes programot. A futás alatt folyamatosan változtatták a LynxOS alatt a fontossági mutatókat, hogy érzékeltessék egy „igazi RTOS” képességeit. Bemutatták az új SpyKer valós idejű eseményábrázoló és nyomkövető eszközt, valamint a PhatNoise-féle Phatboxot, autóba való MP3-lejátszót, mely beágyazott BlueCat Linuxra épült. Intelék standján egy BlueCat Linux alatt futó magas rendelkezésre állású, hibatűrő sakkprogramot mutattak be, az M-Systems pedig a BlueCat Linuxhoz mutatott be DiskOnChip támogatást.

➔ <http://www.linuxworks.com/>

### Metro Link

A Metro Link bemutatta a Micro-X-et, a mely beágyazott Linuxhoz készült ablakkezelő megoldás. „A Micro-X az X Window rendszerprotokollon alapul, nem kell tehát egy új felületet megtanulni beágyazott rendszerek fejlesztéséhez” – mondták. A Micro-X egy StrongARM SA-1110 felületen futott. A Micro-X belefér 575 K-s tábla és az x86-ot, a PowerPC-t és az ARM/StrongARM-ot, nemsokára a MIPS-et is támogatja. ➔ <http://www.metrolink.com/>

### Metroworks

A Metroworks a közelmúltban továbbfejlesztett CodeWarrior fejlesztőkörnyezetét mutatta be, melynek beágyazott hibakeresője és távoli hibakeresési képességei vannak. A bemutató egy Motorola PowerPC 8260 mintakártyán és egy Embedded Planet nevű, PC/104 formátumú PowerPC 823 SBC-n futott. A stand témája a „Beágyazott Linux – növeljük az értelmes készülékek IQ-ját”.

➔ <http://www.metroworks.com/>

### MontaVista

A MontaVistánál rengeteg látnivaló volt, nyolc termékbe-mutatóval és öt partner bemutatójával. Néhány a kiemelkedők közül:

- Egy nagy megbízhatóságú CompactPCI rendszer, Hard Hat Linux, Ziatech és Motorola CompactPCI processzoron futtatva. A rendszeren a MontaVista cég magas rendelkezésre állású programjával, hálózati programmal és menet közben cserélhető meghajtókkal.
- Különböző többfelületes (cross-platform) fejlesztőeszközök, grafikus beágyazott magbeállító eszközök, grafikus fejlesztőkörnyezetek, távoli hibakeresők és teljesítményelemző eszközök.
- @win, egy kis helyigényű (200 KB) felhasználói felület/ablakkezelő rendszer az Adelinuxtól (a MontaVista egyik koreai partnere), mely egy Compaq iPAQ gépen fut (együttműködik a GTK/GTK+ és a QT eszközkészletekkel).
- Egy Compaq iPAQ PDA, mely a Hard Hat Linux és

az X Window rendszer alatt fut, Netscape 6-os internet-böngészővel és egy élő videoadást megjelenítő programmal.

- Valós idejű rendszermag MontaVista-féle megközelítésben. Két azonos kiépítésű gépen hang-alkalmazásokat futtattak. Az egyiket általános Linux mag futott, míg a másikon a MontaVista-féle valós idejű továbbfejlesztések is dolgoztak. Mindkettő processzor- és ütemezési válaszidejét egy folyamatosan frissített diagramon szemléltették.
- Motoroláék 74xx Altivec processzora Hard Hat Linuxot futtatott egy Motorola Sandpoint beágyazott referencia-szerkezeten, a Hard Hat Linux alatti beágyazott Java alkalmazások támogatása az IBM-féle VisualAge Micro Editionon (VAME) és a Hard Hat Linux alatt futó IBM-féle 750CX/CXe PowerPC-n alapult.
- Egy StrongARM-alapú ePort nevű kiskereskedelmi eladói (point-of-sale) alrendszer (RadiSYS for USA Technologies tervezés), mint elektromechanikai alrendszer állítottak ki, egy üdítőital-automatába beépítve, így most kólavásárlás közben is böngészhetünk a Weben.
- Egy 1U keretbe szerelhető ájtjáró berendezést (Diversified Technologies gyártmány) is láthattunk, Hard Hat rendszerrel felszerelve.
- Az egyik legelső működő beágyazott Linux bemutató (természetesen a Hard Hat Linuxsal), mely az Intel új XScale mikrostrukturán futott (a StrongARM processzor legújabb nemzedéke), amely az Intel-féle XScale felület próbaváltozatán futott. ➔ <http://www.mvista.com/>

### Neoware

Bemutatták a Neoware-féle díjnyertes Eon számítástechnikai berendezést (az „Anything box”), mely a Neolinux 2.0 beágyazott Linux-rendszer alatt fut. Egyedülálló Internet-berendezés központú tulajdonságokat mutattak be, mint az ezConnect (egyszerű felhasználói felület) és az ezSnap (terjesztett programok támogatása).

➔ <http://www.neoware.com/>

### NexCom

A NexCom kiállított számos beágyazott és ipari rendeltetésű SBC-t különféle elterjedt kisebb és nagyobb formátumokban. A NexCom termékek képesek Linuxot futtatni.

➔ <http://www.nexcom.com/>

Az OnCore Systems bemutatta a valós idejű terméket az OnCore Linux for Real-Time-ot, egy olyan felületet, amely erősen valós idejű rendszerkörnyezeten belül több példány Linuxot képes futtatni.

➔ <http://www.oncoresystems.com/>

### PEP Modular Computers

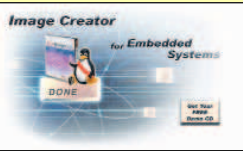
A PEP Modular Computers kiállította Linux támogatású 3U és 6U CompactPCI kártyáit és rendszereit. Ezeket a termékeket nagy megbízhatóságot igénylő területekre, távközlési és internet-elemeknél, ipari vezérléshez, katonai és orvosi alkalmazásokhoz, ajánlják.

➔ <http://www.pep.com/>



## Portwell

A Portwell több beágyazott és ipari rendeltetésű SBC-t állított ki, EBX és más beágyazott és ipari formátumban. A Portwell termékek is rendelkeznek Linux támogatással.  
 ☞ <http://www.portwell.com/>



## RedHat

A RedHat három példát mutatott be beágyazott rendszerekre: a korábban tárgyalt Century Software-t, a RedHat Embedded Linux

Development Kitet, mely egy Motorola MBX860 EBX belsővel rendelkező SBC-n futott és az uClinux és eCos rendszereket, melyek egy NetSilicon NET+Lx referencia tervezőrendszeren futottak.

☞ <http://www.redhat.com/embedded/>

## RedSonic

A RedSonic bemutatta a RedIce-Linux valós idejű Linux operációs rendszert és hozzávaló fejlesztő és hibakereső eszközöket. Ezen kívül a RedSonic bemutatta új set-top megvalósítását. ☞ <http://www.redsonic.com/>

A Sun Microsystems a beágyazott Linux-alapú rendszereket támogató J2ME Java számos beágyazott eszközzel jelent meg. A gépi felületek közt szerepelt a Compaq iPAQ PDA és egy Arcom beágyazott Linux referenci felület. A J2ME kettő és négy megabájt közti tárolóhelyet igényel. A Wireless 802.11 módszerét is bemutatták az iPAQ beágyazott Linux-rendszeren.

☞ <http://www.sun.com/software/>

A TimeSys öt valós idejű Linux bemutatót tartott, melyek a TimeSys-féle Linux/RT operációs rendszer QoS (quality-of-service, szolgáltatás minőségének biztosítása) és valós idejű képességeit szemléltetik:

- Egy Embedded Planet PowerPC 860 lapot, melyen egy TimeSys Linux/RT alatt a TimeTrace futott, a rendszeresemények és tartalomváltások folyamatos figyelésére szolgált.
- A JTime, a TimeSys valós idejű Java virtuális gépe, mely a TimeSys Linux/RT fölött futott, egy vezeték nélküli soros összeköttetésen keresztül irányított egy kis robotot.
- Egy StrongArm-alapú Corel NetWinder rendszer két sakkozó program számára elegendő számítási idő biztosításával szemléltette a Linux/RT QoS képességeit.
- Bemutatta a Linux/RT QoS képességeit egy asztali rendszeren. Egy konferenciaprogram számára hálózati és processzorteljesítmény biztosításával, így a rendszer elfogadható élő kapcsolatot biztosított az erős rendszerterhelés ellenére.
- A Quake III segítségével bemutattak egy asztali gépet, melyen a processzoridő biztosítását egy egyedülálló „QoS gomb” vezérli. A gomb segítségével játék közben is változtathatjuk a játékhoz rendelt erőforráskorlátokat, ezzel érezhetően ronthatjuk vagy javíthatjuk a futás sebességét.

☞ <http://www.timesys.com/>

A Transvirtual Technologies bemutatta a PocketLinux legújabb változatát, egy PDA-khoz, kézi és beágyazott gépekhez készült Linux/Java megvalósítást. Az érdekes új tulajdonságok közt szerepeltek a 802.11 vezeték nélküli összeköttetések, továbbá MPE- és MPEG-lejátszók. Számos PIM-alkalmazás kapható már a PocketLinuxra, általános használatú programokat (naptár, címjegyzék, jegyzetkönyv, levelező, Flash-lejátszó, gyors üzenetküldés, XML/XHTML böngésző stb.), de játékokat is tonnaszámba találunk.

☞ <http://www.transvirtual.com/>

## Trolltech

A Trolltech cég a Qt/Embedded és a Qt Palmtop Environment (QPE) mutatta be, melyek egy Compaq iPAQ PDA-n futottak. Ingyen hajlékonylemezeket is ajándékoztak, melyeken egy PC-n futó beágyazott Linux-rendszer található, Qt/Embedded és PIM-alkalmazások (határidőnapló, címjegyzék, szövegszerkesztő, fájlböngésző stb.), valamint játékok (aknakereső, Tetrix és pasziánsz). A hajlékonylemezek tartalma letölthető a Trolltech webhelyéről is.

☞ <http://www.trolltech.com/>

## Viosoft

A Viosoft bemutatta beágyazott linuxos többfelületes SDK-ját RISC felületekre, köztük szerepel a MIPS, az ARM/StrongARM és a PowerPC. Bemutatták az Arriba! nevű Java-alapú fejlesztőfelületet is, melynek Windows-stílusú grafikus hibakeresője van.

☞ <http://www.viosoft.com/>

## ZF Linux

A ZF Linux Devices MachZ nevű egylapkás rendszeréhez mutatott be támogatást, beleértve egy apró MachZ bemutatókártyát, összetett fejlesztőrendszert (MachZ-alapú internetes referenciaberendezésképzést) és a Tri-M-féle MachZ-alapú PC/104 formátumú SBC-t, amely az Embedded Linux Journal tervezői versenyén tűnt ki

☞ <http://www.zflinux.com/>

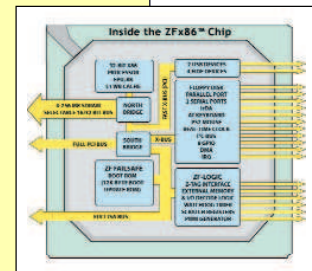
Mint láthatjuk, rengeteg beágyazott termék és bemutató volt a LinuxWorldön – sokkal több, mint a korábbi kiállításokon. Amilyen ütemben a Linux behatol a beágyazott piacra, a következő, augusztus 27–30-a között San Franciscóban rendezendő US LinuxWorld Expo and Conference-en még több beágyazott Linux-rendszer várható.



*Rick Lehrbaum*

(rick@linuxdevices.com) hozta létre a LinuxDevices.com „beágyazott Linuxok portálját”, ez nemrég tagja lett a ZDNet Linux Resource Center-nek. Rick 1979 óta beágyazott

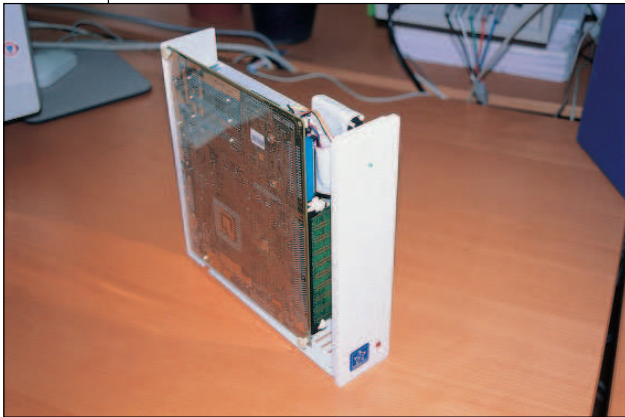
rendszerek fejlesztésével foglalkozik. Társalapítója az Ampro Computersnek, alapító tagja a PC/104 Consortiumnak, és fontos szerepet játszott abban, hogy elindulhatott az Embedded Linux Consortium.





## Portocom MiniPC próbafutam

A Portocom Rt. új termékéről, a MiniPC-ről előző számunkban már beszámoltunk, most személyes tapasztalatainkat szeretnénk megosztani olvasóinkkal, mivel szerkesztőségünkben időzött egyik példánya. Készítői hálózati munkaállomásként ajánlják, akár hálózatról induló, lemez nélküli



elsőként az SSH szolgáltatást, majd a Sambát állítottuk be. Így máris a megszokott irodai környezetben érezhettük magunkat, és a munkánkat ott folytathattuk, ahol a másik számítógépen abbahagytuk.

Legnagyobb előnye a mérete és csomagolása, akár egy nagyobb női táskába is belefér, az íróasztalon pedig kényelmesen elfér. A processzor ventilátorán kívül nincs mozgó alkatrésze, így a meghibásodásra való hajlama is csökken. A hangja kellemes, a hagyományos asztali gépek ventilátorzajához képest szinte nem is hallható a processzorhűtő surrogása. Az áramellátást a hordozható gépeknél használatos táppal oldották meg, mely a házon kívül kapott helyet.

Mindent összevetve a MiniPC kedvező tudású, csendes és hálózati környezetben mindenki számára megfelelően használható készülék, legfőképpen ha a lemez nélküli munkaállomásokat vesszük figyelembe.

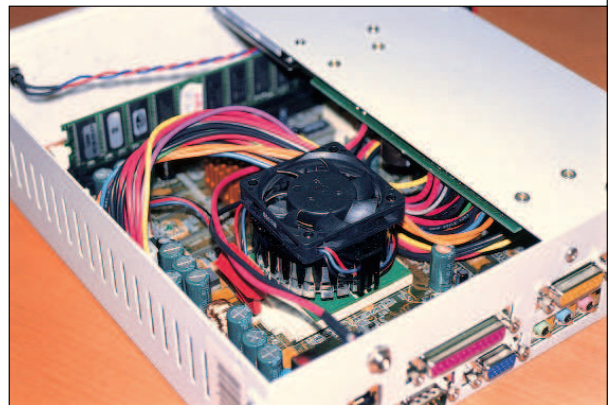
➔ <http://www.portocom.hu>

Hasonló számítógépek:

➔ [http://www.compaq.hu/termeink/asztali\\_gepek/ipaq/](http://www.compaq.hu/termeink/asztali_gepek/ipaq/)

(Diskless), akár a saját merevlemezére telepített operációs rendszerrel. Mindkét eljárásnak megvannak az előnyei és a hátrányai. A jelenlegi Windows operációs rendszereket nem lehet hálózatról betölteni, ezért azoknál csak a Windows 95 vagy 3.1 jöhet számításba. Szerencsére az általunk használt operációs rendszerek egyikének sem jelent gondot ez a művelet. Könnyű hálózati telepítés után máris használatba vehettük a Debian 2.2 Potato operációs rendszerrel felvértezett gépet. A rendszer összes eszközét gond nélkül tudtuk használni. Mivel a készüléket hálózati ügyfélként ajánlják, sem lemezes meghajtó, sem CD-ROM nem csatlakozik hozzá. Ezeket csak a ház megbontásával lehet a számítógéphez csatlakoztatni. Az egész MiniPC alapja egy micro-ATX szabványú alaplappal, melyen megtalálható egy VGA kártya, egy hálózati kártya, egy hangkártya és egy modem is. (A modem használatához egy külön modult kell az alaplaphoz illeszteni, ami a MiniPC-be nincs beszerelve.)

Alapkiépítésben a gép nem tartalmaz merevlemez, 600 MHz-es Celeron processzorral és 64 MB RAM-mal szállítják. Természetesen kérhetünk gyorsabb processzort, nagyobb tárképességű merevlemez, valamint több memóriát. A nálunk járt számítógép processzora 800 MHz-es Pentium III-as volt, 128 megabájt memóriával, egy 20 GB-os merevlemez kíséretében, mellékeltek hozzá egy 32 MB-os Flashlemez is, melyre mini Linuxot telepíthetünk. A VGA kártya népszerű és elterjedt lapkakészletű nVidia TNT2-es, memóriamérete 8–32 MB között állítható, a rendszer memóriájának rovására. A hálózati kártyát pedig a SiS 900/7016 meghajtóval sikerült az adatok fogadására és küldésére bírni. Mivel ezt a szerkezetet hálózati környezetre teremtették,



Csontos Gyula

(Csontos.Gyula@linuxvilag.hu)  
a Linuxvilág hír- és CD-szerkesztője,  
valamint a [www.linuxvilag.hu](http://www.linuxvilag.hu)  
tartalomfelelőse.

## Új termékek

**RackMount-1 UAXe a Rave Systemstől**

A RackMount-1 UAXe egy vékony, 1U formájú kiszolgáló, amit a Sun UltraAXe alaplapjával szereltek. Az alapfelszereléshez tartozik egy 33 MHz-es 32 bites PCI kártyahely, két előlről elérhető lemezmeghajtó, egy ethernet 10/100 BaseT kapu, legfeljebb 1 GB memória, alaplapra szerelt PCI grafikus kártya, egy párhuzamos, egy PS/2 és négy soros kapu, valamint a 150 wattos tápegység. Vásárláskor választhatunk Solaris és Linux között. Lehetőség nyílik más gyártótól származó SCSI vezérlő beépítésére is, ha gyorsabb adatátvitelre és több lemezhelyre van szükség. Az 1UAXe-t 300 MHz-es UltraSPARC-III processzorral, 48x EIDE CD-ROM-mal és előre telepített RedHat operációs rendszerrel szállítják.

Adatok: Rave Computer Association, Inc., 36960 Metro Court, Sterling Heights, Michigan 48312, telefon: 1-800-966-7283  
<http://www.rave.net/>.

**Kétkapus RocketPort soros elosztó**

A Comtrol Corporation bemutatta a RocketPort soros elosztó ipari felhasználásra tervezett kétkapus változatát. A DIN sorba köthető kétkapus modell csökkenti a vezetékigényeket, mert a meglévő ethernethálózatot használja, kiküszöböli a felesleges PC-eket, és egy tartalékkaput biztosít. A kétkapus kiépítés támogatja a gyorsabb ethernetkapcsolatot, a több soros csatlófelületet, a valós idejű operációs rendszereket és az egyszerű kapcsolatokat a COM kapukon keresztül. A Comtron támogatja a 10/100 Base-T ethernetet és az RS-232/422/485, valamint a ModBus eszközökkel való kapcsolatot.

Adatok: Comtrol Corporation, 6655 Wedgwood Road, Suite 120, Minneapolis, Minnesota 55311-3646, telefon: 1-800-926-6876  
 e-mail: [info@comtrol.com](mailto:info@comtrol.com)  
<http://www.comtrol.com/>.

**USB négykapus soros átalakító a Keyspantól**

Ezzel a Keyspan egységgel négy soros eszköz

köthető egyetlen USB kapuhoz. Minden egyes BP9 kapu 960 Kb/mp sebességű adatforgalmat engedélyez az RS-232 soros eszközök felé. Az USB csatló támogatja a Linux 2.4-et. Az eszközt vállalati és ipari környezethez szánták, jól használható laboratóriumokban, POS egységek gyártásánál és értékesítésénél, a folyamatvezérlésben és az adatgyűjtő alkalmazásokban.

Adatok: Keyspan, 3095 Richmond Parkway #207, Richmond, California 94806, telefon: 510-222-0131  
 e-mail: [info@keyspan.com](mailto:info@keyspan.com)  
<http://www.keyspan.com/>.

**BlueCat Linux az Intel Xscale-re**

A LynxWorks bejelentette a BlueCat Linux megjelenését az Intel új processzor-magcsaládjára, az Xscale-re. Az Xscale tervezésénél a legtöbb figyelmet a nagy teljesítményre, a kis fogyasztásra és a mérsékelt hőleadásra fordították, így különféle piaci területeken lehet sikeres, például a vezeték nélküli kézi számítógépek és az internetes eszközök terén. A BlueCat próbaváltozata része lesz az Intel IQ80310 Xscale kipróbálásra szánt fejlesztőkörnyezetének. A BlueCat és az Xscale együttese tároló- és hálózatos eszközökben, valamint az Internetre kapcsolható kézi számítógépekben kaphat szerepet.

Adatok: LynxWorks, Inc., 2239 Samaritan Drive, San Jose, California 95124, telefon: 408-879-3900  
<http://www.linuxworks.com/>.

**48-kapus hálózati csatlókártya az Arieltől**

Már kapható az Ariel RS4200 nagy-sűrűségű 56 K/ISDN PCI hálózati ikerkártyájának 48-kapus változata. Az RS4200 eszközben egyszerre

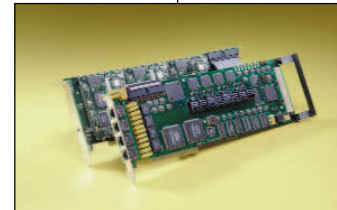
vannak jelen a T1/PRI vagy E1/PRI csatlók, valamint az 56 K-s és alapsebességű ISDN-kapuk. A kártyák segítségével könnyedén építhetünk behívó és kitércsázó rendszereket. A 48-kapus kártyarendszer 24-kapus lépésekben bővíthető, legfeljebb 120 kapuig. A Java-alapú karbantartóprogrammal távolból felügyelhető, így a rendszergazda tetszőleges számú RS4200-as kaput követhet figyelemmel, akár a helyi hálózatban, akár az Interneten. A program további lehetőségei között van a T1/E1/PRI vonalak beállítása, illetve különböző erőforrások üzemén kívül helyezése.

Adatok: Ariel Corporation, 2540 Route 130, Cranbury, New Jersey 08512, telefon: 609-860-2900  
 e-mail: [info@ariel.com](mailto:info@ariel.com)  
<http://www.ariel.net/>.

**TinyTERM v4.13**

A TinyTERM program biztonságos ügyféloldali hozzáférést tesz lehetővé unixos és IBM gépekhez a vállalati belső hálózatán keresztül. Az asztali PC-k felhasználói számára a TinyTERM hálózati segédprogramokat biztosít, például fájl- és nyomtatómegosztást, hálózati nyomtatást, fájlmásolást. A 4.13 változat több beleszólási lehetőséget enged a felhasználóknak a nyomtatási feladatok végrehajtásába. Az új tulajdonságok között található a lapdobás közvetlen eszközre történő nyomtatásnál, a nagyméretű nyomtatási feladatok továbbfejlesztett kezelése, a nyomtatók betűkészlet-kiválasztásának támogatása, valamint továbbfejlesztett Telnet hibakezelést és frissített súgófájlokat tartalmaz. A TinyTERM v4.13 letölthető a <http://tt413.centurysoftware.com/> címről és a v4.12 felhasználói számára ingyenes.

Adatok: Century Software, 5284 South Commerce Drive, Suite C-134, Salt Lake City, Utah 84107, telefon: 1-801-268-3088  
 e-mail: [sales@centurysoftware.com](mailto:sales@centurysoftware.com)  
<http://www.centurysoftware.com/>.



## A hónap szakmai tanácsai



### Önműködő billentyűleütések

Azt a megbízást kaptam, hogy tegyem önműködővé cégünk különböző programokkal végzett feladatait. A baj csak az, hogy rengetegszer kellene a billentyűzetről történő bevittelt utánnoznom. Hogyan lehet parancsfájlból billentyűparancsokat kiadni egy programnak?

Még a billentyűkódok teljes felsorolását sem sikerült megtalálnom.

*Paul, tanx1@stlnet.com*

Próbáld ki az `expect` parancsot (A Linuxvilág a 2000. decemberi számában a 68. oldalon, olvashattál erről – a szerk.!) A legtöbb változat

tartalmazza ezt az eszközt, ennek segítségével az emberi beavatkozást igénylő programok önműködővé tehetők.

*Mario Neto, mneto@argo.com.br*

Én a következő egyszerű módszert használom, ha parancsfájlból indítok programot: `echo "y" | xyz`. Ebben az esetben `xyz` a program neve, és az "y" választ várja a kérdésére.

*Usman S. Ansari, uansari@yahoo.com*

### Az FTP felhasználók feljebb tudnak lépni

Az FTP felhasználóim tudnak felfelé lépkedni a saját könyvtárukból, és egyenesen a / könyvtárba jutni. Ők a `leach` FTP-t használják, és ez a program lehetővé teszi, hogy a saját könyvtáruknál feljebb lépjenek, ez pedig számomra elfogadhatatlan.

A másik gondom, hogy húszt virtuális gépem van (csak a nevük különbözik, az IP-jük azonban ugyanaz) az Apache kiszolgálómon. Sokszor előfordult, hogy amikor hozzáadtam egy új virtuális gépet, nem működött rá a névfeloldás, amíg nem mozgattam át a virtuális gépek listájának tetejére a `httpd.conf`-ban. Kifogytam a virtuális kiszolgálókból? A számítógépem kiépítése a következő: AMD K6 200 MHz processzor 64 MB RAM-mal.

*Aaron, aaron@x56.net*

Meg lehet büttyölni a `wuftpd`-t, hogy a felhasználók ne tudjanak felfelé lépkedni, de jobban jársz, ha a `proftpd`-t telepíted (ha még nem azt használod). Ebben be lehet állítani minden felhasználó gyökérkönyvtárát a saját könyvtárára, vagy tetszőleges másik könyvtárra (lásd még a <http://www.proftpd.net/> címen).

*Marc Merlin, marc\_bts@valinux.com*

Nem fogytak el a virtuális kiszolgálók, de ez a beállítás kicsit trükkös, néhány dolgot ellenőrizned kell. A hibás gépet megelőzően nem szerepelt-e olyan `ServerName` vagy `ServerAlias` érték, amely ugyanazzal a névvel rendelkezik? Ha van egy `ServerAlias *.example.com` érték, akkor a `warez.example.com` nem fog működni. Minden \*-gal kezdődő álnevet a felsorolás végére kell tenni.

Használd az `nslookup` parancsot, hogy meggyőződj arról, hogy minden virtuális gépre működik a DNS. Ellenőrizd, hogy a `NameVirtualHost` bejegyzés a neki megfelelő `VirtualHost` fejezet előtt van-e. Bőgészd át a

<http://httpd.apache.org/docs-2.0/vhosts/details.html> oldalt, valamint a webmesterek leggyakoribb és legkevésbé szeretett olvasmányát, az `error_log`-ot.

*Don Marti, dmarti@linuxjournal.com*

### A shadow-fájl frissítése

Hogyan lehet frissíteni a `shadow` fájlt, miután a `passwd` fájl kézi szerkesztésével hozzáadtam egy új felhasználót? Hogyan lehetne eltávolítani a `suid` bitet tetszőleges alkalmazásról?

*Kedar, kedar.potdar@imandiasia.com*

A felhasználó létrehozása után a `pwconv` paranccsal tudod frissíteni a `shadow`-fájlt.

*Usman S. Ansari, uansari@yahoo.com*

A `suid` bitet a `chmod -s` fájlnev paranccsal távolíthatod el.

*Marc Merlin, marc\_bts@valinux.com*

Előfordulhat, hogy valaki szerkesztés közben nyúl hozzá a `/etc/passwd` fájlhoz. Ez elkerülhető, ha a `vipw` programmal szerkeszted. A `vipw` az általad választott szövegszerkesztőt használja, és nem feltétlenül a `vi`-t.

*Don Marti, dmarti@linuxjournal.com*

### A levélszemelők címekre vadásznak

Az utóbbi időben egyre gyakrabban tapasztalom, hogy a kéretlen levelek küldői úgy kísérelnek meg üzeneteket küldeni a gépeimre, hogy gyakori neveket próbálgatnak végig (pl.: `david@example.com`, `bill@example.com`, `mike@example.com` stb.). Van valamilyen önműködő módszer arra, hogy kizárjam X sikertelen kísérlet után a küldő gépet, vagy ami még jobb, rögtön hozzáadjam az IP-címét a `sendmail` tiltólistájához?

*Waldo Jaquith, waldo@waldo.net*

Írhatnál mini Perl programot, amely a sikertelen próbálkozások naplófájlját dolgozza fel, hozzáadja az IP-t a tiltólistához, és újraindítja a `sendmail`-t.

*Marc Merlin, marc\_bts@valinux.com*

### A PHP nem tud a MySQL-hez kapcsolódni

Telepítettem a MySQL3 és a PHP4 RPM csomagokat. Mind az Apache, mind a MySQL rendben fut a héjból. A PHP4 szintén fut az `infophp()` függvényhívással.

A baj csak az, hogy nem tudok a MySQL-kiszolgálóhoz kapcsolódni. Mi az oka?

*Haidar AM, haidar@netexecutive.com*

Bár nem említetted, feltételezem, hogy a `phpinfo` program kiírta, hogy a MySQL támogatott. Ebben az esetben ellenőrizd a MySQL táblázatot, hogy a felhasználó/jelszó/gépnév adatokat – ez esetben a gépnév a `localhost` – helyesen adtad-e meg.

*Mario Neto, mneto@argo.com.br*

### Nem tudok rendszergazdaként belépni

Valamiért nem tudok rendszergazdaként belépni a helyi linuxos gépemre (nem távolból próbálkozom). Általános felhasználóként tudok csak belépni, majd a `su` paranccsal rendszergazdává változom. A másik gondom, hogy amikor használnom kell az `adduser` parancsot vagy valami



hasonlót, akkor meg kell adnom a parancs teljes elérési útját (/usr/sbin/adduser).

*Devraj Sen, devraj@trihedral.com*

A /etc/securetty fájl tartalmazza azokat a terminálokat, ahonnan a rendszergazda beléphet. Elképzelhető, hogy a konzolos belépéshez tartozó bejegyzéseket eltávolították. Egyszerűen rakd vissza, azaz ha rendszergazdaként szeretnél belépni az egyes konzolon, a fájlban szerepelnie kell a következő sornak:

```
tty1
```

Azért nincs beállítva a PATH környezeti változód, mert a su-t kapcsolók nélkül használva nem bejelentkezési héj indul el, ezért a rendszer a bejelentkezési fájlokat nem olvassa be. Ha azonban a su - parancsot adod ki, ezek is életbe lépnek.

*Andy Bradford, andyb@calderasystems.com*

Amennyire én tudom, a legtöbb – ha nem az összes – RedHat kiadás hibás a tekintetben, hogy különböző okokból a /sbin:/usr/sbin:/usr/local/sbin könyvtárak nincsenek benne a rendszergazdai elérési útvonalban. Erre az a gyors megoldás, hogy hozzáadjuk ezeket a PATH-hoz a /etc/profile-ban.

Szokatlan, hogy nem tudsz belépni rendszergazdaként.

Ellenőrizd, hogy a rendszergazdának van-e érvényes bejelentkezési héja, és a /etc/securetty tartalmazza a ttyx sorokat. Többet is megtudhatsz, ha beírod a tail -f /var/log/messages parancsot és megfigyeled a kimenetét, amikor másik konzolról megpróbálsz rendszergazdaként belépni.

*Marc Merlin, marc\_bts@valinux.com*

### A Perl késleltetve ír, azután meghal

Írtam egy egyszerű Perl programot, amely egy fájlból címetek olvas be, elmegy a megadott címekre, ott keresést hajt végre, majd a talált adatokat kiírja egy másik fájlba. Ezenkívül bejegyzéseket is készít a naplófájlba, valamint a szabványos kimenetre is ír. Az LWP::Simple modul használata során a következő furcsa viselkedést tapasztaltam:

1. a naplófájl nem íródik ki azonnal, úgy tűnik az OS gyorsítázza az írási kérelmeket.
2. egy-két óra szabályos működés után a program meghal, de az újraindítást követően ismét jól működik.

*Dave Barter, dave@phased.co.uk*

Ha megadod a \$| = 1 kifejezést a program elején, akkor a Perl minden print és write után kiad egy gyorsítárítást is (lásd man perlvar), ennek hatására azonnal a lemeze írja az állományokat. A programod halálának az egyik oka az lehet, hogy felemészti az összes memóriát. Futtasd a top programot a tiéddelel egyidejűleg, és látni fogod, hogy ez történik-e. Ha ez a gond, írd újra a programot, hogy a feldolgozás után szabadítsa fel a beolvasott lapok által lefoglalt memóriát. Ha felügyelet nélkül futtatod a programot, akkor az LWP::RobotUA modulot kellene használnod, mert ez figyelembe veszi a webmesterek kívánságát a tekintetben,

hogy a webhelyük melyik része dolgozható fel robotokkal.

*Don Marti, dmarti@linuxjournal.com*

### Hibás főfájrendszer

Szerettem volna telepíteni a RedHat Linuxot a Dell táskagépemen az e célra létrehozott lemezzésre. A telepítés sikeresen befejeződött, de amikor elindítottam a Linuxot, a gép a következő üzenet kiírása után lefagyott:

```
checking root file system
/ was not cleanly unmounted, check
forced
Setting filetype for entry log in /dev
(174593) to 6
Unattached inode 82384
UNEXPECTED INCONSISTENCY; run fsck
MANUALLY
(ie without -a or -p options)
An error occurred during the filesystem
check
Dropping you to a shell; the system
will reboot
Give root password for maintenance
Brian Weigner, brian.weigner@colorado.edu
```

Úgy tűnik, a Linuxod főfájrendszere megsérült. Egyfelhasználós üzemmódban kell elindítanod a gépet, úgy hogy a linux s parancsot írod be a LILO parancssorába, és addig vársz, amíg megjelenik a # jel. Ekkor írd be:

```
e2fsck -r /dev/
a_főfájrendszered_eszközneve
```

Az eszköz ellenőrzése megkezdődik, és a program minden hibánál megkérdi, mit tegyen vele. Valószínűleg sok ilyen kérdést kapsz. Csak akkor tudsz helyesen válaszolni ezekre, ha valamelyest ismered a Linux fájlrendszerét. Ez azonban kockázatos vállalkozás komoly munkára használt rendszeren. Mivel friss telepítésről van szó, az egyszerűség kedvéért válaszolj igennel minden FIX, REPAIR és REMOVE kérdésre. Miután végeztél, próbáld újraindítani a rendszert. Ha nem működik, a legjobb, amit tehetsz, hogy óvatosan újratelepíted.

*Felipe E. Barousse Boue, fbarousse@piensa.com*

Ez a jelenség a jövőben nem fordul elő, ha mielőtt kikapcsolod a gépet, kiadod a shutdown -h now parancsot.

*Don Marti, dmarti@linuxjournal.com*

### Nincs vonalhangom, de tárcsáznom kell

Van egy PCTEL belső modemem, amelyhez letöltöttem a linmodems.org-ról és telepítettem a meghajtóprogramokat. A modem akkor is tárcsázik, amikor a „No Dialtone” üzenet jelenik meg. Csak egyszer sikerült kapcsolatot teremtenem, és akkor is hosszú ideig kellett próbálkoznom, ráadásul az átviteli sebesség is alulmúlta a várakozásaimat.

*Krishna, as\_krishna@hotmail.com*

Próbáld ki, hogy okoz-e változást, ha az indító AT parancssorhoz hozzáadod az X1 kapcsolót.

*Marc Merlin, marc\_bts@valinux.com*

**Mi tehát a linuxos cégek stratégiája? Tudjuk, hogy nyereséggel el tudják adni a gépeket és a szolgáltatásokat. De el tudnak-e adni ugyanekkora nyereséggel programokat?**



© Kiskapu Kft. Minden jog fenntartva

## Alkalmazkodás a Bazár életéhez

Beszélgjünk a programüzletégről! Vegyük először a Microsoftot. Hé, még nem 2010-et írunk! Ha 2001-ben a programokról akarunk beszélni, muszáj foglalkoznunk a Nagy Testvérrrel is. Ez soha sem volt könnyű, de februárban *Jim Allchin*, a Microsoft operációs rendszerfejlesztő részlegének vezetője, még inkább megnehezítette a dolgot. Egy *Bloomberg*gel folytatott beszélgetése során a Linuxot a „szellemi tulajdon megsemmisítőjének”, valamint a nyílt forráskódot az „amerikai nép fenyegetésének” nevezte, néhány egyéb őrült, üldözési rögeszmére valló jelző mellett. Ezzel szeretne volna elterelni a figyelmet azokról a lassú és kevésbé címlapra kívánczó változásokról, amelyek az utóbbi években zajlottak a cégénél. Azokról a jelekről, melyek sejtetik, hogy a cégénél is észrevették, hogy a piacaik be vannak hálózva és éppen ezért, egyre kevésbé tekinthetők egyszerű fogyasztónak, eldobható „célpontnak”. Most már ők is sokkal inkább hasonlítanak bazárok látogatóira.

A bazár igazi piac: rendezetlen és lármás nyilvános hely, ahol a gazdaság ellentétes oldalai – kereslet és kínálat, termelés és fogyasztás – karnyújtásnyira találhatók egymástól.

Az igazi piacon a vevő véleménye sokkal nagyobb hatással van az eladóra, mint az eladó marketingterve a fogyasztóra. Mivel a marketing inkább az elméletekkel szeret foglalkozni, és nem törődik közvetlenül a vevőkkel (ez a kereskedők feladata), fokozatosan feleslegessé válik a bazári környezetben. A fogyasztói marketing elavult fogalommmá válik, valami más foglalja el a helyét, valami, ami a piaci részvétel sokkal mélyebb szintű, sokkal szervezettebb módját jelenti.

Valami ehhez fogható indult el a Microsoftnál 1996-ban, amikor a cég szokatlanul értelmes dolgot tett: minden nagyobb termékükhöz ellenőrzés nélküli Usenet hírcsoportokat hoztak létre. Látszólag a semiből eszmecsereire alkalmas övezetek jöttek létre, ahol a vevők, az alkalmazottak, vagy bárki más elmondhatta a véleményét. Egyesek úgy gondolták, hogy a Microsoft csupán újabb lehetőséget talált arra, hogy hirdesse termékeit, és meséket terjesszen azokról (azóta is gyakran vádolják ezzel a Microsoftot). Azonban az alkalmazottak is megjelentek a listán, és bátorították az őszinte véleménycserét, mert tudni szeret-

ték volna, hogyan teljesítenek programjaik alkalmazás közben. A Microsoft egyik akkori vezetője elmondta nekem, hogy ez egyszerű marketingtrükk és elárulta, meglepi, hogy versenytársai nem alkalmaznak hasonló eljárásokat.

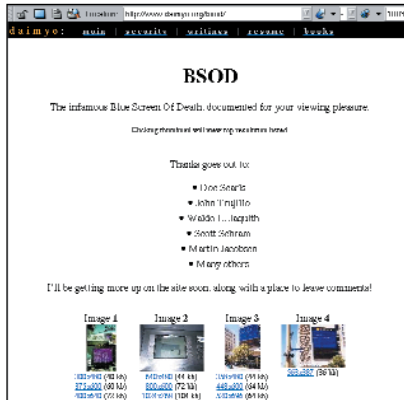


www.salon.com

Ez a marketingtrükk akkoriban még csak elmélet volt. *Theodore Levitt* azt tanította, hogy a marketing célja „a vevők igényeinek kielégítése, bármi áron”. A usenetes és a telefonos terméktámogatáshoz beérkező visszajelzések ellenére a Microsoft folytatta az összeomlásra hajlamos operációs rendszerek gyártását. Nyomós oka volt rá: milliószámba értékesítette termékeit. A pénz beszélt, a piac pedig elfogadta. 1998 elején fordult a kocka, a piac már nem csak a pénz nyelvén beszélt. Ekkor tette közzé a [salon.com](http://www.salon.com) a Haiku Error Message (Sír felirat hibáüzenet) pályázat eredményeit. Szinte azonnal, a közlést követő percekben, a nyertes jelmondatok szédületes sebességgel száguldottak a Hálózatban, megjelentek a postafiókokban és a Weben. Igaz, hogy *Peter Rothman* nem nyerte meg ezt a versenyt, de hibáüzenetei mégis görgeteget indítottak el.

A Windows NT elszállt.  
Elhoztam a Kék Halált.  
Senki sem hallja üvöltését.

A hálózat köszönhetően, mindenki üvöltése a viccek kiapadhatatlan forrásává vált. A Windows nevetség tárgya lett. Készítőinek a saját programjuk torz tükörképével kellett szembenézniük. Keress rá a Weben a „blue screen of death” kifejezésre! A megjelenő lista elején felé megtalálható *Matt Michie* oldalát (<http://www.daimyo.org/bsod/>). Matt a [Linux.com](http://www.linux.com)-nál dolgozik. A weblapján galériát nyitott, ahol bemutatja a nyilvános



www.daimyo.org/bsod/



www.linux.com

kék halál képernyőkről készült fényképeit. Láthatók itt Las Vegas-i óriástábláról, pénzkidó automatáról, reklámkielzőkről, reptéri menetrendet mutató képernyőről és kaszinóban található játékgépekről is képek. Ezek természetesen sok helyen megtalálhatók a hálózaton. A hálózati pletyka már csak ilyen. És minél nagyobb a bazár, annál gyorsabban terjednek a hírek. Eljött az a pont, amikor a zaj már elnyomott minden mást. Ez egyértelműen tavaly nyáron következett be, amikor Jim Allchin néhány barátjával egy hajón töltötte a szabadságát. Csak nemrég árulta el nekem, hogy két nagyon fontos dolgot tanult meg az út során, és mindkettő megdöbbentette. Először is megfigyelte, hogy az emberek mennyi mindenre szeretnék használni a PC-jüket. Ennek kapcsán rájött, hogy rengeteg gondot okoz nekik, hogy a PC-jüket működésképp bírják. Elmondta, hogy a tapasztalatai megalázóak voltak, ezért még eltökéltebben tenni szeretne valamit ennek megoldására. Vegyük észre a „még” szót. Ez azt jelenti, hogy már régóta próbál tenni valamit, ami nem is olyan hihetetlen, hiszen bárki bármit is mond, Jim már régi motorosnak számít a unixos világban. Mostanában megfigyelhetjük, hogy a Microsoft „öt kilences” (99,999%) megbízhatósággal hirdeti a Windows 2000 Professional-t. Az állítás vitatható, de az nem, hogy a Microsoftot foglalkoztatja a megbízhatóság.

Miért vált hirtelen fontossá a megbízhatóság? Véleményem szerint azért, mert a piac elveszti az érdeklődését a szórakoztató elektronikai cikként reklámozott PC-k iránt. 2000-ben a fogyasztók többsége rájött, hogy a tavalyi PC még idén is jó. Ugyanígy tökéletesen megfelel az Office 97, Quicken 7, a Photoshop 5.5 és a többi program (még

Egy ember sétál az utcán

*A man walks down the street*

Egy furcsa világ utcáján

*It's a street in a strange world*

Talán ez a Harmadik Világ

*Maybe it's the Third World*

Talán most látja először

*Maybe it's his first time around*

Nem beszéli a nyelvet

*He doesn't speak the language*

És pénze sincs

*He holds no currency*

Egy távoli országból érkezett

*He is a foreign man*

Minden oldalról körülveszi a hang

*He is surrounded by the sound*

A hang

*The sound*

Jószágok a piacon

*Cattle in the marketplace*

Koldusok és árvák

*Scatterlings and orphanages*

**Paul Simon**

ha nem is mindig működnek tökéletesen). Nem hagyhatjuk figyelmen kívül az operációs rendszerek szerepének ártékelődését. Régen csinos termék volt. Mára már csak alap. Egyre több ember számára a Hálózat az, ami számít és nem a PC operációs rendszere. Ott van továbbá a mindenütt jelenlévő hálózat tagadhatatlan sikere. A hálózat sokkal többet köszönhet a szabad és nyílt forráskódú programfejlesztési módszereknek (és természetesen a fejlesztőknek), mint bármelyik programokat fejlesztő és forgalmazó cégnek, amely azonnal arra törekszik, hogy magához láncolja a vevőket. A Linuxnak különösen nagy szerep jutott a fejlesztésben. A Microsoft nem hagyhatja figyelmen kívül a Linux sikerét, vagy az alapok folyamatos építésének fontosságát.

Az alapozáshoz szükséges, mindenhol jelenlévő építőelemek természetüknél fogva kereskedelmi termékek. Talán kereshetsz pénzt az árusításukkal, de a nagy nyereség máshol van. Mivel valahol minden ettől az áruba bocsátott alaptól függ, beleértve a hálózatot és a Linuxot is, stratégiaileg fontos, hogy hozzájáruljunk az építéséhez. Ha nem ezt teszed, a tudásod elavul, és megfosztod magad az újítás lehetőségétől.

Végre a Microsoft is elkezdte tervezni ezeket a dolgokat. Ezért dolgoztak együtt más cégekkel a SOAP és XML-RPC protokollokon. Ez csak csepp a tengerben, ahhoz képest, amit a linuxos közösségek gyökerező RedHat, VA Linux, Caldera és más cégek évek óta tesznek. De az elmélet végre valahára megegyezik.

Mi tehát a linuxos cégek stratégiája? Tudjuk, hogy nyereséggel el tudják adni a gépeket és a szolgáltatásokat. De el tudnak-e adni ugyanekkor nyereséggel programokat? Igen. Még hozzá úgy, hogy a nyereséget hozó termék forráskódját nem adják ki és ezzel párhuzamosan arra törekszenek, hogy minél több forrást elérhetővé tegyenek. Ezt teszi a Caldera a Volutionnal. Ezt teszi a Borland a Kylixszal. Ezt teszi a TurboLinux a TurboClusterrel. Az előbb említett cégek termékstratégiájának döntő eleme a nyílt forráskód, ennek eredményeképpen sok mindent oda kell adniuk, viszont a Volution, a Kylix és a TurboCluster termékek forráskódja nem nyílt. Egyikük sem ezzel reklámozza magát, de a tény, ettől még tény.

A hagyományosan zárt forrású cégek, mint az IBM, a Sun, a Borland, sőt, még a Microsoft is, kezdik alkalmazni a nyílt forrás elméletét (igen különböző módon, és eltérő sebességgel). Vajon muszáj-e a hagyományosan nyílt forrású cégeknek elmozdulniuk a zárt forrás irányába, ha nyereségesen akarnak programokat eladni a céges felhasználóknak? A válaszuk jelenleg: igen.

Mindannyian ismerik az érveket, melyek a nyílt forrás mellett szólnak.

Ez azt jelenti, hogy amíg valamelyik cég meg nem kockáztatja, hogy nagy árréssel nyílt forrású programokat adjon el cégeknek, addig a nagy nyereséggel járó céges üzletek zárt forrásúak maradnak, még a Nyílt Forráskód Közösségében gyökerező cégek számára is.



Doc Searls  
(doc@ssc.com) a  
Linux Journal szerkesztője és a Cluetrain  
Manifesto társszerzője.

## Önképzés

Aki ért hozzá, műveli. Aki nem ért hozzá, az tanítja. Aki már tanítani sem tudja, az ír róla.

**E**gyszer így jellemezték nekem a tanárokat és újságírókat. Az oktatók, illetve az oktatás szerepét és fontosságát azonban hiba lenne megkérdőjelezni. Mint ahogy tévedés lenne kételkedni a rejtett képességek fejleszthetőségében és a becsvágyban. Ha ezeket kihasználjuk, könnyebben léphetünk előre az érvényesülés útján. Sokan sajnos nem hisznek abban, hogy képzéssel fejlődhetnek. Pedig mit sem érnek a méregdrága tanfolyamok – még ha oly neves előadók tartják is –, ha nem követeljük meg önmagunktól a tanulást. Miért akarunk tanulni? Néhány a lehetséges válaszok közül: mert köteleztek rá, versenyképesebb képzettséghez juthatunk, szakmai fejlődésünkhöz szükséges. Ha csupán kényszernek érezzük, akkor felesleges elkezdünk. Vajon meddig marad meg ez a tudás? Csak az első vizsgáig, amire jelentkezzük, hogy megkapjuk a szükséges papírokat, amivel a cégeknél házalhatunk. De ez mind csak sajtópapír értékű, ha nincs mögötte valódi tudás. Próbáljuk meg követni a – szerintem – fontos alapelveket!

1. Elsőként olyan területet válasszunk, amely valóban érdekel! Ez létfontosságú, ugyanis ha valóban vonzó témával foglalkozunk, akár éjszakákat is áldozunk az elsajátítására, egy-egy nehezebb feladat megoldására. Ha csupán kényszerűságnak érezzük, akkor hamarabb feladjuk.
2. Az előző pont kiegészítése: mindentől függetlenül ismerjük meg a rendszer alapvető kezelését!
3. A következő szabály: sose adjuk fel! Minden csak idő és kitartás kérdése. Ha adott témához csak egy másik megoldásán keresztül vezet az út, amit esetleg nyűgnek érzünk, összpontosítsunk arra, hogy az eredeti célunkat csupán így érhetjük el.
4. Sose forgácsoljuk szét erőnket! Lehet, hogy hozzájutunk több ezer oldalnyi leíráshoz, minden témához beszerezzük a megfelelő könyveket, ha azonban nem olvassuk végig egyiket sem, akkor hiába minden, csupán elvesztegettük az időnket.
5. Próbáljunk általános rálátást szerezni a rendszerre! Rövidebb-hosszabb idő után elfordulhatunk a csak számunkra érdekes kérdésektől, alkalmazásoktól és elkezdhetjük a rendszer egészének megismerését.
6. Használjunk ki minden adatforrást! Ne szégyelljünk tanácsot kérni attól, aki fiatalabb, idősebb vagy egyszerűen „csak” sokkal tapasztaltabb nálunk. Mindenki volt kezdő!
7. Próbáljunk segíteni másoknak. Ekkor esetleg közösen juthatunk olyan megoldásra, mely mindkét (vagy akár több) fél számára hasznos.
8. Sose elégedjünk meg! Mindig forduljunk új célok felé, bővítsük tudásunkat!
9. Ne viseljük el előítélettel valami iránt csak azért, mert sokan szidják, vagy esetleg nem a legnépszerűbb alkalmazás (lásd a BIND és a djbdns, illetve a sendmail és a postfix használóinak viszályát).
10. Időről időre frissítsük fel tudásunkat. Nem szabad hagyni „berozsdásodni” jártasságunkat!

A következőkben megpróbálom bemutatni, hogy – szerintem – mi módon kell olyan adott feladatot megoldani, amivel eddig még nem találkoztunk.

Vizsgáljuk meg először a leírást. Elcsépelet, de igaz, hogy a Linux-, illetve a Unix-rendszerek kitűnően dokumentáltak. Ha nem jutunk előbbre a sűgőoldalak (manoldalak) használatával, próbáljunk kutakodni az Interneten. Először nézzük át a program honlapját. Ha ez nagyobb alkalmazás, akkor valószínűleg keresőmotor is található a honlapon. Használjuk bátran! Ha itt sem jutunk előbbre, akkor még mindig ott vannak a levelezőlisták és a hírcsoportok levéltárai. Ezek közül az egyik legismertebb a dejanews. A másik lehetőség, a keresők használata. Én a Google-t ajánlom, hiszen van külön linuxos tagozata [www.google.com/linux](http://www.google.com/linux), valamint magát a rendszert is linuxos gépeken üzemeltetik.

Ha egyik módszer sem segített, akkor keressük meg a helyi Linux-felhasználók levelezőlistáját vagy az alkalmazás készítői által üzemeltetett levelezőlistát. Ne feledjük, hogy nem mindig kész megoldást kapunk, kövessük az irányvonalakat! Ne legyünk türelmetlenek! Találkoztam olyan emberrel is, aki azért mondta le a listatagságát, mert a gondjára nem kapott betűről betűre megoldást. Ezt mégsem várhatjuk el! Ha van rá lehetőség, alakítsunk ki otthon a kipróbáláshoz megfelelő környezetet. Ez azonban nem mindig egyszerű. Ehhez nyújthat segítséget a VMWare vagy a plex86, melyekkel virtuális gépeket futtathatunk gépünkön. Miért fontos a kipróbálás? Mert előfordul, hogy az alkalmazások nem a leírásnak megfelelően viselkednek, vagy a megszerzett adatok már nem helytállóak, mert az alkalmazás időközben új képességekkel bővült. De az is megeshet, ami nemrég az egyik barátommal történt: egy kereskedelmi Unix-rendszere kellett az egyik legnagyobb adatbázis-kezelő rendszert – különleges követelményeknek eleget téve – telepíteni. A rendszerhez azonban két leírás is tartozott. Két ellentétes állítással ugyanahhoz a követelményhez kapcsolódva, amire ott és akkor szükség volt. Ilyenkor jól jön az a tapasztalat, amit a kipróbálás során szereztünk, vagy a rendszer modellezése tud nyújtani. Az összegyűjtött tapasztalatokat mindig rögzítsük és mentjük el! Nem elegendő csak a lépéseket leírni, jegyzeteljük le azt is, hogy mi miért történt! A pontos leírás nem csak itt fontos, ha ugyanis megszokjuk, akkor később könnyebb lesz üzemeltetési naplót vezetni, valamint a többi, gyakran nyűgnek tekintett, de vész esetén „életmentő” dolgot leírni.

Ha valaki hozzánk hasonló gondokkal fordul a levelezőlistánk tagjaihoz, ne habozzunk közölni, hogy mi mire jutottunk! Ez több előnnyel is jár, például így nyilvános, tőlünk független helyen is tárolásra kerül a megoldás (bár erre más módszerek is léteznek). Másik előnye, hogy esetleg javaslatunknál jobb megoldásra jutnak a lista tagjai. Ez nem is annyira hihetetlen, mint első hallásra. Lehet, hogy ők összetettebb környezetben, több nehézséggel találkozva kedvezőbb megoldást találnak. Ez számunkra is hasznos lehet.

*Deim Ágoston*

### Kapcsolódó címek

➔ <http://www.linuxdoc.org> címen minden megtalálható, amihez „hivatalos” leírás létezik a Linux-rendszerben. Itt nemcsak HOGYAN-ok, hanem hosszabb útmutatók (guide-ok) is találhatóak.

Magyar tükre a ➔ [www.geekfinder.hu/LDP](http://www.geekfinder.hu/LDP) alatt is elérhető.

➔ [www.geekfinder.hu/mlp](http://www.geekfinder.hu/mlp) és ➔ [magyar.linux.hu](http://magyar.linux.hu) a magyar leírások gyűjteménye.

## Pingvin az iskolapadban

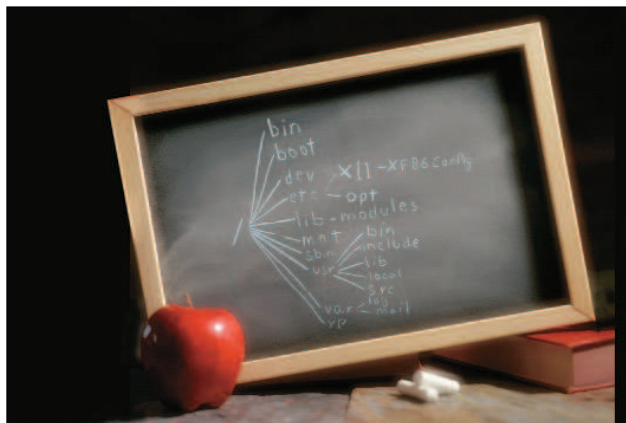
**A** világ változik és a Linuxot már nem csak a fejlesztők, és szigorúan a szakmabeliek használják. Azok számára, akik először találkoznak a rendszerrel, nehézséget okozhat megismerése és az alapok elsajátítása. Rendszergazdaként valószínűleg sokan találkoztak már ehhez hasonló helyzettel: a vezetőség rendel egy rendszert valamelyik cégtől, melynek munkatársai szépen be is állítják. Azonban senki sem ért hozzá, a pénzügyi keret viszont nem teszi lehetővé, hogy mindig a szállító céget hívják a jelentkező nehézségek miatt. Általában ekkor szokták a rendszergazdákat tanfolyamra küldeni.

Kísérletet teszek arra, hogy felvázoljam a tanfolyamok piacának jelenlegi helyzetét és annak visszasságait, valamint azt, mire figyeljünk a megfelelő tanfolyam kiválasztásánál, és végül bemutatok néhány céget, melyek többek között linuxos rendszerek oktatásával is foglalkoznak.

Az előzőekben felvázolt helyzet, a „másik” rendszeren nevelkedett rendszergazda gyötrelmeire jellemzőek. Nézzünk egy esetet, melynek kapcsán levonhatjuk a tanulságokat. Délután van, a rendszergazda nem tudja miért állt le a fájlkiszolgáló. Eddig tökéletesen működött. Mi történhetett? Kétségbeesetten telefonál a rendszer üzembe helyezőjének. Igen, holnap ki tud jönni, nem gond. Mennyibe kerül? Na, ezt nem tudjuk kifizetni! A főnök rövidre zárja a vitát, majd utasítja a szerencsétlen rendszergazdát, hogy keressen valamilyen leírást a Neten, hiszen „ott minden adat megtalálható, nem azért lett bevezetve az ISDN, hogy szórakozásra használja, végre a munkájában is segíthet”. Na igen. Az összes kereső bejárása, az összes HOGYAN letöltése, elolvasása és végrehajtása után az egyetlen eredmény: a cég ismét néhány ezer forinttal gazdagított egy monopolhelyzetben lévő céget. Ekkor a főnök határoz. Mégis kihívja a külsős céget, azok rendbe is rakják a gépet. A galiba nem szokványos volt, a kiküldött szakember is csak nemrég olvasott hasonló esetről napi ötszáz levele között, onnan ismerte a megoldást. A főnök következő lépése, hogy megbízza a rendszergazdát, keressen tanfolyamokat, ahol Linuxot oktatnak. A főnök mérlegel, majd kihívja a legolcsóbbat, mert az biztosan csak rossz lehet. A drága pedig valószínűtlenül drága, ezt nem engedheti meg a cég. Végül egy középső árcsoportba eső tanfolyamot választ. Elküldi a rendszergazdát és hátradől: lám, okosodik! A tanfolyamnak vége. Két hét múlva ismét beüt a baj. És újfent ki kell hívni a külsős céget! Hol rontották el a dolgot?

Nos, elképzeltük mennyi kellemetlen pillanatot élhetett át szegény rendszergazda! És mennyit a főnök! Állt a munka, csökkent a bevétel. A tanfolyam látszólag semmit sem ért. Mi lehet az oka? Megpróbálom röviden összefoglalni, melyek a leggyakrabban elkövetett hibák.

1. *Tanfolyamokba vetett vak hit.* A tanfolyamok nem tesznek csodát, nem lesz tőle hirtelen mindentudó a résztvevő. Nemcsak a cégek, hanem az egyéni jelentkezők is azt hiszik, hogy mindent meg lehet tanulni egy csapásra, jelentősebb megerőltetés és tapasztalatszerzés nélkül. Ez tévedés. A legfontosabb tehát, hogy ne olyan tanfolyamra menjünk el, mely gyors haladást ígér, hanem olyanra, amelyik megfontoltan halad és az oktatók meg tudják osztani mindennapi tapasztalataikat a hallgatókkal.
2. *Rossz választás.* Nem biztos, hogy a legdrágább a legjobb és a legolcsóbb a legrosszabb. Nézzünk körül a választott tanfolyam székelyén. Ez sok mindent elárulhat. Ismerősöm például jelentkezett egy olcsó angol tanfolyamra, és majdnem rászakadt a



mennyezetről a vakolat. A tanfolyam azonban még ennél is rosszabb volt, mert a szervező cég nemcsak a terembérleten takarékoskodott, hanem az oktatók képzettségén is. Kérjünk referenciákat! Ha van ismerősünk, aki már részt vett a tanfolyamon, kérjük ki a véleményét.

3. *Tanulás kényszer hatására.* Ha úgy küldünk el valakit tanfolyamra, hogy kényszernek érzi a tanulást, akkor a hatékonysága mérsékelt lesz.
4. *Saját bizonytalanságunk és félelmeink akadályként tornyosulnak.* A keleti bölcsék és az összes vallás szerint a legnagyobb feladat önmagunk legyőzése. Ha a tanfolyamon félünk kérdezni, akkor hiába koptatjuk a padot.
5. *Ne féljünk kísérletezni!* Ha a tanfolyamot okosan szervezték, úgyis biztonságos a rendszer, kárt legfeljebb magunknak okozhatunk, mert elveszhet addigi munkánk, természetesen csak akkor, ha nem mentettük azt.

Az általam egyéb gondnak minősítettek közül első helyen szerepel a Windows-alapú rendszerek azon hibája, hogy teljesen elszakítják a felhasználókat és a rendszergazdákat a rendszerek alapos ismeretétől és a nem grafikus felülettől. Így kimaradnak az életükből az olyan alapvető dolgok, mint a hálózati protokollok megismerése és a napló-fájlok elemzése. Ez rendkívül súlyos hiba, de erről szerintem nem ők tehetnek, olyan világba csöppentek, ahol minden csillog-villog, de jóformán semmibe sincs beleszólásuk, ha nincs hozzá grafikus eszközük. A második gond a valódi szakemberek hiánya. Jelenleg sokan használják úgy a Linuxot, hogy valójában nem is értenek hozzá. Esetleg egy-két HOGYAN-t elolvastak, de az összes beállítást a Webmin vagy egyéb előre megírt felületen végzik. És ezután elmennek tanítani, mert ők értenek hozzá. A tanfolyamokat szervező cégek pedig kapva kapnak utánuk is, hiszen a tanfolyamok népszerűek, bevételt jelentenek. Sok cég nem ismeri fel, ha a hallgatók egyfolytában elégedetlenek, akkor lassan rossz híruk lesz a piacon és senki nem fog jelentkezni. Az általános hozzáállás az, hogy addigra már úgyis más tanfolyam lesz keresett, akkor pedig lehet újra kezdeni, a cég neve majd Fantom2 Bt. lesz, és új titkárnőt vesznek fel. Szerencsére vannak szép számmal hozzáértők is, viszont az idejük között, hiszen számukra is csak huszonnégy órából áll a nap. Ezenkívül nem csak tanítaniuk kell, ez általában melléktevékenység vagy a támogatás kiegészítése, hiszen ha naprakészek akarnak maradni, nekik is részt kell venniük a munkában. Az eszményi megoldást nem ismerem, de ötleteim vannak. Gyanítom viszont azt is, hogy az nem fog tetszeni a cégvezetőknek. Ugyanis szerintem nem elég tanfolyamra elküldeni az embereket. Biztosítani kell számukra azt a környezetet, ahol a „vegytiszta” tantermi körülményeken kívül is tudnak gyakorolni, kísérletezni. Ez rövid távon is hasznos, mert könnyebb megszokni a rendszert, és lehet rajta kísérletezni anélkül, hogy



hiba esetén tönkremenne az egész. Ez azonban rendkívül költséges megoldás. Az újabb virtuális gépet biztosító programok valamennyit enyhítenek ezen, de áruk szintén nem kevés. A másik sarkalatos pont, hogy a cégek vezetői adjanak időt az alkalmazottaiknak. Ez a jelenlegi kiélezett versenyhelyzetben nem könnyű. Sőt, több szakember alkalmazását is igényelheti egy-egy feladat megoldása. A legnagyobb cégek ezt úgy igyekeznek megoldani, hogy egyetemekkel kötött szerződések alapján biztosítanak helyet a diákoknak gyakorlatra, illetve kutatótermetek adnak az egyetemeknek, ahol az ő módszereikkel dolgoznak, és azt fejlesztik. A cégek ily módon gondoskodhatnak a szakember utánpótlásról is. Még egy égető gond: a magyar nyelvű leírások hiánya. Ha valaki nem beszél angolul, akkor elveszett. Ezen próbál segíteni több csapat, például a Magyar Linux Projekt, ennek karbantartását jelenleg a Geekfinder csapata végzi az LME-vel karöltve.

Az önképzést akadályozó másik tényező, hogy nincs jelenleg olyan szabadon elérhető tematika, amit irányításként használhatnának a kezdők. Ebben a témakörben reményeim szerint nagy előrelépés lesz a hamarosan induló linuxoktatás.hu <http://www.linuxoktatás.hu>, ahol szabadon letölthető tematikát és egyéb segítséget találnak majd az érdeklődők a Linuxról és a kapcsolódó anyagokról.

Végezetül bemutatnék néhány céget, melyek oktatással (is) foglalkoznak. A lista természetesen nem teljes, a cégeket ábécésorrendben mutatom be, hogy elkerüljem a részrehajlást.

Azonban máris ellentmondva önmagamnak először egy ingyenes tanfolyamot szeretnék bemutatni. Ezt az LME szervezi, alkalmanként változó témakörben. A Fővárosi Pedagógiai Intézetben (Budapest, Vas utca 10., 220-as terem) folyik az oktatás. Délután egy órától már be lehet menni és megbeszélni a gondokat, tanácsadást is tartanak, az előadások pedig délután kettő órától zajlanak. Akkor lássuk a cégeket!

#### CoDe Kft.

A legnagyobb Unix-gyártók számára biztosít októkat, de önállóan is vállalnak oktatást. A megrendelő kívánsága szerint HP-ra, Compaqra vagy SCO-ra kihelyezett tanfolyamokat is indítanak. Kéthetes délelőtti vagy igény szerint egyhetes, egész napos tanfolyamokat tartanak. Az elérhető tanfolyamok:

Unix alapok és haladó témakör, időtartama 40 óra, ára 120 000 Ft;

Sysadm, azaz rendszerkarbantartó tanfolyam, egyhetes, délelőtti, 20 óra, ára 90 000 Ft;

Netadm, azaz hálózatfelügyelő tanfolyam, szintén egyhetes, 20 óra, ára 90 000 Ft;

Külső tanításnál óradíjat számolnak fel, az óradíj 15 000 forint és legfeljebb tíz fő jelenhet meg az oktatáson. A rendszerfelügyeleten kívül tartanak programozási tanfolyamokat is, C, C++ és Java témakörben. A cég az oktatáson kívül vállal karbantartást, felhasználói támogatást és egyedi hibaelhárítást is.

Honlap: <http://www.code.hu>

e-mail: [info@Code.hu](mailto:info@Code.hu)

#### HP Magyarország

Tanfolyamaikat külsős oktatók tartják, a tantermeket és a számítógépeket a HP Magyarország biztosítja. Tanfolyamaik rövidek és egész naposak, több egymást követő napon vehetünk részt rajtuk. A hallgatók létszáma legfeljebb nyolc, illetve tizenkét fő lehet. A linuxos oktatáson kívül természetesen tanulhatunk a HP-UX-ről és programozásról is. A linuxos tanfolyamokat több, összesen nyolc részre osztották fel, időtartamuk és áruk is ennek megfelelően változik. Az elérhető tanfolyamok és költségvonzataik.

Bevezetés a Linux rendszerfelügyeletbe: időtartama két nap, ára 84 000 Ft;

Linux telepítés, beállítás és felügyelet: időtartama három nap, ára 126 000 Ft;

Linux gyakorlott Windows NT rendszergazdák számára: időtartama két nap, ára 84 000 Ft;

Linux webkiszolgálók üzemeltetése: időtartama két nap, ára 96 000 Ft;

Linux levelező-kiszolgálók üzemeltetése: időtartama két nap, ára 96 000 Ft;

Linux fájl- és nyomtatókiszolgálók: időtartama két nap, ára 96 000 Ft;

Linux hibafelderítés és -elhárítás: időtartama két nap, ára 96 000 Ft;

Linux-biztonság: időtartama egy nap, ára 48 000 Ft;

Honlap: <http://www.hp.hu>

e-mail: [training@hp.hu](mailto:training@hp.hu)

#### LSC Linux Support Center Kft.

Más cégek számára biztosít októkat, de cégek számára helyszíni oktatást is vállal kiscsoportos rendszerben. Nemcsak Linux, hanem PHP, SCO Unix és NonStop telepekről is tartanak tanfolyamokat, ha igény van rá. Vállalják Linux- és SCO-rendszerek telepítését és karbantartását, valamint egyedi programfejlesztéssel is foglalkoznak. Külső tanításnál az óradíj 5-8000 forint, a tanfolyam jellegétől és erősségétől függően. A tanfolyamokon a hallgatók száma megegyezően alapul. A tanfolyamokat alapszinttől a rendszergazdai szintig végzik. Egyedi igényeket is kielégítenek.

Honlap: [www.geekfinder.hu/lsc](http://www.geekfinder.hu/lsc)

e-mail: [info@lsc.hu](mailto:info@lsc.hu)

#### Pilátus-Comp Kft.

A cég három szinten ajánl tanfolyamokat, kezdő, haladó és rendszergazdai szinten. Vállalnak kihelyezett egyéni tanfolyamokat is. Az oktatáson kívül célterületük még a rendszerkarbantartás és linuxos rendszerek telepítése és kiépítése is. A tanfolyamok árai és időtartamai a következők:

Kezdő tanfolyam, időtartama 15 óra (négy vagy öt alkalom), ára 14 000 Ft.

Haladó tanfolyam, időtartama 24 óra, hat vagy nyolc alkalom, ára 24 000 Ft.

Rendszergazdaképző tanfolyam, időtartama 30 óra, 10 alkalom, ára 32 000 Ft.

Honlap: <http://www.pilatus-comp.co.hu>

#### TradeGate Kft.

Az oktatások fő helyszíne saját belvárosi októteremük, de vállalnak kihelyezett tanfolyamokat is. A tanfolyamokat két csoportra osztották fel, az alaptól a haladó szinten át a rendszergazdai szintig. A tanfolyamok igény szerint délelőtti vagy délutániak, heti két vagy három alkalommal zajlanak. A Linux-oktatáson kívül vállalnak, illetve folyamatosan indítanak PHP és webmester tanfolyamokat, utóbbit alap és haladó szinten, több programnyelvre, illetve alkalmazásra összpontosítva. Áraik és a tanfolyamok:

Linux – az alapok: időtartama 36 óra, ára 39 900 Ft;

Linux rendszergazda: időtartama 40 óra, ára 56 900 Ft;

Honlap: <http://www.tradegate.hu>

e-mail: [info@tradegate.hu](mailto:info@tradegate.hu)

A felsorolt tanfolyamokon kívül várom más cégek jelentkezését is, akik Linuxot oktatnak, de elkerülték a figyelmemet. Az ismertetőket és elérhetőségüket – honlap és cím – a következő levélcímről küldjék: [ago@lsc.hu](mailto:ago@lsc.hu).



Deim Ágoston ([ago@lsc.hu](mailto:ago@lsc.hu))

Kedveli a sört, szereti a futást és imádja Szabó Lőrinc verseit. Nem hisz vakon egyik rendszerben sem. Vonzódik a BSD-hez is. Tagja az LME-nek és a MBE-nek. Mottója: a gép nem lehet fontosabb az embernél.

## Mire jók a tanúsítványok?

Tobin Maginnis, a Sair elnöke a tanúsítványok mellett sorakoztat fel néhány érvet.

**M**indannyian ismerjük a szállóigévé vált Shakespeare idézetet: „Lenni, vagy nem lenni. Ez itt a kérdés”. Ha a Linux-hívők közé soroljuk magunkat, előfordulhat, hogy ma is ugyanezt a kérdést tesszük fel, kissé elferdítve: „Minősítéssel, vagy anélkül. Ez itt a kérdés.” A Linux-közösség valójában megoszlik a kérdésre adott válasz kapcsán.

Az egyik tábor, melyet gyakran a „szabad élet vagy halál” szemlélettel szoktak jellemezni, úgy tartja, hogy nincs szükség tanúsítványokra. Nézőpontjuk szerint, mivel a Linux – ellentétben a fő versenytársakkal, a Microsofttal és a Novellel – nyílt forrású, az Internetről szabadon letölthető programrendszer, a vele kapcsolatos tudás is a gyakorlati használat, a vele végzett munka során szerezhető meg, így nincs szükség arra, hogy tanúsítványokkal bizonygassuk.

Ezzel szemben az ellentábor úgy tartja, hogy a tanúsítvány egyben biztosítékot is jelent. Ha egy harmadik, független fél ellenőrzi tudásunkat, és úgy találja, hogy rendelkezünk mindazokkal az ismeretekkel, amelyek egy összetett operációs rendszer üzemeltetéséhez szükségesek, az mind az informatikai szakértő, mind az őt alkalmazó cég számára előnyös lehet.

A tanúsítvány egyaránt pénzt jelent a cég és a minősítést megszerzett személy számára. A cég emellett számos további előnyt is nyújthat. Az International Data Corporation (IDC) kutatása kimutatta, hogy a Microsoft, a Novell vagy a Cisco programok használatában egyaránt jártas, és tanúsítvánnyal rendelkező informatikai szakemberek egyértelműen kedvezőbb fogadtatásra találnak munkahelyükön. Idővel ugyanez igaz lesz a minősített linuxos szakemberekre is.

A minősítést szerzett alkalmazottak hatékonyabb és termelékenyebb munkát végeznek, általában hamarabb be tudnak állítani egy asztali számítógépet vagy kiszolgálót, mint a minősítés nélküliek. A megfelelően képzett Linux-programozó képes lesz a kiszolgáló folyamatos üzemeltetésére, miközben a minősítés nélküli alkalmazott csupán próbálgatással kísérli meg a hibák elhárítását.

A minősítést szerzett munkavállaló hagyományosan kétszer annyi berendezés felügyeletére képes, mint a minősítés nélküli, és sokkal összetettebb, gondosabban kidolgozott hálózatot tud felépíteni.

A képzés költsége általában kilenc hónap alatt megtérül. Nem szabad elfeledkezni arról sem, hogy a tanúsítványok fontos előnyt jelenthetnek, amikor az informatikai vagy személyzeti vezető „keresztülüssa” magát az önéletrajzok tömegén.

A tanúsítvánnyal rendelkező munkavállalók számára elérhető előnyök közé sorolható a magasabb fizetés, a kellemesebb munkakörnyezet, a nagyobb létbiztonság és az eladható tudásból fakadó függetlenség. Mindezek a minősítés megszerzése után hamarosan kézzelfoghatóvá válnak. Az alkalmazottak szempontjai között legtöbb esetben a fizetés nagysága áll az első helyen. A felmérések kimutatták, hogy Amerikában a Linux területén szerzett tanúsítvány 5–10 dollárral magasabb órabért jelenthet tulajdonosa számára. Ez 10–20 ezer dolláros fizetésemelkedést jelent éves szinten. Egyesek úgy vélik, a tanúsítványok egyben nagyobb hibaelhárító készséget és jobb minőségű munkavégzést is jelentenek. Ennek fényében miért is választanánk a Linuxot az egyéb, kereskedelmi operációs rendszerek helyett?

A Linux napjainkra a világ második legelterjedtebb operációs rendszere lett, elterjedtségének növekedése meghaladja a többi operációs rendszer együttes eredményét. A [Survey.com](http://Survey.com) egyik nemrég,

2000 hálózaton elérhető informatikai szakember részvételével végzett kutatásában arra az eredményre jutott, hogy az elkövetkező két év során ötszáz százalékkal fog növekedni a linuxos alkalmazások fejlesztésének üteme. Az International Data Corporation kimutatásai szerint a Linux jelenleg ugyan nem jelent számottevő piacot (56 millió dollárra becsülik a 2001-es év eredményét), azonban a Linux támogatásból származó bevétele 2004-re 285 millió dollárra nőhet majd, ez pedig 86,9 százalékos átlagos éves növekedést jelent.

*Louis Gerstner*, az IBM elnöke kisebbfajta sokkot idézett elő a számítástechnika világában tavaly decemberi bejelentésével, miszerint az óriáscég idén egymilliárd dollárt kíván linuxos fejlesztésekre költeni. Ő úgy vélekedett, a Linux a legjobb mód arra, hogy megfeleljünk az Internet által támasztott követelményeknek.

Az IBM terveire remekül jelzik, milyen lesz a Linux jövője. A személyi számítógép már legalább tíz évvel korábban létezett, mint ahogy az IBM 1983-ban piacra lépett, és bevezette volna a PC-t, mely forradalmasította az iparágat. Ahogy ők mondják: a többi immár történelem. Az IBM képviselője már 1999. január 10-én nyilvánította elkötelezettségét a Linux iránt, amikor a *New York Times* egyik cikkében kijelentette, hogy a PC és az Internet után a Linux jelentheti a számítógépes iparág harmadik forradalmát.

Az IBM nem az egyetlen cég, mely Linuxot használ. Felsorakozik mögötte a Sony Electronics, a ZDNet, a Sallie Mae, a Boeing és a Compaq, csak hogy néhány ismertebb nevet említsünk. A Linux nem lesz alkalmas arra, hogy megfeleljen minden egyes cég igényeinek, ám bizonyos területeken szinte minden szervezet kihasználhatja előnyeit.

### A Sair Linux and GNU Certification

A Sair Linux and GNU Certification (SLGC) – szerintem – a legjobb ajánlatot jelenti ezen a területen. Az SLGC többszintű, gyártóktól független képzést nyújt, mely néhány röpké év alatt mintegy ezer minősített szakembert, illetve kétszáz minősített rendszergazdát bocsátott ki kapuin. A képzés meghatalmazott oktatási központokon (ACE) keresztül zajlik, többek közt a New Horizon Learning Centers és a Productivity Point International képzési hálózataiban. Vizsgázni a Thomson Prometric és a VUE vizsgaközpontokon keresztül lehet, az önképzéshez szükséges útmutatók a John Wiley & Sons Kiadó kiadványai közt találhatóak.

Mivel az oktatás gyártóktól független, a képzés pedig átfogó, anyaga legalább hét különböző Linux-változathoz nyújt hasznos segédletet. Úgy gondoljuk, ha egy adott gyártó termékének ismeretéhez kötnénk a tanúsítvány kiadását, akkor az alapvetően magának a gyártónak a sikerétől vagy bukásától függene.

Az SLGC különbözik az egyéb Linux-minősítési próbálkozásoktól, hiszen nem tesz közzé különálló vizsgákat, ehelyett tudásbázist kínál, melyet részletes célrendszer és képességlista kísér. Ez a részletes tudásbázissal kapcsolatos követelményrendszer másoknak is útmutatásul szolgálhat, ha tananyagot kívánánk készíteni vizsgáinkhoz.



*Tobin Maginnis*

([ptm@sairinc.com](mailto:ptm@sairinc.com)) a Sair Linux and GNU Certification elnök-vezérigazgatója. A számítástechnikai tudományok docense a Mississippiai Egyetemen.

## MPEG-1 filmlejátszók

Nézzük meg, hogy a legfrissebb MPEG-lejátszók elnyerték-e médiaszakértőnk tetszését.

**E**havi cikkünkben az MPEG-1 lejátszókat vesézzük ki. A mintagépbe telepítünk egy Sound Blaster Live! hangkártyát, valamint egy Hauppauge WinTV tunerkártyát.

Előzetesként annyit, hogy az aKtion, a gmpeg, a gxanim, az MPlayer, a plaympeg, az XAnim, a xine és az Xtheater telepítése valószínűleg mindenkinek sikerülni fog, azonban a többi lejátszó némelyikével már akadhatnak gondok. A BeOS-t is telepítjük, hogy összehasonlíthassuk videós képességeit a Linuxszal.

Múlt hónapban láttuk, hogy hogyan lehet egy meghajtót (akkor egy NE2000-es kártyához tartozót) magfoltozás után modulként életre kelteni. Rendszermag-moduljainkkal kapcsolatban fölmerülhet néhány, a modulok közötti feloldatlan függőségekre panaszkodó hibaiüzenet. Kíséréljük meg a foltozást ismét, ezúttal jobban figyelve ezekre a gondokra! Először is telepítjük a rendszermag forrását (melyet Debian alatt a dselect vagy az apt-get segítségével szerezhethünk be):

```
cd /usr/src
tar xvfl kernel-source-2.2.17.bz2
ln -s kernel-source-2.2.17 linux
```

Amennyiben Hauppauge WinTV tunerkártyával rendelkezünk, a 2.2-es rendszermag foltozásával érhetjük el a 2.3-as i2c modulját, amely ehhez a kártyához készült. A rendszermag fordítása előtt forrásból kell telepíteni az új i2c illesztőprogramot. Ehhez a [www.struse1007.de/bttv/faq.html](http://www.struse1007.de/bttv/faq.html) címen található bttv leírás nyújt útmutatást. Nagyon ügyeljünk a modulütközések elkerülésére:

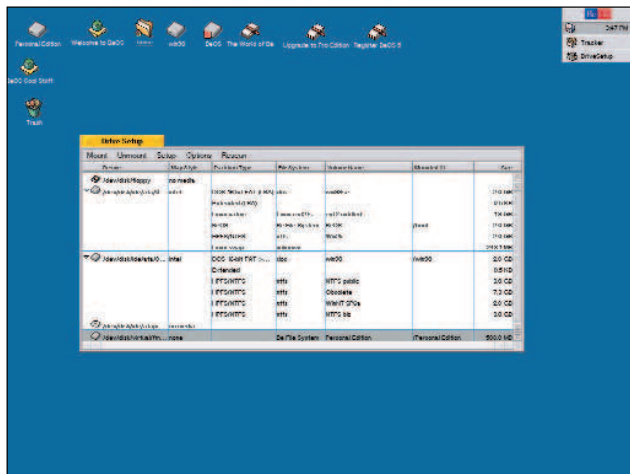
```
cd /usr/src/linux/
mv include/linux/i2c.h i2c.2.2.h
mv include/linux/i2c-id.h i2c-id.2.2.h
```

Most újraépítjük a rendszermagot és a modulokat. Semmiképpen se felejtjük el átnevezni a /lib/modules/2.2.17 könyvtárat a régi meghajtókkal való ütközés elkerülésére. Az átnevezett 2.2-es i2c fejlécek néhány elem felépítését megakadályozzák, de a make által kiírt hibaiüzenetek alapján gyorsan átírhatjuk a megfelelő fejléceket úgy, hogy azok az átnevezett 2.2-es i2c fejlécekre mutassanak. Következő lépésként felépítjük a foltozott i2c modult (mely már az új fejléceket használja):

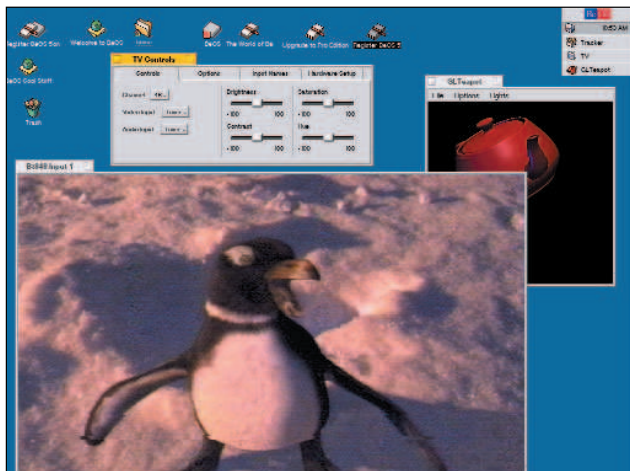
```
ln -s /usr/local/install/v41/i2c.2.5.4 i2c.2.5.4
ln -s i2c-2.5.4/kernel/i2c.h include/linux/i2c.h
ln -s i2c-2.5.4/kernel/i2c-id.h
    include/linux/i2c-id.h
ln -s /usr/local/install/v41/bttv-0.7.51
    bttv-0.7.5.1
cd i2c-2.5.4
make
make install
depmod -a
```

Ha most az `lsmod` paranccsal megvizsgáljuk rendszermag-moduljaink egészségi állapotát, akkor egyetlen „unresolved dependencies” hibaiüzenetet sem látunk.

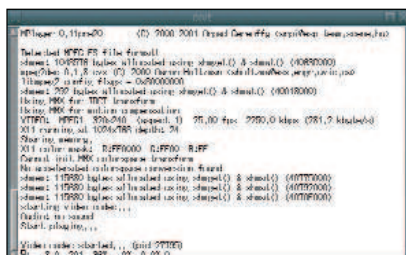
A hangtámogatás nem része a rendszermag Debianban található alapváltozatának. Ehhez új rendszermagot kell építenünk, mely az i2c foltozás végrehajtása közben le is zajlott. A Sound Blaster Live! hangkártyánk még egy különleges modul felépítését is szükségessé teszi. A rendszermag felépítésekor ne az általános Sound Blaster vagy AWE32 támogatást válasszuk, mivel a Live! másik vezérlőlapot tartalmaz, ezért az emu10k1 meghajtómodult kell használnunk. Az emu10k1-et kétféleképpen szerezhethetjük be: maga a modul az [ftp://opensource.creative.com/](http://ftp://opensource.creative.com/) címen érhető el, de az ALSA hangmeghajtó rendszer részeként is letölthetjük (<http://www.alsa-project.org/>). Nálunk mindkét módszer használatával működött, de inkább az ALSA-t érdemes választani. Az önálló modul használatakor ki kell adni a `modprobe emu10k1` parancsot, majd saját kezűleg elindítani az esd hangkiszolgálót. Az önműködő indulás beállításához azonban nem találtunk elegendő adatot a leírásban. Az ALSA telepítése is könnyebb, a leírása is bővebb és sokkal többet is tud.



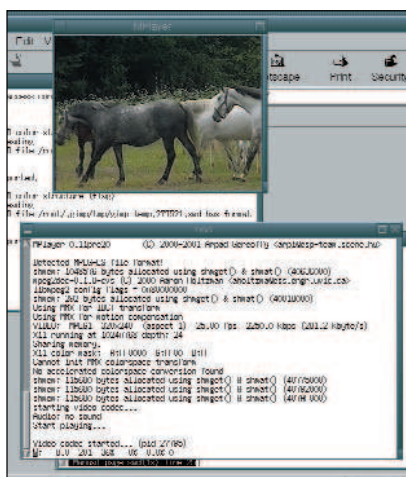
A BeOS lemezrészeli



A BeOS képernyőmentése



A megbízható MPlayerfuttatása



Az MPlayer működés közben

A telepítés befejezéséhez frissítenünk kell a /etc/modules.conf fájlt az 1. listán látható hangkártya-beállítások alapján. A Debianban viszont a modules.conf fájlt a rendszer hozza létre. A módosításokat tehát a /etc/modutils/aliases fájlba kell bejegyezni, majd innen egy hívással frissíteni a modulokat. A szükséges beállításokat világosan ismerteti az ALSA leírása, túl sok változtatásra azonban nincs is szükség.

Ezt követően a felhasználók csoportjogosultságait kell beállítanunk úgy, hogy mindenki megnyithasson médiaeszközöket. Például a forráskeverő jogosultságainak megtekintéséhez az `ls -l /dev/mixer` parancsot kell használnunk. Az eszközök csoportbeállításainak módosítása előtt ellenőrizzük, hogy a megváltoztatni kívánt elem valóban az eszköz, vagy csak egy arra mutató hivatkozás (például a /dev/video valójában a /dev/video0)! Ha az X kapcsolat során csoportváltoztatásokat végzünk, akkor utána a módosítások érvényesítéséhez teljesen ki kell lépünk az X-ből és az X-et indító konzolból.

```
chgrp cdrom /dev/hdc
adduser rower audio
adduser rower video
adduser rower cdrom
```

A hang alapértelmezés szerint ki van kapcsolva. A bekapcsolást az amixer nevű konzolos keverővel végezhetjük, mely az ALSA csomag része. Egyszerű hangpróbaként küldjünk ki egy hangfájlt az eszköze:

```
cat test.wav > /dev/dsp
```

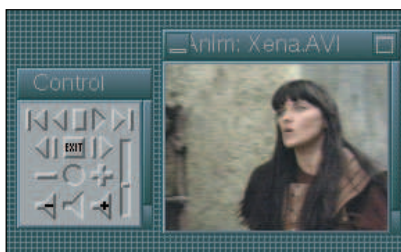
A GNOME Mixer (A GNOME Media Utilities csomag része, <http://www.gnome.org/>) és a GTK GCD-lejátszó



A KDE aKtion lejátszó



GTK gcd lejátszó



Az XAnim lejátszó

(<http://www.nostatic.org/grip/>) segítségével könnyedén játszhatunk le zenét Linux alatt. Mi nem a GNOME-ot használjuk, de a GNOME keverő így is működni fog. A GCD CD-lejátszó is tökéletesen használható.

A BeOS a magas szintű videós teljesítményéről híres, különösen, ha a BT848-as videolapkáról van szó. Ezt a Hauppauge WinTV kártyán is megtalálható lapkát használjuk a Video4Linux és az XawTV csomagokkal. Telepítsük a BeOS-t és nézzük meg, hogyan teljesít! Jé, az én gépe-men már öt operációs rendszer van!

Aki a PC-jén valami olyasmit szeretne látni, ami jobban hasonlít a Mac felületére (én nem tartozom közéjük), akkor érdemes a BeOS-t kipróbálni. Bár a BeOS a Linuxszal ellentétben nem nyílt forrású, ennek ellenére letölthetjük az ingyenesen használható Personal Edition 5 (PE5) változatot a

<http://www.be.com/cimr/>. Ez eredetileg csak egy 500 MB-os, a Windowson belül létrehozott virtuális lemezrész használatát teszi lehetővé, de mi a LILO segítségével egy 2 GB-os lemezrészre helyezhetjük. Miután a BeOS-t telepítettük Windows alatt, indítsuk újra a gépet és lépünk át BeOS-ba, ahogy azt a LoadLin segítségével tennénk!

A BeOS Macintosh-jellege azonnal látható. Néhány ikon egymás után feléled a betöltési folyamat jelzésekképpen, s a rendszer egyből grafikus üzemmódban indul. A BeOS start menü jobb felső sarokban található, de az asztalra jobb gombbal kattintva is elérhetjük. Az 1024x768-as felbontás beállításához válasszuk a *Screen-t* a *Preferences* menüből. Ha baj van, az **CTRL+ALT-SHIFT-F12** billentyűvel bármikor visszaállíthatjuk az alap-

értelmezett képernyőbeállításokat. A képeket a *Print Screen* lenyomásával a /boot/home/screen1.tga, screen2.tga stb. néven a lemezre is menthetjük.

BeOS lemezrész létrehozásához válasszuk a *Preferences* menü *Drive Setup* pontját! Készítsünk egy 2 GB-os lemezrész és a BeOS újonnan létrehozott lemezrészre helyezéséhez válasszuk az *Applications Installer*-ből az *Installer-t*! Miután a LILO-t a hagyományos DOS lemezrészekhez hasonlóan beállítottuk, máris indíthatjuk a rendszert az új 2 GB-os BeOS lemezrészről.

A BeOS BT848 meghajtó igen magas színvonalú, a grafikai rendszerhez jól illesztett eszköz. A BeOS TV ablakának mérete remegésmentesen változtatható egészen a teljes képernyős méretig. Egy szó, mint száz: a BeOS felhasználói felülete szinte a tökéletességig csiszolt, a telepítése pofonegyszerű, a képi megvalósítása pedig lenyűgöző. Néhány gond azért adódik: a tévé alkalmazásról készített BeOS képernyőmentéseken itt-ott foltok láthatók, pedig ez nem lenne törvényszerű. Ami még rosszabb, hogy indításkor néha véletlenszerű rendszermag-összeomlásokat tapasztaltunk. Fontos megjegyeznünk, hogy a BeOS PE5 csak Pentium processzorokon működik, tehát a sarokban heverő x486-oson nem fog elindulni.

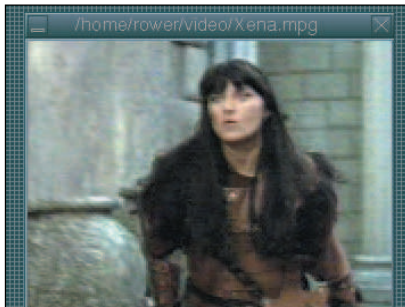
Nos, ennyit a televízióról, most nézzünk filmeket!

A filmek tömörítésére számos formátum létezik. Tömörítés nélkül a videofájlok hatalmas méretűek lennének (bár sokak szerint így is azok). A fájlok átviteli, kép- és hangsávokat tartalmaznak. Ezek közül az első szervezi egészé és írja le a másik kettőt. A kép- és hangsávok tömörítve foglalnak helyet a fájlban. Ne felejtjük el, hogy a lejátszó

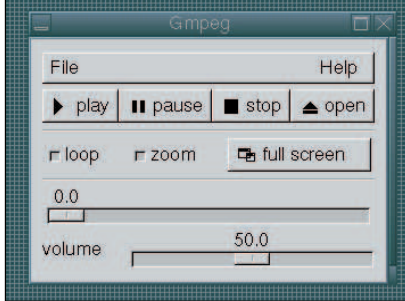
© Kiskapu Kft. Minden jog fenntartva



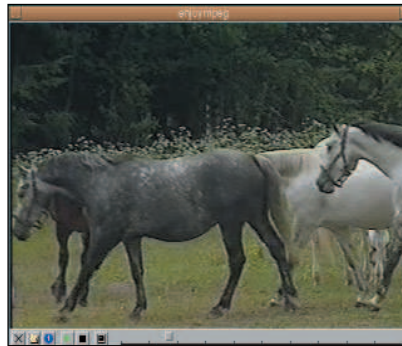
A xine lejátszó



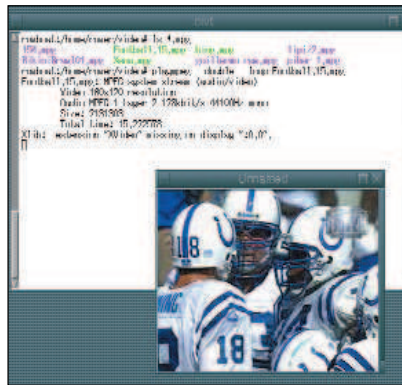
Az Enjoympeg lejátszó



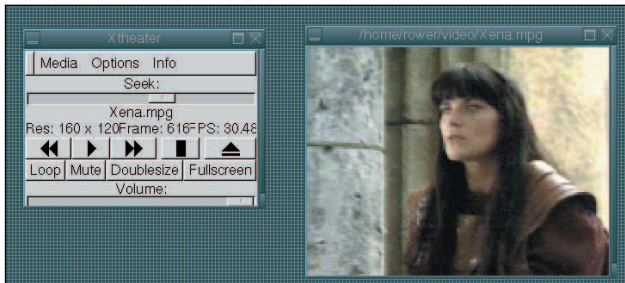
A GTK gmpeg lejátszó



A plaympeg lejátszó



Az Xtheater lejátszó



mindhárom sávot ki kell tömörítenie! Megeshet az is, hogy csak a mozgóképet látjuk, a hang pedig nem hallható a megfelelő kódoló (codec) hiánya miatt. A kép- és átviteli sáv nélküli, azaz csak hangsávot tartalmazó MPEG-1 fájlt egyébként MP3 néven ismerjük. Az MPEG (Moving Pictures Expert Group, <http://www.mpeg.org/>) talán a legszélesebb körben használt videoformátum. Három fő változata létezik: az MPEG-1, az MPEG-2 és az MPEG-4. Az első a legrégebbi, a video CD-k használják és máig ez az elfogadott képlemez-formátum Kínában (másutt általában a DVD). Az MPEG-2 a DVD-ken és a digitális műholdas sugárzásban használatos. Elődjénél sokkal jobb hang- és képminőséget biztosít, de sajnos négyszeres fájl méret mellett. Az MPEG-4 nagyjából az MPEG-2 minőségét nyújtja, viszont még az MPEG-1-nél is kisebb fájl mérettel. Az MPEG-4 egyik legismertebb változata a Microsoft DivX. Cikkünkben most csak az MPEG-1 lejátszókkal foglalkozunk. Az MPEG-1 egy eljárást használ a kép és a hang tömörítésére. Elméletben bármilyen MPEG-1 lejátszó képes bármilyen MPEG-1 fájlt lejátszani, azonban nem minden MPEG-1 fájl teljesen szabványos, így néhány, az Internetről letöltött fájl esetében sok lejátszó csodát mond.

Bár csak konzolos felülettel rendelkezik, ennek ellenére a Gereöffy Árpád és csapata által írt MPlayer-t találtuk a legkisebb igényű és legmegbízhatóbb lejátszónak. Sok MPEG-1 fájllal kipróbáltuk és mindegyiket képes volt lejátszani. A szerző honlapjáról (<http://thot.banki.hu/esp-team/MPlayer.html>) az is kiderül, hogy a program még a sérült MPEG-1 és AVI fájlokkal is elbánik, ami nem mondható el például a híres Windows Médialejátszóról. Az általunk kipróbált többi lejátszó a Debian Potatóval telepített csomagoknál frissebbekre vágyott, tehát töröltük a megjegyzéseket a `/etc/apt/sources.list` fájlból. Így a Debian csomagokat az Internet több helyén is figyelhetjük (a `.bashrc` fájlban pedig a 2. listán látható proxybeállításokat hajtottuk végre). Általában a [debian.org](http://debian.org)-ról célszerű letölteni a csomagokat, de más helyeket is megadhatunk, találhatóunk egy tükröt az [ftp.linuxvilag.hu/debian](http://ftp.linuxvilag.hu/debian) címen is. Külső fejlesztők által üzemeltetett forrásoldalakat is felvehetünk a listába, én például az OMS-t is így szereztem be. A csomagok megbízható (stable), kipróbálás alatti (testing), illetve megbízhatatlan (unstable) változatokban érhetők el, így mindenki kénye-kedve szerint egyensúlyozhat a naprakészség és a lefagyások miatti hajtépések között. A `.deb` csomagok kezelésére legalább három módszer létezik. A legismertebb ezek közül a `dselect` nevű program. De ha egy csomagot az Internetről töltöttünk le, akkor a legegyszerűbb a `dpkg`-vel telepíteni:

```
dpkg --install opera_5.0-0beta6-20010208-016_i386.deb
```

Amikor a `/etc/apt/sources.list` fájlba új, kipróbálás alatti és megbízhatatlan csomagokra mutató bejegyzéseket írtunk, akkor a `dselect` használatkor hibák keletkezhetnek. Szinte az összes telepített csomagot sikerült törölnünk, amikor a `dselect` valamelyikből frissített változatot észlelt és önműködően telepíteni kívánta azt. Érdemes tehát inkább az `apt-get` nevű programot használni. Egy kis külön interaktivitás kedvéért az alábbi sorral bővítsük az `apt.conf` fájlt:

```
// /etc/apt/apt.conf
DPkg::Pre-Install-Pkgs
{"/usr/sbin/dpkg-preconfigure --apt";};
APT::Get::Show-Upgraded "yes";
```

Az `apt-get update` parancs hatására az `apt-get` az elérhető csomagok adatbázisát frissíti. Találtunk néhány csomagot a `debian.org`-on is, de ezeket forráskód-ból kell felépíteni. E művelet mikéntje változó, de általában az alábbi parancsokra van szükség:

```
./configure
make
make install
```

Néhány lejátszó nem épül fel a hiányzó könyvtárfájlok miatt. A <http://packages.debian.org/> útmutatásai alapján az `apt-get` segítségével telepítettük a lejátszók leírásában említett szükséges elemeket:

### 1. lista A /etc/modules.conf frissítése a megfelelő hangbeállítások érdekében

```
# /etc/modutils/aliases
# A másodnevek (alias-ok) tudatják az
# insmod/modprobe programokkal,
# hogy mely modulokat kell használni.

alias sound emu10k1

# ALSA rész
alias char-major-116 snd
alias snd-card-0 snd-card-emu10k1

# OSS/Free rész
alias char-major-14 soundcore
alias sound-slot-0 snd-card-0
alias sound-service-0-0 snd-mixer-oss
alias sound-service-0-1 snd-seq-oss
alias sound-service-0-3 snd-pcm-oss
alias sound-service-0-8 snd-seq-oss
alias sound-service-0-12 snd-pcm-oss
```

```
apt-get install libc6-i686
apt-get install libgtk1.2-dev
apt-get install libstdl1.1-dev
apt-get install libqt2.2-dev
apt-get install alsa-source
```

Kipróbálásra szánt MPEG-1 fájljaink sokféle forrásból származnak, és a Windows Médialejátszója tökéletesen kezeli ezeket. Az XAnim (☞ <http://xanim.va.pubnix.com/home.html>) és változatai, a KDE aKtion (☞ <http://www.geocities.com/SiliconValley/Haven/3864/aktion.html>) és a GNOME gxanim egyik próbafájlt sem játszotta le tökéletesen. Azonban a kipróbált lejátszók közül az aKtion kezelte legjobban az AVI fájlokat. A lejátszók nagy részének elvileg az MPEG és AVI fájlokat egyaránt le kellene játszania. A xine (☞ <http://xine.sourceforge.net/>) használata kissé nehézkes, hiszen a kezelőfelületet DVD-k lejátszására tervezték. A parancssori lejátszási lehetőség is csak bosszantó, és az MPEG fájljaink nagy részét egyébként sem játszotta le a program. Az OMS (☞ <http://www.linuxvideo.org/oms/>) nevű omi\_gtk lejátszóval sem sikerült jobb eredményeket elérnünk. A GTK gmpeg lejátszó esetében (<http://gmpeg.sourceforge.net/>) a Makefile-t kellett szinte teljesen újraírni, hiszen a csomaghoz nem tartozik beállító parancsfájl. Bár a program még gyerekcipőben jár, ennek ellenére ígéretes: szinte az összes MPEG fájljt sikerült lejátszanunk vele. Az Enjoympeg ([http://people.freenet.de/for\\_Ki/](http://people.freenet.de/for_Ki/)) szép kezelőfelülettel rendelkezik és a próbafájlokat is helyesen kezelte. Azonban bármilyen más fájltypus (például AVI) megnyitása esetén a program ész nélkül lefagyott. Tűrőképesség tekintetében tehát elmarad a kevésbé tetszetős felületű MPlayer-től. A plaympeg (☞ <http://www.lokigames.com/development/smpeg.php3/>) nevű lejátszó se vezérlővel, se kezelői felülettel nem rendelkezik. Úgy tűnik, hogy e programot inkább az smpeg könyvtár kipróbálására készítették, azonban a teljesen sima átméretezési lehetősége lenyűgöző. A többi lejátszó legtöbbjénél a képméret kötött (legalábbis az XFree86 3.3.6 használatakor). Grafikus felülettel rendelkezik az

### 2. lista A /etc/apt/sources.list fájl megjegyzéseinek törlése

```
export http_proxy='http://192.168.1.1:80/'

# /etc/apt/sources.list
deb ftp://ftp.linuxvilag.hu/debian stable main contrib non-free
deb ftp://ftp.linuxvilag.hu/debian-non-US stable/non-US main contrib non-free
deb ftp://ftp.linuxvilag.hu /debian testing main contrib non-free
deb ftp://ftp.linuxvilag.hu /debian unstable main contrib non-free
deb http://http.us.debian.org/debian stable main contrib non-free
deb http://non-us.debian.org/debian-non-US stable/non-US main contrib non-free
deb http://security.debian.org stable/updates main contrib non-free
deb http://http.us.debian.org/debian testing main contrib non-free
deb-src http://http.us.debian.org/debian testing main contrib non-free
deb http://http.us.debian.org/debian unstable main contrib non-free
deb-src http://http.us.debian.org/debian unstable main contrib non-free
deb http://linuxvideo.org/oms/data/debian ./
```

Xtheater (☞ <http://xtheater.sourceforge.net/>) MPEG-lejátszó. A kezelőfelület jól használható és a lejátszás is megbízható. Néhány próbafájl azonban valamilyen nem tetszett neki, az AVI fájlokkal pedig egyáltalán nem volt hajlandó elindulni. Az Xtheater (a legtöbb MPEG-lejátszóhoz hasonlóan) 24-bites színkezelésre és az XFree86 3.3.6-ra készült. Kivételesen az MPlayer volt, ahol az XFree86Config segítségével 32-bites színmélységre kellett átváltanunk. Sok lejátszó konzolos hibáüzenettel állt le, jelezvén, hogy a rendszerben nem érhető el az XVideo YUV támogatás (ez csak az XFree86 4.x-től kezdve létezik). A gvplayt (☞ <http://www.jtang.org/gvplay/>) ki sem próbáltuk, hiszen nem képes hangot lejátszani. A lamp (☞ <http://pauillac.inria.fr/lamp/>), az MpegOrion (☞ <http://www.CrewOrion.net/>), a Sparky (☞ <http://sparky.sourceforge.net/>) és az xmovie (☞ <http://heroinex.sourceforge.net/xmovie.php3/>) lejátszókat szívesen kipróbáltuk volna, de képtelenek voltunk felépíteni ezeket. Talán majd legközelebb. A következő hónapban az AVI-lejátszókat tekintjük át, és mozgóképet is digitalizálunk a WinTV kártyával.



Robin Rowe

([robin.rowe@movieeditor.com](mailto:robin.rowe@movieeditor.com))

a MovieEditor.com internetes és televíziós videoalkalmazásokat készítő cég egyik partnere. Írásai a Dr. Dobb's Journalban, a C++ Reportban, a C/C++ Users Journalban, a Data

Based Advisorban jelentek meg és számos tanácskozási anyagában megtalálhatók. A Robin által készített programok sorában van többek közt kiszolgálóalapú videoszerkesztő rendszer, amit a Manhattan 24 óras televíziós hírcsatorna, a Time Warner New York One, illetve a kapcsolódó honlap (☞ <http://www.ny1.com/>) is használ.

## Könnyű álom (6. rész)

### A 2.4-es rendszermag hálózati védelmi lehetőségei

**A** Linux 1994 óta tartalmaz csomagszűrőt. Az elsőt még a méltán híres *Alan Cox* építette be az 1.1-es sorozatba a BSD ipfw-jének alapjain. Ezt fejlesztette tovább *Jos Vos* és még néhányan, így kerül be a 2.0-s rendszermagba. Az emberek legnagyobb része egyszerűen ipfwadm-nek hívja a kezelőprogramja alapján. 1998 közepén a 2.2-höz *Rusty Russell* és *Michael Neuling* teljesen átdolgozták a csomagszűrő alrendszerét, és elkészítették az ipchains nevű programot, mellyel a rendszermag beállításait lehet piszkálni. A rendszer jó, csak kissé rugalmatlan. Ezt látva *Rusty* 1999 közepén hozzáfogott a negyedik sorozatú Linux csomagszűrő tervezésének, melyet a 2.3.15-ös fejlesztői rendszermagba építettek be először. Azóta számos kisebb-nagyobb fejlesztésen ment át, és jelenleg a 2.4.4-ac9-ben üzembiztos, jól használható eszközzé vált. Erről szól ez a mese. Leírjuk, hogy milyen lehetőségeket ad a rendszer, hogyan használható. Nem törekszünk azonban a referencia szintű leírásra, mivel több ilyen leírás is létezik (bár ezek a leírások általában nehezen emészthetők).

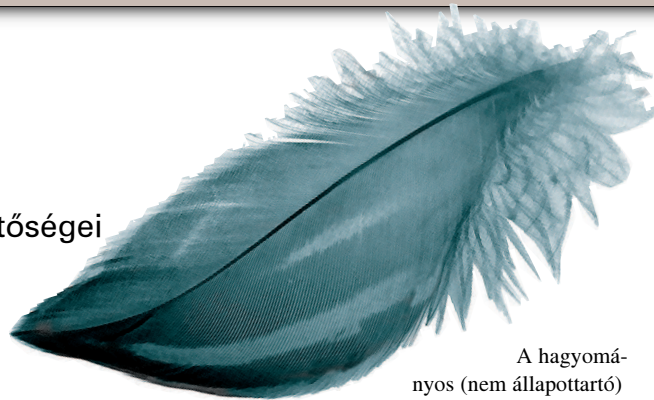
Mielőtt azonban belekezdenénk a rendszer ismertetésébe, soroljuk fel, hogy kiknek köszönhetjük a rendszer létrejöttét: *Paul „Rusty” Russell*, *Harald Welte*, *James Morris*, *Marc Boucher* (külön öröm, hogy a rendszert folyamatosan gyógyítgatják magyar varázslók is: *Kadlecsik József* és *Kis-Szabó András*). Meg kell jegyeznünk még valamit: a netfilter elsődleges célja a csomagszűrés, de nagyon fontos lehetőségei vannak ezen kívül, ami ugyan a biztonság peremterületeire visz minket, de meggyőződésünk szerint segít a biztonságosabb rendszer kialakításában. Mivel a lehetőségek meg lehetőségek bőségesek, így a sorozat ezen darabja két összefüggő részből fog állni, így teljes értékűnek csak a másikkal együtt tekinthető.

#### Néhány alapvető fogalom

A hálózati védelem helyes beállításának alapvető feltétele az, hogy tökéletesen átlássuk a hálózati kapcsolattartás működését, és az adott védelmi rendszer által nyújtott lehetőségeket. A Linux védelmi rendszerének helyes beállításához meg kell ismernünk a kapcsolódó hálózati fogalmakat, és a csomagszűrő rendszer lehetőségeit. Először igyekezzünk megvilágítani néhány későbbiekben felmerülő fogalmat. Ezek megértése szükséges ahhoz, hogy valaki helyesen tudjon beállítani egy csomagszűrőt.

#### Csomagszűrő

Csomagszűrőnek nevezzük azt a rendszert, amely képes a hálózaton közlekedő csomagokat valamilyen szempontok szerint szétválogatni jó és rossz csomagokra. A szempontokat a rendszer fejlettsége határozza meg. Minden csomagszűrő képes a csomagok feladójára és céljára alapján válogatni, a jobbkat a csomagok több tulajdonságát is megvizsgálják (IP-lehetőségek, TCP-zászlók, ethernet forráscím stb.). A Linux 2.2-es sorozata megbízható, de egy nem túl nagy tudású csomagszűrővel rendelkezik. A régebbi rendszer lehetőségeit fejlesztője szerint sem lehet egyszerűen kiterjeszteni, nem moduláris, bizonyos alapvető szükségletek ellátására nem képes, és még sorolhatnánk. Az új magba fejlesztett rendszer minden területen felülmúlja elődeit. A lehetőségei modularitásának köszönhetően kimeríthetetlenek. Ha valaki hiányol valamilyen lehetőséget, nekiállhat modult fejleszteni rá, de az alaplomdok az egyik legjobban használható csomagszűrővé teszik.



A hagyományos (nem állapotartó) csomagszűrők rossz tulajdon-

sága, hogy nem képesek a kapcsolatokat egységként kezelni, így olyan csomagokat is átengedhetnek, amelyek szabályosnak látszanak ugyan, de nem tartoznak semmilyen korábban felépített kapcsolathoz. Így hagyományos csomagszűrőn keresztül bizonyos körülmények között fel lehet építeni egy kapcsolatot úgy, hogy a forrás és a célpont nem használ SYN-es csomagot. Ez természetesen csak akkor lehetséges, ha a támadó a teljes IP-vermet lecseréli. Tétélezük fel – csak a vita kedvéért –, hogy támadónk elég elszánt és valóban belepiszkál a védett hálózat egyik gépének IP-vermébe. Ha a hagyományos csomagszűrőnk a külső területről 80-as, azaz a webkapuról érkező csomagot lát, amelyben nincs beállítva a SYN zászló, akkor joggal hiheti, hogy ez már egy élő, bentről kezdeményezett kapcsolathoz tartozó csomag. Mint ilyen, nincs vele semmi gond, így nyugodtan be lehet engedni. Ha a támadó felkészített egy belső rendszert, akkor ezzel a csomaggal akár kapcsolatot is kezdeményezhet, így akkor jár be, amikor csak akar. Mindezt azért teheti, mivel a rendszer minden csomagról különálló egységként dönt. Ha a csomagszűrő meg tudná állapítani, hogy ez a csomag egy még el sem kezdett kapcsolat része, akkor védekezhetne ellene. Ha a csomag kizárólag alkalmazásszűrő tűzfalon (a későbbiekben bővebben lesz szó róla) tudna bejutni a védett hálózatba, akkor a tűzfal azonnal eldobná a csomagot, mivel az nem szabályos.

#### Állapottartó csomagszűrő

Az állapotartó csomagszűrő (Stateful Packet Filter – SPF) olyan csomagszűrő rendszer, amely nem elégszik meg azzal, hogy a csomagokat önmagukban minősíti, hanem egy kapcsolattáblában jegyez minden szabályosan felépített kapcsolatot, és így a fenti példában megadott esetben észlelni tudja, hogy a csomagok csak látszólag helyesek, valójában nem tartoznak semmilyen szabályosan felépített kapcsolathoz.

E szűrők már ki tudják szűrni a lopakodó kapupászttázásokat (stealth scan – lásd lent), mivel azok éppen a helyesnek tűnő csomagok elvén próbálnak behatolni a védett hálózatba. Ha a pászttázáshoz különleges csomagokat használnak (Xmas vagy Null), azt egy jobb hagyományos csomagszűrő is ki tudja szűrni. Ha azonban a csomagok magukat felépült kapcsolatnak álcázva igyekeznek bejutni a védett területre, akkor azt az állapotartó csomagszűrő megakadályozza.

Az állapotartó rendszerek apró szépséghibája a hagyományossal szemben az, hogy itt már oda kell figyelni a rendszer esetleges újraindításánál, mivel a már élő kapcsolatok ilyenkor elszállhatnak. Nagy számú kapcsolat esetén előfordulhat, hogy a kapcsolattábla telítődik. Ha a sysctl támogatást befordítjuk a magba, a tábla mérete állítható (/proc/sys/net/ipv4/ip\_conntrack\_max, vagy szerencsésebb az ip\_conntrack modul 'hashsize' értékét módosítani – ip\_conntrack\_max=hashsize\*8), de DoS (vagy DDoS), támadás esetén bármekkora tábla meg fog telni.

A jobb minőségű SPF-ek a csomagok továbbítása közben le tudják cserélni azok azonosítóit (sequence number), így védve a belső hálózat rossz minőségű IP-alrendszerrel rendelkező gépeit. Az alkalmazásszintű tűzfalak ezt szükségképpen tudják, de erről a későbbiekben még szó lesz.

## Kapcsolatok lehetséges számának meghatározása (rate limiting)

Sokan gondolták már, hogy megoldották a DDoS kérdését. Láttam már ilyen magkiegészítést (patch) is. Ez természetesen a támadás jellegéből adódóan nem működhet, hiszen a DDoS esetén már a bejövő csomagok mennyisége is akkora lehet, hogy a hálózat eldugul. Ha DoS-ról van szó, akkor bizonyos esetekben lehetséges a védekezés. Csak a DoS-támadások egy része szűrhető ki, hiszen DoS az is, ha a kiszolgálót össze lehet omlasztani egyetlen jól irányzott kéréssel. Ha egy rendszert a hagyományos módszerrel próbálnak elérhetetlenné tenni, azt meg lehet akadályozni. Hogyan?

A kiszolgáló típusától függően jól leírható időminta szerint szolgálja ki ügyfeleit. A webkiszolgálótól egy ügyfél sem kérdez percenként néhány tíznél többször (a töltőgép-robotoktól eltekintve, de az már DoS-támadásnak tekinthető). Ha be tudom határolni az ügyfelektől érkező kérések számát valamilyen időszelvényben (másodperc, perc, óra, év stb.), akkor a rendszert nem lehet túlterhelni. Nyilván olyan számokat kell választani, hogy a kiszolgáló a legnagyobb terheléssel még működjön, de ne lehessen a terhelést e fölé vinni.

## Kapupásztázás (portscan)

A nyitott kapuk (port) megkeresése valamilyen módszerrel. A hagyományos módszer megpróbált felépíteni egy kapcsolatot a célgép vizsgált kapujával, így érzékelve, hogy az adott kapuban figyel-e valamilyen démon. Ebben az esetben a démonok naplózhatták a kapcsolat felépítésének a tényét, illetve a tűzfalakon az ilyen kapupásztázások nem tudtak keresztüljutni, mert olyan kapcsolatokat akartak felépíteni, amelyek nem voltak engedélyezettek. A későbbiekben egyre inkább a különleges csomagokkal vagy hagyományos csomagokkal, de különleges módon történő kapupásztázás kerül előtérbe. Ezen módszerek alapelve az, hogy egy rendszer nyitott kapuja másként viselkedik egy adott csomagra, mint egy zárt. Például egy SYN-es csomag küldése esetén a nyitott kapu szabályos SYN+ACK csomagot küld vissza, a zárt viszont RST-t, FIN csomag küldésekor a nyitott nem válaszol, a zárt RST-t küld vissza. Ha tehát ezeket a módszereket használjuk, a nyitott kapuk megállapíthatók. Mennyivel előnyösebb ez a támadó számára? Mivel a kapcsolat még nem épült fel, így a rendszer csak külön erre a célra készített eszközzel tudja érzékelni és naplózni, tehát nehezebb észrevenni. Előnye továbbá, hogy mivel nem feltétlenül kell SYN-es csomagot küldeni a kapupásztázáshoz, így bizonyos tiszta csomagszűrőkön át tud hatolni. Mivel azonban az új mag csomagszűrőjét felszerelék kapcsolat-követő (connection tracking) rendszerrel, így helyes használatával az ilyen pasztázások kivédhetők (Fjodornak, az nmap írójának őszinte sajnálatára).

## MAC cím

Az ethernethálózatok belső címzési rendszere a MAC címen alapul. Elvileg minden ethernetprotokollt ismerő eszköznek egyedi címe van. Kivéve az ötdrangú tajvani holmikak, ahol a gyártó költségcsökkentés miatt azonos címeket használt... Ha azt akarjuk, hogy az IP-cím átvételével még ne lehessen átvenni valakinek (vagy adott gépnek) a jogosultságait (hiszen a tiszta csomagszűrők szótárából hiányzik a felhasználó kifejezés), beállíthatjuk, hogy kizárólag egy adott MAC-cím tehesen valamit. Ez nem túlzottan erős feltétel,

mivel szinte minden operációs rendszer lehetővé teszi a MAC cím programból történő állítását, de azért ez is valami. Az, hogy ezt a hálózati kártyák támogatják, a Decnet protokollcsaládnak köszönhető. Köszönjük szépen, Digital!

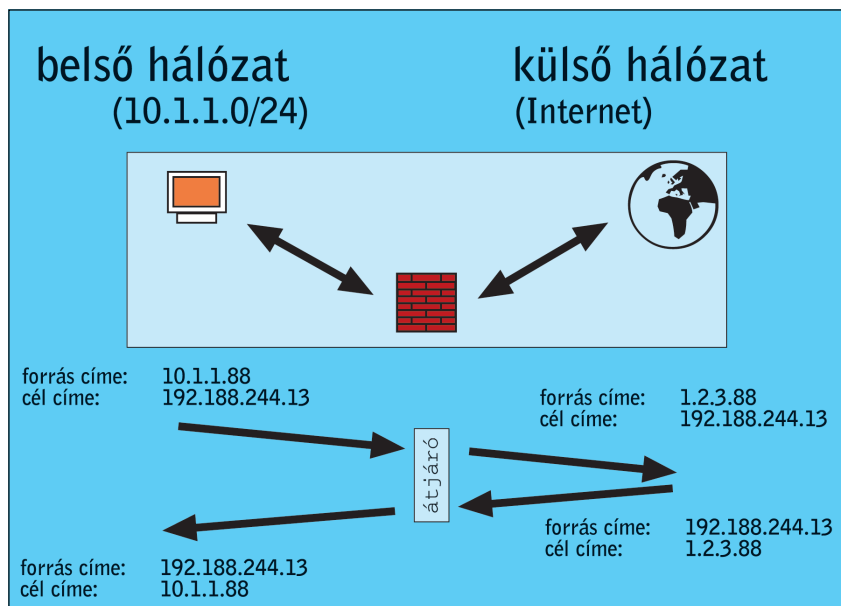
## Hálózati cím-váltás

A hálózati cím-váltás (Network Address Translation – NAT) azt jelenti, hogy az útválasztó két hálózat között úgy viszi át a csomagokat, hogy közben azok forrás-, vagy célcímét módosítja. A NAT folyamata:

- a csomag IP-fejlécében a cél-, illetve forráscímek szükség esetén átiródnak,
- a csomag UDP – vagy TCP – fejlécében a kapucímek szükség esetén átiródnak (PAT, lásd később),
- ha a kapuk megváltoztak, a TCP-fejléc ellenőrzőösszeg mezője újraszámítódik,
- az IP-csomag ellenőrzőösszeg mezője újraszámítódik,
- a rendszer további jó utat kíván a csomagnak.

Ha a csomag valamelyik jellemzőjét megváltoztatjuk, hogyan talál vissza az eredeti feladóhoz, vagy hogyan talál oda a címzettjéhez? Úgy, hogy a cím-váltó a két hálózat között helyezkedik el, és az egyik irányba menő csomagok módosításait visszafordítja. Ha például a belső hálózatunk címe 10.1.1.0/24, és ezt fordítjuk Interneten is használható 1.2.3.0/24 (Ez csak példa!) címtartományra, akkor a folyamat a következőképpen zajlik (1. ábra):

- a kimenő csomag megérkezik az útválasztóhoz, az észleli, hogy a csomag 10.1.1.88-as forráscímmel igyekszik kijutni a hálózatból,



1. ábra Hálózati cím-váltás (NAT)

- megállítja és kicseréli a forráscímét 1.2.3.88-ra,
- továbbküldi a külső hálózati csatolón keresztül,
- a válaszcsoport az 1.2.3.88-as célcímre fog érkezni a külső láb felől, ezt az útválasztó észleli,
- a csomagot megállítja, a célcímét lecseréli, és továbbítja a belső láb felé és így tovább...

A NAT szükségességének oka lehet például, hogy az egyik hálózat felől a másikat egyetlen címűnek szeretnénk láttatni (álcázás, átirányítás, terhelés-elosztás), vagy a hálózatok valós címtartományát kell eltolni (például mindkét hálózatnak 10.1.1.0/24 címe van). Elmondható, hogy a NAT leggyakrabban IP-címeket [1.] alakít külső hálózati



ton is használhatóvá, e címeket ugyanis az útválasztók nem továbbíthatják az Internetre. Így ilyen magánhálózatok kívülről való elérését a legegyszerűbb valamilyen címfordító eszköz segítségével megoldani. Lehetséges alkalmazásszintű átjáróval is (később lesz róla szó), de az szintén NAT feladatot lát el. A továbbiakban a következő szakszavakat fogjuk használni: a belső (local) cím cserélődik ki egy külső (global) címre és viszont. Lehet sok-sok és sok-sok-egy típusú NAT. Alapvetően kétféle NAT létezik: statikus és a dinamikus. Ezeket a kifejezéseket használják a kapcsolódó témájú RFC-k [2.][3.], de mivel a fejlesztőknek ez valamiért nem tetszett, a netfilter leírásai más felosztást használnak. Először megmutatjuk, hogy mit jelent a statikus és dinamikus fogalom ebben az esetben, a későbbiekben ezt megfeleltetjük a netfilter fogalmainak.

**Statikus NAT**

Ahogy a neve is mutatja, merev egy az egyes címmegfeleltetést jelent a külső és a belső címtartományok között. A fordítandó címtartománynak legalább annyi címmel kell rendelkeznie, amennyi az aktív belső címek száma.

**Dinamikus NAT**

A belső címek csoportját fordítja a külső címek csoportjára. Itt csak az a kitétel, hogy legalább annyi cím legyen a fordítandó címtartományban, ahány az egyszerre működő belső címek száma.

**Kapuváltás**

A kapuváltás (Port Address Translation – PAT) azt jelenti, hogy a PAT-ot végző eszköz a TCP-, vagy UDP-csomagok forrás-, illetve célkapuit módosítja.

**DNAT IP-tömörítéssel**

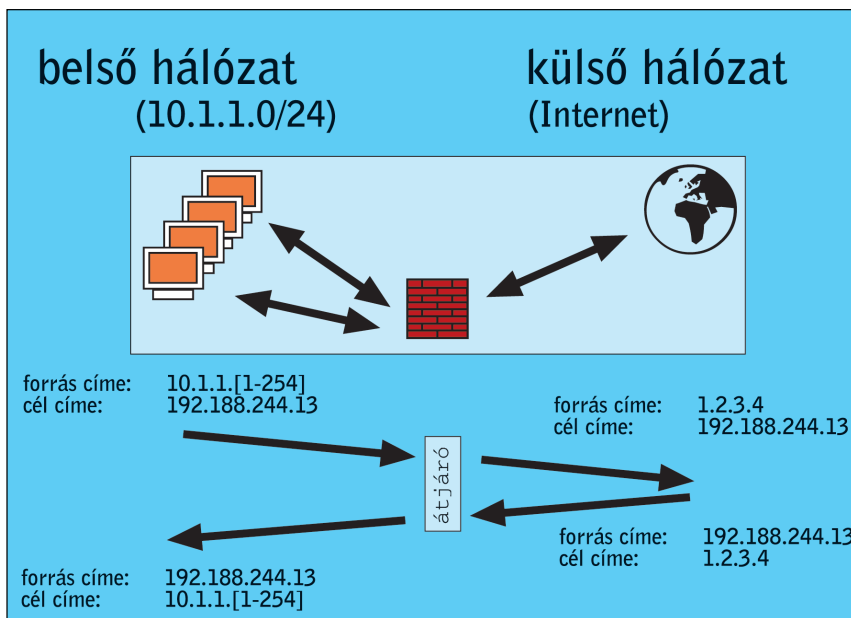
A PAT-ra például akkor lehet szükség, ha nem áll rendelkezésre annyi külső cím, amennyi az egyszerre működő belső címek száma. Itt a PAT-ot végző eszköz általában a forrás címét és kapuját módosítja, így lehetővé válik több belső címről érkező kapcsolat akár egyetlen külső címen láttatása (2. ábra). Ilyen eset a 2.2-es rendszermag jól ismert álcázás (masquerade) lehetősége.

**DNAT külső csatoló többszöröséssel (Interface Redundancy)**

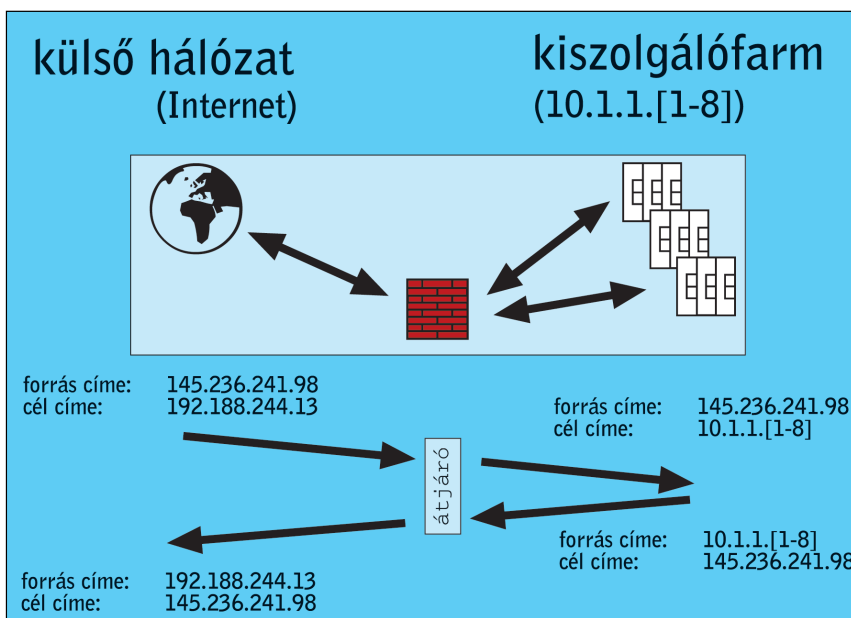
Ha a rendszernek több külső csatolója van, akkor lehetséges, hogy a belső címeket a külső címek olyan csoportjaira képezzük le, amelyek lehetővé teszik, hogy kihasználjuk a több külső csatoló adta nagyobb sávszélességet. Ilyen eset lehet, ha több internetes kapcsolattunk van, és szükség esetén növelni akarjuk a sávszélességet.

**Kapcsolat eltérítése (redirection) SNAT + PAT**

Előfordulhat, hogy a kapcsolatokat az eredeti céltől eltérő irányba kell továbbítani. Ilyen eset lehet, ha a rendszer egyetlen valós, külső hálózati címen szeretnénk több szolgáltatást is láttatni, melyek valójában több gépen futnak.



2. ábra Álcázás (DNAT IP-tömörítéssel)



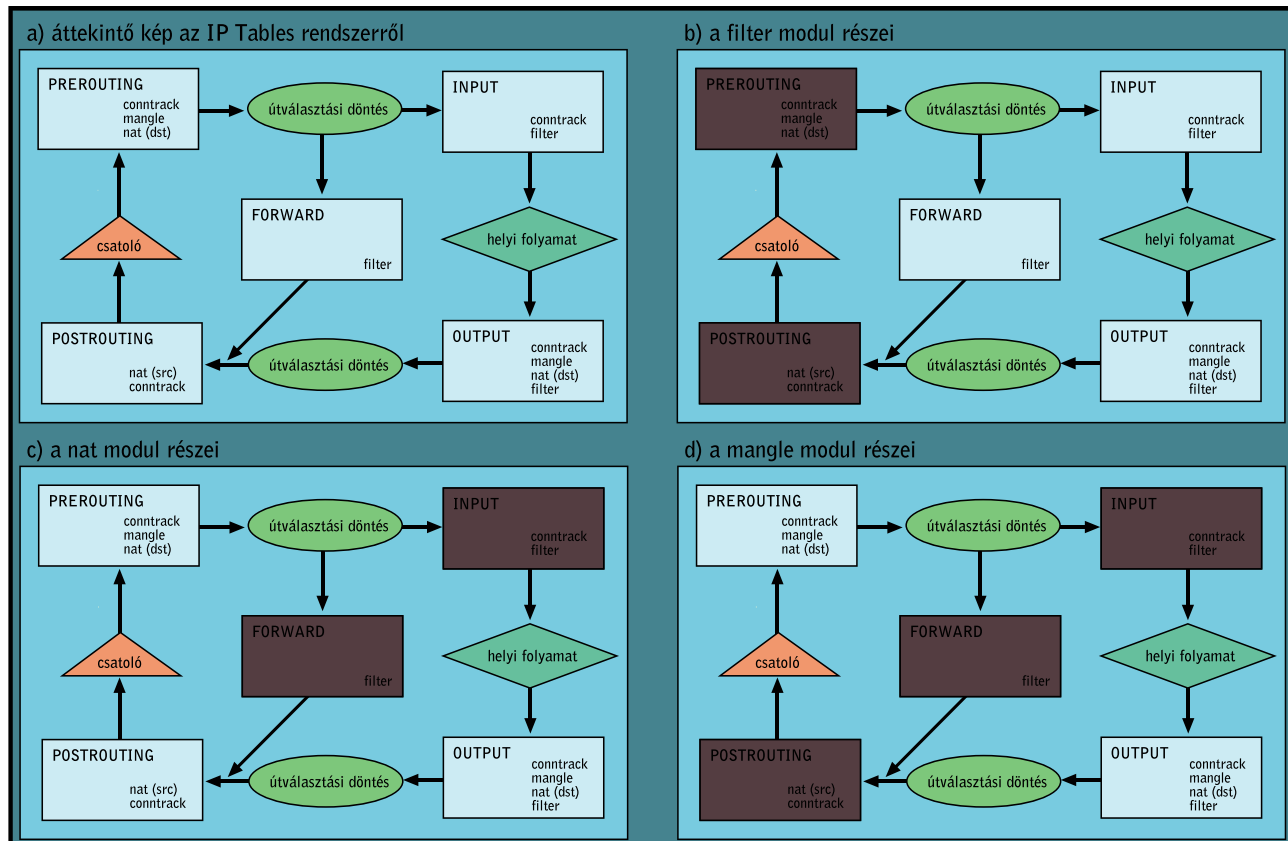
3. ábra Terheléelosztás (DNAT, PAT)

**Terheléelosztás**

A terheléelosztás (Load balance) különleges kapcsolateltérítés, ahol a cél több gép. Így egy erősen terhelt szolgáltatást egy kiszolgálócsoporthoz visz oly módon, hogy a külvilág számára azok mindegyike egyetlen címen látszik (3. ábra).

**Átlátszó proxy**

Az átlátszó proxy (Transparent proxy) korszerű hálózati határvédelmi eszközök által támogatott módszer. Azt jelenti, hogy a belső hálózat gépei számára nem látható, köztük és az igénybe vett kiszolgáló között bármilyen kapcsolatellenőrző eszköz lehet. Használata esetén nincs szükség arra, hogy az ügyfélprogram támogassa a tűzfalon keresztül történő kapcsolateltérítést. A cikksorozat negyedik részében erről már bővebben írtunk.



4. ábra Az IP Tables rendszer és annak részei

## A netfilter működésének alapelvei

A netfilter működési elve a Unix filozófiáját követi. Modulokból áll, amelyek célja a hálózaton közlekedő csomagok módosítása. Ez azt jelenti, hogy módosíthatók a csomagok jellemzői, szükség esetén a rendszer bele is szólhat a kapcsolatba. Mivel a netfilter nem más, mint általános keretrendszer csomagok módosítására, így elvileg tetszőleges hálózati protokollal tud dolgozni. Jelenleg még csak az IPv4 protokollt támogató modulok vannak készen és bizonyos részek az IPv6 kezelésére, de elvileg lehetőség van akár IPX vagy NetBEUI szűrők elkészítésére is. Más kérdés, hogy ezek feltehetőleg soha nem készülnek el. Az IP-kezelő alrendszer az IP Tables és a hozzá tartozó alkalmazásszintű eszköz neve iptables.

Az IP Tables szűrőkből (chain) áll, ezek a szűrők szabályokat tartalmaznak, melyek egy feltételből és egy műveletből állnak. A feltétel lehet összetett, és a csomag valamely jellemzőjét vagy kapcsolatának állapotát vizsgálja. A hozzá rendelt műveletben pedig meghatározzuk, hogy a feltétel teljesülése esetén mit csináljon a rendszer.

Az IP Tables felépítése a 4. ábrán látható. A rendszer teljesen moduláris, így annak egyes részei csak akkor élnek, ha arra szükség van. Az ábrán a rendszer egyes részein feltüntettük, hogy melyik modulnak részei. A teljes rendszerben a csomag életútja a következő módon néz ki:

A hálózati csatolókról bejövő csomagok először a PREROUTING szűrőbe esnek be, ezután a rendszermag hálózati útvalasztója eldönti, hogy a csomag helyi vagy továbbítandó (esetleg nem továbbítható – ebben az esetben azonban „network unreachable icmp” csomagot küld vissza). Ha a csomag *helyi*, akkor az INPUT szűrőn keresztül helyi folyamathoz kerül a benne lévő adat. Ha *továbbítandó* és a csomagtovábbítás engedélyezett (ip\_forward), akkor a FORWARD szűrőn keresztül halad tovább és a POSTROUTING-on át jut a háló-

zati csatolóra. A helyi folyamatok által feladott csomagok az OUTPUT után szintén a POSTROUTING szűrőn át kerülnek a csatolóra. A pontosság kedvéért el kell mondanunk, hogy ezek a szűrők nem mindig állnak összefüggésben egymással. A rendszer bizonyos szolgáltatásai használják, mások pedig nem. Az ábra arra fekteti a hangsúlyt, hogy áttekinthető képet adjon, az egyes tábláknak csak bizonyos szűrők részei, de ha látjuk, miként mozog a csomag a rendszerben, akkor érteni fogjuk a szűrők egymásra gyakorolt hatását is.

A rendszer jelenleg három alapvető részt tartalmaz: a filter, a NAT és a mangle modulokat. Minden modul tartalmaz bizonyos szűrőket (előfordul átfedés is), amelyekbe a szabályokat a felhasználó beillesztheti. A szabályok feltételei szinte korlátozás nélkül használhatók (ilyen szabály lehet például az, hogy mi egy csomag forráscíme) minden szűrőben. Azt, hogy egy-egy szűrő melyik alrendszerhez tartozik, az határozza meg, hogy milyen műveletet rendelhetünk az adott szabályhoz. Míg csomagot eldobni (DROP) minden szűrőben lehet, addig a csomag célját megváltoztatni (REDIRECT) kizárólag a NAT-modul által használt PREROUTING és OUTPUT szűrőkben lehetséges.

A szűrő alrendszer tiszta csomagszűrő rendszer kifinomult szűrési lehetőségekkel. A 2.2-es sorozatú rendszermagokban lévő csomagszűrő (ipchains) hasonló tulajdonságokkal bírt, bár a források teljes újraírása mellett sok tulajdonság megerősödött. Az INPUT, a FORWARD és az OUTPUT beépített szűrőket tartalmazza. Célja a bejövő, áthaladó és kimenő csomagok szűrése.

A NAT-alrendszer célja a hálózati címfordítás megvalósítása. A szűrői olyan műveleteket támogatnak, amelyek segítségével a csomagok forrás-, és célcímei módosíthatók.

A mangle-alrendszer általános támogatást nyújt az útvalasztás előtti csomagmódosításhoz. A csomagokat megjelölhetjük, ezt az útvalasztók a későbbiekben figyelembe veszik.

## Az IP Tables szűrőinek (chain) működése

A szűrőkben szabályok érvényesülnek. A szabályok felülről lefelé kiértékelődnek, és egy csomag vizsgálata addig tart, amíg valamelyik szabályban megadott feltétel nem bizonyul igaznak a csomagra nézve, és a szabályban meghatározott művelet (TARGET) nem utasítja megállásra a szűrőt. A kijelentésből látszik, hogy kétféle művelet lehetséges: egy amelyik lezárja a vizsgálatot, és egy amelyik nem. A működése példán keresztül érthetőbbé válik. Adott egy csomag, a 10.1.1.1-es (a továbbiakban egyes) címről tart a 10.1.2.2-es (a továbbiakban kettes) címre. Ha a rendszerünk a két gép között tűzfalozolgáltatást lát el, akkor a korábbiakban vázolt minta szerint a csomag áthalad a FORWARD szűrőn. A szűrőben az alábbi szabályok érvényesek:

1. szabály >> feltétel: a csomag forrása a 10.1.1.5-ös gép?  
művelet: továbbítás elutasítva
2. szabály >> feltétel: a csomag forrása a 10.1.1.0-s hálózat?  
művelet: naplózni
3. szabály >> feltétel: a csomag forrása a 10.1.1.0-s hálózat?  
művelet: továbbítás engedélyezve
4. szabály >> feltétel: nincs feltétel  
művelet: naplózni
5. szabály >> feltétel: nincs feltétel  
művelet: továbbítás elutasítva

A fenti szabályrendszer esetén az adott csomaggal a következő történik: az első szabály által megadott feltételnek nem felel meg, így a szabály művelete sem hajtódik végre. A második szabály illik rá. A szabályhoz tartozó utasítás megkéri a rendszert, hogy a csomag jellemzőit a rendszer naplóállományában rögzítse (LOG művelet). Ez az művelet nem állítja meg a csomag további vizsgálatát, mivel arról nem hozott egyértelmű döntést, hogy az továbbítható-e vagy sem. A harmadik szabály ugyanazzal a feltétellel már világos döntést hoz a csomag sorsáról: továbbítható (ACCEPT művelet). Amennyiben a csomag jellemzői nem felelnek meg egyik korábbi feltételnek sem, akkor az utolsó szabály célszerűen az, hogy minden más csomagot naplózás után elutasít (DROP vagy REJECT művelet). Ennek jelentőségéről bővebben írtunk a cikkso-rozat negyedik részében. Mivel a szűrőnek minden esetben valamilyen világos művelettel kell végződnie, így minden beépített szűrő rendelkezik egy végső művelettel, amely meghatározza, hogy mi történjen az addig le nem kezelt csomagokkal. Ezt hívják a szűrő alapértelmezett szabályának (default policy).

Felmerülhet a kérdés, miért fogalmaztunk úgy: minden beépített szűrő. Talán van valami egyéb is? Nos igen. Az IP Tables örökölte az ipchains egyik nagyon hasznos tulajdonságát: a felhasználó is meghatározhat szűrőket, és szükség esetén a csomagot átadhatja ennek a szűrőnek vizsgálatra. A felhasználói szűrőknek nem lehet alapértelmezett szabálya, ha a csomagra egyik szabály feltételei sem illenek, akkor a csomag ellenőrzése a felhasználói szűrőt meghívó utáni szabállyal folytatódik. Ha egy ilyen szűrőből vissza szeretnénk térni a meghívó szűrőbe, akkor a RETURN célt kell használnunk. Lehetséges újabb saját szűrők meghívása akár egymásba ágyazva is, de ha ciklus alakul ki (tehát a csomag visszaér a kiindulóponttra), akkor a rendszer eldobja a csomagot.

A szűrőket a következőképpen célszerű használni: eldöntjük, hogy milyen feltételekkel szabad a szűrőnek átengednie a csomagot. A feltételeket a később ismertetett formában átírjuk IP Tables által is értelmezhető feltételekké. Amennyiben valamennyi feltételünk teljesül, a csomagot elfogadjuk, vagy visszatérünk a hívó szűrőbe. Ha a csomagra nem teljesül egyik feltételünk sem, akkor naplózunk és elutasítjuk. A szűrők utolsó szabálya tehát általában feltétel nélküli tiltás. Amennyiben nem így teszünk, akkor az esetleges feledékenységgel (elfelejtünk tiltani valamilyen forgalmat) szívárgást idézhet elő.

A beépített szűrők alapértelmezett szabálya csak ACCEPT vagy DROP lehet. Ha egy beépített szűrőben RETURN műveletet hajtunk végre, az alapértelmezett szabály lép életbe.

## Az IP Tables által értelmezett feltételek

A rendszer lehetővé teszi nagyon finoman hangolható szabályok leírását. Az alábbiakban röviden bemutatjuk, milyen feltételek alapján vizsgálhatunk meg egy csomagot. Ahol megadható a feltételben felkiáltójel, ott a feltétel tagadását jelenti.

Tehát a -s 10.1.1.1 azt jelenti, hogy a csomag forrása a 10.1.1.1 című gép, míg a -s ! 10.1.1.1 azt, hogy a forrás nem a 10.1.1.1. A feltételek együtt is használhatók, ekkor a feltétel az egyes alfeltételek logikai és kapcsolata. Így ha adott a következő feltétel: -s 10.1.1.1 -p TCP, az azt jelenti, hogy a csomag TCP protokollt tartalmaz és a forrás-címe 10.1.1.1. A rendszer kétféle feltételt különböztet meg. A belső feltételek modul betöltése nélkül használhatók, a kiterjesztett feltételeket csak a megfelelő modul betöltése után használhatjuk.

### Belső feltételek

-p | --protocol [!] protokoll

Megadhatjuk vele, hogy a csomag IP-kerete milyen protokollazonosítót tartalmazhat (erről a sorozat harmadik részében olvashattak).

-s | --source [!] cím[/maszk]

-d | --destination [!] cím[/maszk]

Segítségével a csomag forrás- és cél címét határozhatjuk meg.

-i | --in-interface [!] [csatolónév]

-o | --out-interface [!] [csatolónév]

Megvizsgálhatjuk, hogy a csomag mely csatolón jött be, és melyiken fog távozni. Amennyiben a csatoló nevének végén '+' van, akkor minden olyan csatolóra illeszkedni fog a feltétel, amelynek így kezdődik a neve.

[!] -f | --fragment

Igaz, ha a csomag töredezett és nem az első töredék. Ha a "!" is meg van adva, akkor a töredezett csomagnak csak az első darabjára és a nem töredezett csomagokra ad vissza igazat. Kapcsolatkövetés (connection tracking) esetén – amit a nat használata automatikusan bekapcsol – nincs értelme, mert a rendszer ilyen esetekben törede-zettségmentesíti a csomagokat.

### Kiterjesztett feltételek

Ezeket a feltételeket általában modulok tartalmazzák, melyeket használatuk előtt be kell tölteni. Ez kétféleképpen lehetséges: közvetlenül a "--match" kapcsoló után megadott modulnévvel, vagy közvetett úton, a protokoll meghatározásával (--protocol kapcsoló), mivel ilyenkor az adott protokollra használható feltételek használhatók. Az alábbi felsorolásban először a tartalmazó modul nevét adjuk meg, utána a használható feltételeket.

**tcp modul:** TCP protokoll esetén alkalmazható

--source-port [!] [port[:port]]

--destination-port [!] [port[:port]]

A forrás és a cél TCP-kapuját lehet behatárolni vele. Amennyiben két kapu is meg van adva kettősponttal elválasztva, akkor a két szám között lévő (beleértve a két szélsőt is) kapukra illik majd a feltétel. Használható még a \_\_sport\_ és a \_\_dport\_ rövidítés is.

--tcp-flags [!] mask comp

Igaz, ha a csomag TCP lehetőségei a meghatározott mintának megfelelnek. Mindkét érték a következőkből áll: SYN ACK FIN RST URG PSH ALL NONE. Az első érték megadja, a megvizsgálandó zászókat, a második pedig meghatározza, hogy a megadottak közül melyiknek kell beállítva lenni.

[!] --syn

Csak azokra a TCP-csomagokra igaz, melyekben a SYN bit be van állítva, de a FIN és az ACK nincs. Ezek azok a csomagok, melyek a TCP-kapcsolat felépítését kezdeményezik. Ha ezeket a csomagokat egy bizonyos hálózathoz kilitjük, akkor oda hagyományos módon nem lehet kapcsolatot felépíteni. Más kérdés, hogy készíthető olyan trükkös eszköz, amely így is megoldja. Használata egyenértékű az előzőleg említett lehetőség következő használatával:

```
--tcp-flags SYN,RST,ACK SYN
```

```
--tcp-option [!] number
```

Akkor igaz, ha az adott TCP-lehetőség (option) be van állítva.

**udp modul:** UDP protokoll esetén

```
--source-port [!] [port[:port]]
```

```
--destination-port [!] [port[:port]]
```

Az UDP-csomag forrás-, és célkapuját határozhatjuk meg vele.

Ha két kaput adunk meg, akkor ugyanúgy viselkedik, mint a TCP hasonló lehetősége.

**icmp modul:** icmp protokollt vivő csomagokra

```
--icmp-type [!] typename
```

Az icmp csomag típusát határozhatjuk meg vele. A típus megadható számmal, vagy a típusnévvel, amit az iptables `-p icmp -h` parancs listáz ki.

**mac modul:** a csomagok ethernetcímének meghatározására

```
--mac-source [!] address
```

Igaz, ha a csomag ethernetforrása az adott cím. A cím megadásának formátuma a következő: `XX:XX:XX:XX:XX:XX`, ahol minden X egy hexadecimális számjegy. Csak a PREROUTING, FORWARD vagy az INPUT szűrőkben van értelme.

**limit modul:** egy szabály illeszkedésének lehetséges száma szabályozható a segítségével. Mivel működése viszonylag összetett, és a rendszerleírása [5.] részletesen ismerteti, így annak részletes leírásától itt eltekintünk.

```
--limit rate
```

Adott időtartományban meghatározza a lehetséges illeszkedések számát. A tartomány lehet „second”, „minute”, „hour”, „day”.

```
--limit-burst number
```

**multiport modul:** A feltételben több kapu megadására nyújt lehetőséget. (Kevesebb, mint 16.)

```
--source-port [port[,port]]
```

```
--destination-port [port[,port]]
```

A forrás- vagy célkapuk meghatározására használható. Több nem összefüggő kapu is megadható.

```
--port [port[,port]]
```

Akkor illeszkedik, ha a forrás- és célkapu egyenlő valamint a felsoroltak között van.

**mark modul:** a korábban jelölt csomagok felismerésére.

```
--mark value[/mask]
```

Azokra a csomagokra illeszkedik, melyek a megadott módon vannak jelölve. Ha a mask meg van adva, akkor a jelölés logikai és művelet útján kerül csak összehasonlításra.

**owner modul:** a helyben keletkezett csomagok létrehozóját lehet ellenőrizni vele. Csak az OUTPUT szűrőben használható.

```
--uid-owner userid
```

```
--gid-owner groupid
```

Illeszkedik, ha a csomagot a megadott felhasználói vagy csoportazonosítóval rendelkező folyamat hozta létre.

```
--pid-owner processid
```

Illeszkedik, ha a kibocsátó folyamat azonosítója a megadott.

```
--sid-owner sessionid
```

Illeszkedik, ha a csomagot létrehozó folyamat a megadott munkamenet-azonosítóval rendelkezik.

**state modul:** a modul a kapcsolatkövetéssel (connection tracking) együtt használva a csomag állapotát határozza meg annak kapcsolatának ismeretében.

```
--state state[,state]
```

Illeszkedik, ha a csomag állapota a megadott listában szerepel.

Érvényes állapotok: INVALID – nincs ismert kapcsolat, ESTABLISHED – ismert kapcsolat része, NEW – új kapcsolatot kezdeményez, RELATED – valamilyen már felépült kapcsolathoz tartozik (például FTP) vagy egy kapcsolathoz tartozó ICMP üzenet.

**unclean modul:** nincsenek választható lehetőségei. A hibás csomagokra illeszkedik.

**tos modul:** az IP-fejléc Type of Service mezőjének ellenőrzésére.

```
--tos tos
```

A szolgáltatás típusának meghatározására. A típus megadható számmal, vagy névvel (az iptables `-m tos -h` parancs kiírja).

**ttl modul:** az IP-fejléc ttl mezőjének ellenőrzésére.

```
--ttl ttl
```

Illeszkedik a megadott ttl értékre.

**tcpmss modul:** a TCP SYN-es csomagok MSS mezőjének vizsgálatát teszi lehetővé, ez határozza meg a kapcsolat során használható lehető legnagyobb csomagméretet.

A cikk folytatása a Linuxvilág következő számában lesz olvasható.

## Irodalomjegyzék

- [1.] (RFC 1918) Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, E. Lear: Address Allocation for Private Internets (1996. február)
- [2.] (RFC 2663) P. Srisuresh, M. Holdrege: IP Network Address Translator (NAT) Terminology and Considerations (1999. augusztus)
- [3.] (RFC 2993) T. Hain: Architectural Implications of NAT (2000. november)
- [4.] (RFC 3022) P. Srisuresh, K. Egevang: Traditional IP Network Address Translator (Traditional NAT) (2001. január)
- [5.] Paul 'Rusty' Russel: IPTABLES-HOWTO ↗ <http://netfilter.samba.org/>



**Mátó Péter** (atya@andrews.hu), informatikus mérnök és tanár. Biztonsági rendszerek ellenőrzésével és telepítésével, valamint oktatással foglalkozik. 1995-ben találkozott először linuxos rendszerrel. Ha teheti, kirándul vagy olvas.



**Borbély Zoltán** (bozo@andrews.hu), okleveles mérnök-informatikus. Főként Linuxon futó számítógépes biztonsági rendszerek tervezésével és fejlesztésével foglalkozik. A 1.0.9-es rendszermag ideje óta linuxozik. Szabadidejét barátaival tölti.

*A főszerkesztő ezúton kér elnézést a Tisztelt Olvasótól és a Szerzőktől, ha úgy érzik, hogy a szerzők „technicus terminusainak” magyarázata csorbitotta a szövegek érthetőségét.*

## DMZ hálózatok tervezése és használata

Bemutatjuk, hogyan gondoskodhatunk azokról a szolgáltatásainkról, amelyek sérülékeny hálózatokkal is kapcsolatba kerülhetnek.

**A** DMZ-típusú (DeMilitarized Zone – szabad terület) hálózati felépítés manapság a tűzfalak egyik leghasznosabb kiegészítő eszköze. Olyan hálózatrészről van szó, amelyben elhelyezhetjük az összes nyilvánosan hozzáférhető szolgáltatásunkat. Egyrészt így könnyebben szemmel tarthatjuk ezen szolgáltatások működését, másrészt pedig elszigetelhetjük azokat a belső hálózattól. A szabad területek, a bástyakiszolgálók és a Linux együtt kifejezetten jó csapatot alkotnak.

Mit is jelent a szabad terület? Megtervezhető-e többféleképpen is? Vajon minden internet-szolgáltatónak szüksége van szabad területekre? Ezekkel a kérdésekkel régebben nem foglalkoztunk, éppen ezért most átfogóbban vizsgáljuk a szabad területek biztonsági kérdéseit. Ha esetleg úgy gondoljuk, hogy a jelenleg használt tűzfalrendszer, szabad terület nélkül tökéletesen megfelel az igényeinknek, akkor is célszerű végigolvasni ezt a cikket. Minden olyan gép és szolgáltatás – akár szabad területen helyezkedik el, akár nem –, amely nem megbízható hálózatokkal is kapcsolatba kerülhet, különleges odafigyelést igényel, és a cikkben bemutatott ötletek között számos olyan is van, amelyik DMZ-, és szokványos környezetben egyaránt használható.

### A szóhasználatról

Mielőtt továbblépnünk tisztázzunk néhány kifejezést. Elképzelhető, hogy a felsorolt meghatározások némelyikét korábban másképp használtuk, éppen ezért szeretném már az elején leszögezni, hogy ebben a cikkben milyen értelemben hivatkozom rájuk:

- **Szabad Terület (DMZ):** nyilvánosan is elérhető kiszolgálókat tartalmazó hálózatrész, amely megfelelő módon elszigetelt a „belső” hálózattól.
- **Belső hálózat:** az a hálózatrész, amelyet szeretnénk megvédeni – a végfelhasználók rendszerei, a bizalmas adatokat tartalmazó kiszolgálók és az összes többi olyan rendszer, amelyet szeretnénk elzárni a kívülről érkező kapcsolatok elől. Védett területnek is nevezhetjük.
- **Tűzfal:** olyan rendszer vagy eszköz, amely az egyik hálózatot elszigeteli a másiktól. Ez lehet egy útválasztó, azaz olyan számítógép, amely hálózati forgalmat irányító programot futtat, esetleg egy ilyen rendeltetésű egyedi eszköz, vagy bármilyen más rendszer, amely csomagszűrésre, proxyszolgáltatások biztosítására és egyéb, hozzáférés-vezérléshez tartozó feladatok elvégzésére alkalmas. A cikk során egyetlen, többkártyás számítógépre gondolunk.
- **Többkártyás számítógép:** egy olyan számítógép, amely egynél több hálózati csatlakozóval rendelkezik.
- **Bástyagép:** olyan rendszer, mely nyilvánosan hozzáférhető szolgáltatásokat nyújt, de ön maga nem tűzfal. Általában a bástyagépeket a szabad területen helyezük el (bár máshol is elhelyezhetjük). A bástya kifejezés arra vonatkozik, hogy az operációs rendszer valamilyen módon megerősített, ez a feltétel azonban nem mindig teljesül.
- **Csomagszűrés:** olyan eljárás, mely az IP-csomagok fejlécében tárolt adatok (forrás IP-cím, cél IP-cím, forráskapu és célkapu) megvizsgálása után továbbengedi, vagy pedig eldobja az adott csomagot. Ez az eljárás a csomagok tartalmával nem foglalkozik, azaz a hibás felépítésű csomagokat nem feltétlenül veszi észre

(feltéve, hogy a csomag fejléce megfelelő adatokat tartalmaz). Szinte minden tűzfal végez csomagszűrést, ez azonban önmagában még nem ad elegendő védelmet minden támadással szemben. A legtöbb útválasztó (és sok gyenge tűzfal) kizárólag csomagszűréssel védelmezi a rábízott hálózatot.

(Lásd e témával kapcsolatban a 34. oldalon található Könnyű álom című cikket.)

- **Proxy:** olyan szolgáltatás, mely közvetítő szerepet tölt be a belső és a külső gépek között. Proxy használata közben a felhasználó nem közvetlenül a kiszolgálóval tartja a kapcsolatot, a proxy „közvetít”. Az eljárás lehetőséget ad kifinomultabb szűrések használatára, mivel itt az alkalmazási réteg adatait elemezhetjük (ez a módszer hatékonyabb az egyszerű csomagszűrésnél). Vannak olyan tűzfalak, amelyek kifejezetten proxykra épülnek.
- **Állapotvizsgálat:** legegyszerűbb formájában arra a háromlépéses kézfogásnak a megfigyelésére vonatkozik, amely egy adott TCP-kapcsolat felépülésekor zajlik a gépek között (gép1: SYN, gép2: SYN+ACK, gép1: ACK). Kifinomultabb változata magában foglalja ennek és az ezeket követő összes többi állapot adatainak nyomon követését. Az utóbbi változat sokkal kevésbé elterjedt, mint az első.

Ez a rengeteg szakkifejezés elsőre talán egy kicsit nagy falatnak tűnhet, viszont nagyon hasznos az ismeretük. Most már készen állunk arra, hogy beássuk magunkat a Szabad Területek felépítésébe.

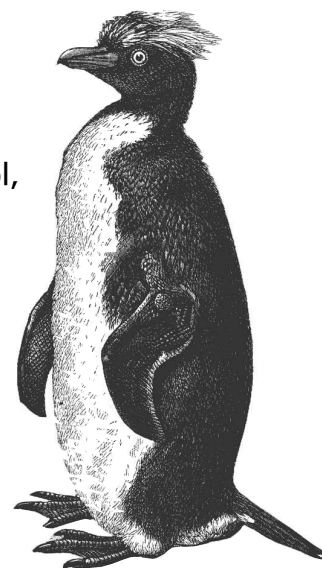
### A tűzfaltípusok és a DMZ hálózatok lehetséges szerkezete

A drága kereskedelmi tűzfalak világában a tűzfal kifejezés szinte mindig egyetlen olyan számítógépet vagy különleges eszközt jelöl, amely több hálózati csatlakozással bír. Ez a meghatározás a gyengébb megoldásokra is ráhúzható: a hálózati kártyák manapság már igen olcsók, akár csak a PC-k.

Már elmúltak azok a napok, amikor egyetlen számítógép önmagában nem volt képes arra, hogy egy nagyobb hálózatban nyomon kövesse az összes bejövő és kimenő csomagot. Akkoriban még kizárólag az útválasztók kaphattak helyet a hálózatok elleni támadások első védelmi vonalában (az egyszerű számítógépek nem).

Mára a helyzet teljesen megváltozott. Manapság már az erős internet-kapcsolatokkal rendelkező szervezetek is többkártyás tűzfalakkal igyekeznek megvédeni hálózataikat. Jelenleg már a viszonylag lassú PC-k is képesek arra, hogy kifinomult ellenőrzést végezzenek akár T1-es (1,544 Mb/mp) hálózati kapcsolatokon is.

Az 1. ábrán azt a tűzfalszerkezetet láthatjuk, amellyel manapság a leggyakrabban találkozhatunk. Ahogy azt az ábrán is láthatjuk, a biztonság első, de nem kizárólagos védelmi vonalában egy csomagszűrő útválasztó kap helyet. Közvetlenül az útválasztó mögött található magát a tűzfalat (ez lehet Sun SpareStation is), melyen RedHat fut. A belső hálózatnak sem az Internettel, sem



pedig a külső útválasztóval nincs közvetlen kapcsolata: minden hálózati forgalomnak át kell haladnia a tűzfalon.

Véleményem szerint minden külső útválasztónak tartalmaznia kell valamilyen csomagszűrést. Még ha az útválasztóhoz drága, és lehetőség szerint jól beállított és karbantartott tűzfal csatlakozik, akkor sem árt, ha bizonyos veszélyekkel szemben többszörösen is biztosítjuk magunkat.

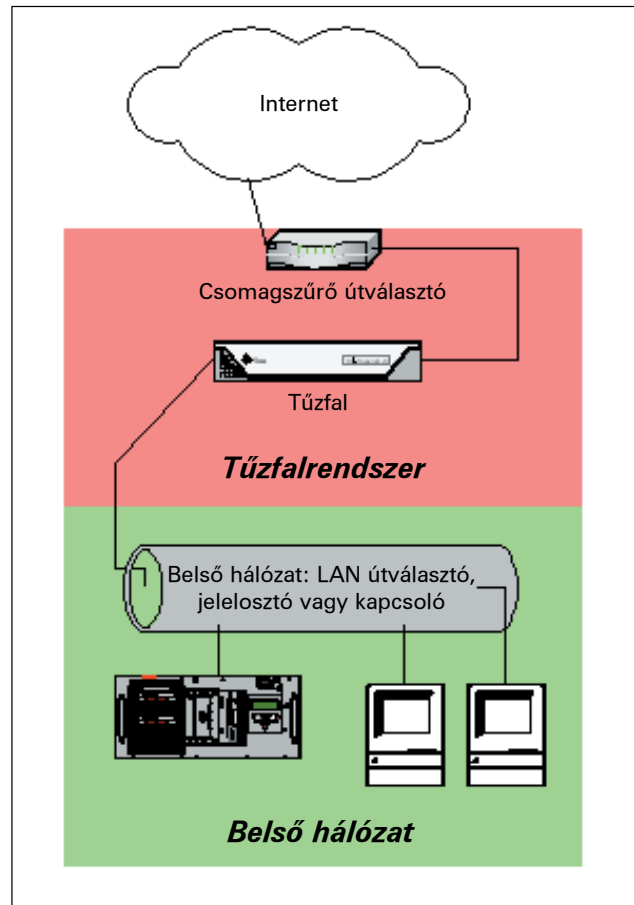
Mi a baj az első ábrával? Mi hiányzik róla? Mindössze annyit állítottam, hogy ez a szerkezet igen elterjedt, azt viszont nem mondtam, hogy tökéletes. Az olyan nyilvános szolgáltatásoknak, mint az SMTP (elektronikus levelezés), a DNS vagy a HTTP szintén át kell menniük a tűzfalon, vagy pedig magán a tűzfalon kell elhelyezni ezeket.

Az a tény, hogy az ilyen jellegű hálózati forgalmat is a tűzfalon keresztül bonyolítjuk, önmagában még nem növeli meg annak az esélyét, hogy a belső hálózat számítógépeit támadás érje, jelentősen súlyosbítja azonban az ilyen támadás következményeit. A nyilvános szolgáltatásoknak a tűzfalon történő elhelyezése nem feltétlenül rossz ötlet, mert hol lehetnének nagyobb biztonságban ezek a szolgáltatások, mint magán a tűzfalon. Ebben az esetben azonban nyilvánvaló, hogy a megfelelő teljesítmény eléréséhez a tűzfalnak minden létező erőforrását a csomagok ellenőrzésére és továbbítására kell fordítania. (Ez alól van néhány kivétel, amiről rövidesen részletesebben is beszélünk majd.) Emellett ilyenkor kényesebbé válik a tűzfal karbantartása is, hiszen érzékeny programok is futnak rajta.

Hol tudjuk tehát elhelyezni nyilvános szolgáltatásainkat úgy, hogy azok sem közvetlenül, sem pedig közvetve ne jelentsenek veszélyt a belső hálózatra, és ne terheljék túl a tűzfalat sem? Pont erre való a szabad terület. Legegyszerűbb formájában Szabad Területnek tekinthetünk minden olyan hálózatot, amely kívülről elérhető, a belső hálózattól azonban el van szigetelve. Esményi esetben azonban a szabad terület is egy tűzfal védelmét élvezi. A 2. ábrán ezt az eszményi szerkezetet láthatjuk. Ezen az ábrán a tűzfal szerepét egy háromkártyás számítógép tölti be. A nyilvános szolgáltatásokat biztosító gépek egy saját hálózatban helyezkednek el, amely a tűzfalhoz kapcsolódik. A hálózat többi része is ezt a tűzfalat használja, azonban másik hálózati felületen keresztül csatlakozik ahhoz. Ha megfelelően van beállítva, akkor a tűzfal más és más szabályokat alkalmaz bármelyik hálózati részből bármelyik rész felé áramló forgalom kiértékelésére. Úgy tűnik, hogy ez a megoldás sokkal nagyobb felügyeleti költséggel jár, mintha a nyilvános szolgáltatásokat a belső hálózatban, vagy pedig magán a tűzfalon helyeznénk el, valójában azonban sokkal egyszerűbb, mivel a DMZ-t egyetlen egységként kezelhetjük. Ilyen jellegű hálózatot természetesen sokféleképpen összeállíthatunk, a 3. ábrán rögtön két példát is láthatunk. A védett alhálózat (Screened Subnet) biztonsága teljes egészében a külső és a belső útválasztók biztonsági rendszerétől függ. A hálózat belseje közvetlen kapcsolatban áll a külvilággal, és ezen az úton a csomagok szabad áramlását kizárólag egy útválasztó csomagszűrő szabályai gátolhatják meg.

A 3. ábra jobb oldalán látható kiépítésnek a „Vergődés a szél-ben” nevet adtam. Ebben az esetben a belső hálózatot teljes értékű tűzfal választja el az Internettől, a szabad terület azonban a tűzfalon kívül helyezkedik el, valamint védelmét kizárólag a csomagszűrő útválasztó biztosítja.

Mindkét megoldással találkozhatunk a tűzfalakkal foglalkozó könyvekben (elképzelhető, hogy ott más néven szerepelnek), véleményem szerint azonban ezek a rendszerek túlságosan nagy bizalmat fektetnek az útválasztóba. Ez a túlzott bizalom több okból is veszélyes: egyrészt elképzelhető, hogy a tűzfal és az útválasztó nem ugyanannak a rendszerfelügyelőnek a hatáskörébe tartozik. Lehet, hogy az útválasztóért felelős személy nem használ kellőképpen erős jelszavakat, nem fektet elegendő hangsúlyt a hozzáférés-szabályozási listákra. Sőt, még modemet is elhelyezhet az útválasztóban mondván, így a gyártó könnyebben karbantarthatja. Ezenkívül az útválasztóra



1. ábra Többkártyás tűzfal

általában sokkal könnyebb behatolni, mint egy jól beállított számítógépre (az útválasztókat például szinte minden esetben el lehet érni telnet protokoll segítségével, ami egyáltalán nem nevezhető biztonságosnak). Továbbá a csomagszűrés valójában nem alkalmas arra, hogy segítségével kifinomult módon szabályozhassuk a hálózati forgalmat. Még egy nyílt forráskódú, ingyenes tűzfalprogram is támogathatja az IPSEC, az alkalmazásszintű proxyk, az állapotvizsgálat, a Radius hitelesítés és még sok más olyan szolgáltatás használatát, amelyekkel az útválasztókon nem találkozhatunk. Egyszerűen összefoglalva: az útválasztókat arra tervezték, hogy irányítsák a forgalmat, nem pedig arra, hogy megvédjék a hálózatot.

Mi a helyzet a Cisco Pixszel? A Pix tűzfal is útválasztó ugyan, amit azonban a Cisco IOS operációs rendszer megerősített, biztonságközpontú változatával szereltek fel. Habár erősen támaszkodik az egyszerű csomagszűrésre, rengeteg olyan kiegészítő szolgáltatással is ellátták, amelyek alkalmassá teszik arra, hogy jól beállítva nagyszerű tűzfal lehessen. Amikor megkérdőjelezem az útválasztók tűzfalként történő felhasználását, akkor az általános célú útválasztókra gondolok. Összefoglalva, a szabad terület szerkezete elsősorban a tűzfal(ak) felépítésétől függ. Egy többkártyás tűzfal köré olyan hálózatot építhetünk, amelyben a DMZ hálózatrész az Internettől és a belső hálózattól egyaránt teljes mértékben el van választva (lásd 2. ábra).

### Mi legyen a Szabad Területen?

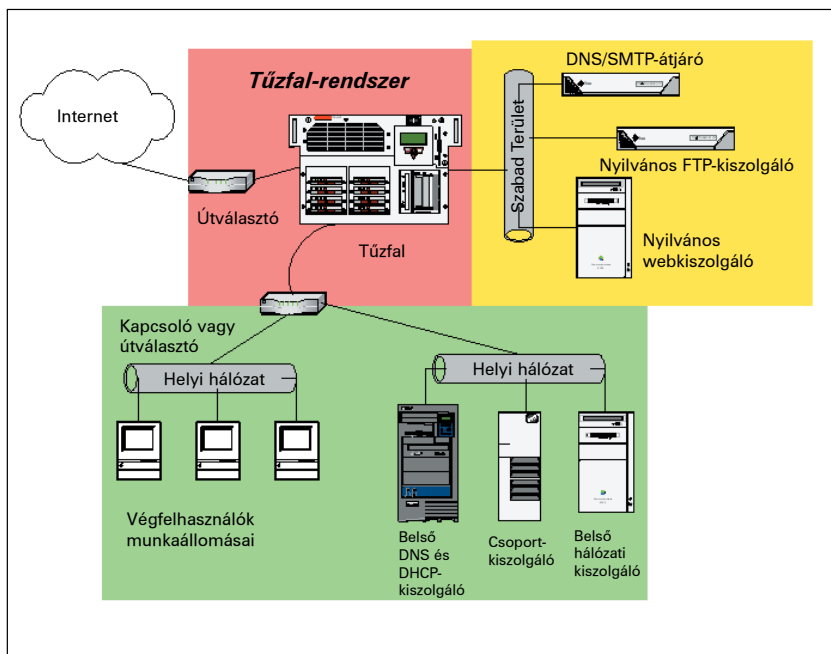
Ha már eldöntöttük, hogy hol helyezzük el a DMZ hálózatot, pontosan meg kell határozni azt is, hogy mit fogunk elhelyezni benne. Célserű minden nyilvánosan elérhető szolgáltatást ide helyezni. Túlságosan sokszor találkozom olyan hálózatokkal, amelyekben egy

vagy több – biztonsági szempontból nézve kényes – szolgáltatást a tűzfalon keresztül juttatnak el belső géphez annak ellenére, hogy a rendszerben helyet kap egyébként nagyszűrően működő DMZ hálózatrész is. Általában olyan programokról van szó, mint amilyen például az MS-Exchange, amit nem láttak el az internetes alkalmazásoknál megkövetelhető biztonsági alrendszerrel.

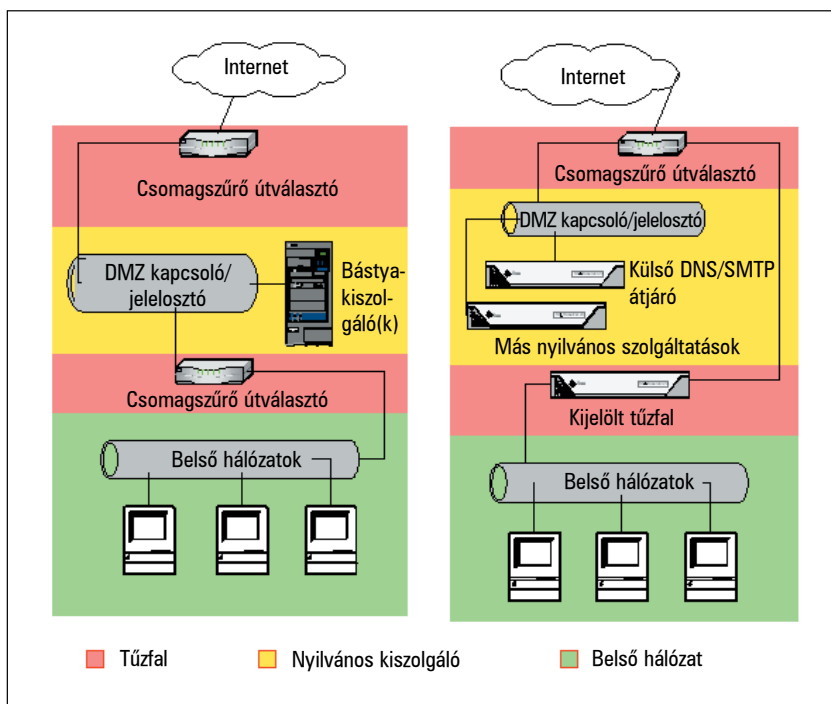
Egyetlen ilyen alkalmazás elég ahhoz, hogy rést nyisson az egyébként biztonságos rendszeren: mindössze annyi kell, hogy valamelyik alkalmazásunkat egy egyszerű átmeneti tártúlsordulással meg lehessen támadni, és a hivatlan látogató máris hozzáférhet az összes olyan géphez, amely az adott gépről elérhető. Egy ilyen esetben nem mindegy, hogy a megtámadott gépről csak a szabad terület gépeihez lehet hozzáférni, vagy pedig a belső hálózat összes gépéhez. Ezt a szempontot nem lehet eléggé kihangsúlyozni: a szabad terület legnagyobb értéke az, hogy szétválaszthatjuk a külvilág és a belső hálózat számára szükséges szolgáltatásokat.

Elképzelhető, hogy az a személy, aki a tűzfalon keresztüláramló szolgáltatásért felelős, más, mint a tűzfalat és a szabad terület kiszolgálóit kezelő szakember. Meglehet, hogy az előbbi nem ügyel annyira a biztonságra, mint amennyire kellene. Ez már önmagában is elegendő ok arra, hogy a nyilvános kiszolgálókat a Szabad Területen helyezzük el, mivel így ezek a gépek a tűzfal rendszergazdájának felügyelete alá kerülnek, aki ideális esetben rendkívül kényes a gépek biztonságára. Ez azt jelenti, hogy a vállalati levelezést, a névszolgáltatást és a többi fontos szolgáltatást szintén a DMZ hálózatban kell elhelyezni? Egyáltalán nem! A megoldás ebben az esetben az, hogy a szolgáltatást szétválasszjuk külső és belső részre (lásd 2. ábra). A DNS-t például fel kell bontanunk külső és belső DNS-re. A külső DNS adatbázisa, amely az Internetről is hozzáférhető gépekről tartalmazhat adatokat. A nem nyilvános gépek adatait pedig a másik, belső DNS adatbázisba kell elhelyeznünk, amely a külső gépek számára tökéletesen láthatatlan. Ugyanez a helyzet az elektronikus levelezéssel is. A belső levelezést (a belső gépekről belső gépekre elküldött leveleket) szigorúan belső gépeknek kell kezelniük, az Internetre kimenő vagy onnan beérkező leveleket pedig egy szabad területen elhelyezett kiszolgálónak (ezt a gépet SMTP-átjárónak is szokás nevezni).

Szinte minden olyan szolgáltatás szétbontható két részre, amely saját és nyilvános módon egyaránt használható. Míg ez látszólag sok többletmunkával jár, valójában egyáltalán nem jelent külön megterhelést, sőt szabadságot ad, mivel lehetővé teszi, hogy a belső szolgáltatások kialakításakor a kényelemi szempontokra, a külső szolgáltatások esetében pedig a teljesítményre és a biztonságra összpontosítsunk. Fontos szempont az is, hogy ily módon Linux, OpenBSD és más nyílt forráskódú programokat is beépíthetünk az egyébként kereskedelmi programokra épülő környezetünkbe. Mondani sem kell, hogy a nyilvános használatra szánt szolgáltatásoknak kizárólag a szabad területen kell



2. ábra Többkártyás tűzfal szabad területtel



3. ábra A védett hálózat és a „Vergődés a szélben” változatok

megjelenniük. Összegezve az eddigieket: az összes nyilvános szolgáltatást, beleértve a kívül és belül egyaránt használt szolgáltatások nyilvános részét is (feltéve, hogy az adott szolgáltatás felbontható két részre), kivétel nélkül a DMZ hálózatrészben kell elhelyezni.

### Az erőforrások megosztása a DMZ hálózatban

Minden nyilvános szolgáltatás a szabad területre kerül. De külön gépre lesz szükség minden egyes szolgáltatáshoz? Esetleg bizonyos szolgáltatásokat elhelyezhetünk magán a tűzfalon is? Használjunk-e jelelőztőt vagy hálózati kapcsolót a szabad területen?

Az utolsó kérdés a legkönnyebb: mivel a kapcsolt kapuk ára évről évre csökken, a kapcsolók használata minden hálózatban, így a DMZ hálózatrészekben is ajánlott. A kapcsolók két tekintetben is nagyon hasznosak: biztonsági szempontból vizsgálva használatuk azért előnyös, mert lehetetlen lehallgatni azt a forgalmat, amely el sem jut a kapcsoló adott kapujához. Mivel a szabad területen lévő gépeket nagyobb valószínűséggel éri majd támadás, mint a belső hálózat gépeit, ez a nézőpont igen fontos. Nemcsak azt kell végiggondolnunk, hogyan védhetjük meg ezeket a gépeket a támadástól, hanem tisztán kell látnunk az esetleg sikeres támadás következményeit is. A kapcsoló természetesen a teljesítmény tekintetében is jobb választás, mint a jeleselőző. Ne feledkezzünk meg azonban arról, hogy a kapcsolók teljesítménye korlátozott. Ha például a kapcsolónk másodpercenként legfeljebb 800 megabit továbbításra képes, akkor hiába van rajta tíz, egyenként 100 Mb/mp sebességű kapu, akkor sem fog tudni másodpercenként 1000 megabitet feldolgozni. Mindezek ellenére leszögezhetjük, hogy még az alacsonyabb osztályba tartozó kapcsolók is bőven túlszámalyják a velük egy szinten lévő jeleselőzők teljesítményét. Ettől még elképzelhető, hogy szabad területünk kiszolgálásához egy jeleselőző is elegendő. Ez elsősorban attól függ, hogy hány gép van a hálózatrészben, ezek mekkora forgalmat bonyolítanak, és mennyire aggódunk amiatt, hogy az egyik gépen keresztül esetleg a hálózat többi gépét is feltörhetik.

A másik két kérdést általában úgy is megválaszolhatjuk, hogy a biztonsági szempontok helyett más tényezőket helyezünk előtérbe (például költség, várható terhelés, hatékonyság stb.), feltéve természetesen, hogy a szabad területen lévő gépek már megfelelően biztonságosak. Figyeljünk arra is, hogy a Szabad Területről kimenő, illetve az oda beérkező hálózati forgalmat szabályozó tűzfal is a lehető legszigorúbb módon legyen beállítva.

### A DMZ gépek biztosításának irányelvei

Nyilvánvalónak tűnik, hogy a szabad terület gépeit is védenünk kell. Ennek ellenére sokszor találkozom olyan szervezetekkel, amelyek kellőképpen igényesek ahhoz, hogy szabad területet tartsanak fenn, annyira azonban nem, hogy gondoskodjanak ennek a hálózatnak a biztonságáról. A jó hír az, hogy egy kis időráfordítással jelentős mértékben csökkenthetjük rendszerünk sebezhetőségét.

Az operációs rendszerből, a programokból és a rendszermagból is mindig a legfrissebb megbízható változatot használjuk, és azonnal telepítjük a különböző biztonsági hézagokat befoltozó javítócsomagokat, amint azok megjelennek!

Ha mindenki megfogadná ezt az egyszerű és magától értetődő tanácsot, akkor a [www.hackernews.com](http://www.hackernews.com) lapon megtalálható feltört weblapok listája valószínűleg sokkal rövidebb lenne. Általában elmondható, hogy a hálózatokba történő betörést a legtöbb esetben az teszi lehetővé, hogy az adott hálózatban valamely programnak nem a legfrissebb változata fut. Mindannyian tisztában vagyunk ezzel, azonban nem mindig vagyunk hajlandók időt szánni a gondok orvoslására.

Kapcsoljuk ki az összes olyan szolgáltatást és démont, amelyekre nincs szükségünk. A használaton kívüli programok karbantartása általában nem megfelelő, így nyilvánvaló támadási felületet biztosítunk a betörők számára. A gond nagyon könnyen megoldható. A rendszer telepítésekor egyszerűen töröljük le, vagy pedig nevezzük át az összes felesleges hivatkozást a `/etc/rc.d/` megfelelő könyvtárban.

Ha például webkiszolgálót szeretnénk telepíteni, de nincs szükség arra, hogy a gép DNS-szolgáltatást is nyújtson, akkor kiadhatjuk a következő parancsot:

```
mv /etc/rc.d/rc2.d/S30named
/etc/rc.d/rc2.d/disabled_S30named
```



[http://www.atstake.com/security\\_news/](http://www.atstake.com/security_news/)

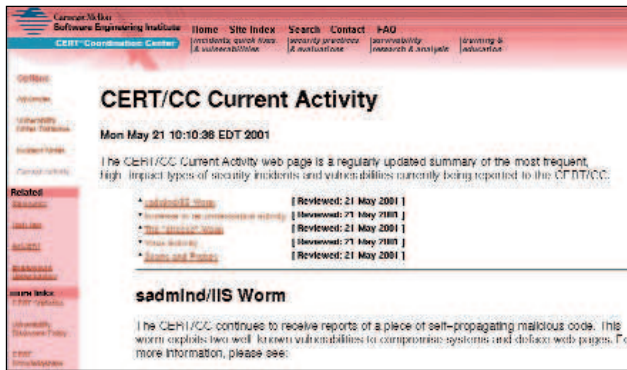
(A szolgáltatások a különböző Linux-változatok alatt más és más helyen engedélyezhetők, illetve tilthatók le.)

Amikor a felesleges szolgáltatások letiltásán dolgozunk, különösen figyeljünk oda az alábbi szolgáltatásokra:

- **RPC:** a Sun távoli eljárásvezérlés (Remote Procedure Control – RPC) protokollja – ez manapság szinte minden Unix-változatban megtalálható – lehetővé teszi, hogy az rsh, rcp, rlogin, nfs stb. programok segítségével utasításokat hajtsunk végre távoli rendszerben. Ez a protokoll sajnos nem biztonságos, főleg nem a szabad területeken. Ne nyújtunk tehát ilyen szolgáltatásokat a külvilág számára. Ha szükségünk van ezekre a szolgáltatásokra, akkor használjuk az ssh-t, ezt ugyanis kifejezetten az rpc szolgáltatások felváltására tervezték. Töröljük (vagy nevezzük át) az nfsd és az nfsclientd állományokat a `/etc/rc.d` könyvtár alatt fel-lelhető összes alkönyvtárban, és távolítsuk el az `r*` parancsokra vonatkozó sorokat a `/etc/inetd.conf` fájlból is.
- **inetd:** az Internet démon (inetd) kiválóan alkalmas arra, hogy figyelje a rábízott kapukat, és szükség esetén elindítsa a kapukhoz tartozó szolgáltatásokat. Ez a hasznos kis szolgáltatás azonban még az igen részletes naplózási lehetőség ellenére sincs olyan biztonságos, mintha az egyes szolgáltatásokat démonként futtatnánk. Egy FTP-kiszolgáló esetében például semmi okunk nem lehet arra, hogy ne futtassuk állandóan az FTP-folyamatokat. Sőt, a legtöbb olyan szolgáltatás, amely az inetd.conf fájlban alapértelmezés szerint engedélyezve van, teljesen felesleges vagy pedig nem biztonságos (esetleg mindkettő). Ha mindenképpen szükségünk van az inetd használatára, akkor a `/etc/inetd.conf` fájl átszerkesztésével tiltsuk le az összes olyan szolgáltatást, amelyre nincs szükségünk (és azokat is, amelyekről még soha sem hallottunk). Sok olyan rpc szolgáltatás van, amely az inetd.conf-ból indul.
- **linuxconf:** annak ellenére, hogy a linuxconf (olyan eszköz, melynek segítségével a rendszergazdák távolról is hozzáférhetnek a géphez) jelenlegi változata egyetlen olyan ismert hibát sem tartalmaz, amit esetleges támadás során fel lehetne használni, a CERT jelentése szerint a szolgáltatás pártizálásával felfedhetők a rendszer egyéb gyengeségei  
 ➔ [http://www.cert.org/current/current\\_activity.html](http://www.cert.org/current/current_activity.html)
- **sendmail:** sokan azt gondolják a sendmail szolgáltatásról – amely alapértelmezés szerint engedélyezve van a legtöbb Unix-változatban –, hogy azokon a gépeken is szükséges, amelyek kizárólag saját maguknak küldenek leveleket (felügyeleit célból). Ez azonban nem igaz, a sendmail (vagy postfix, qmail stb.) szolgáltatásra csak azokon a gépeken van szükség, amelyek más gépekkel is váltanak leveleket. A sendmail szolgáltatás általában a `/etc/rc.d/rc2.d` vagy a `/etc/rc.d/rc3.d` alatt kerül elindításra.
- **telnet, FTP és POP:** Ennek a három protokollnak megvan az a közös vonása, hogy mindegyikük egyszerű szöveg formájában

© Kiskapu Kft. Minden jog fenntartva





➔ [http://www.cert.org/current/current\\_activity.html](http://www.cert.org/current/current_activity.html)

továbbítja a felhasználók nevét és jelszavát a hálózaton keresztül. A telnet és az FTP helyett használhatjuk az ssh-t, illetve ennek a fájlátvitelre alkalmas scp nevű segédprogramját. Az elektronikus leveleket önműködően továbbíthatjuk egy másik géphez, meghagyhatjuk a szabad területtel bíró gépen úgy, hogy csak ssh-n keresztül lehessen hozzáférni, vagy a POP felhasználásával továbbíthatjuk az ssh-hoz. Mindhárom szolgáltatást az inetd indítja el.

Használjunk chroot-ot, ahol csak lehet!

Egyes démonokat (ilyen például named) a chroot börtönében is futtathatunk (ebben az esetben a démon a chroot mellett megadott könyvtárat tekinti gyökérkönyvtárnak, és nem is tud kilépni abból). Ez nagyon értékes biztonsági szolgáltatás, ha ugyanis a chroot-tal futtatott szolgáltatást sikerül is feltörni valahogy, a betörő akkor sem fog hozzáférni a szolgáltatás gyökérkönyvtárán kívül eső fájlokhoz. Linux alatt bármilyen parancsot futtathatunk ezzel a módszerrel: egyszerűen adjuk ki a `chroot elérési_út` utasítást. Ha például a bubba `-v plop` parancsot úgy szeretnénk futtatni, hogy az ne tudjon kilépni a `/var/bubba` könyvtárból, akkor gépeljük be a következőt:

```
chroot /var/bubba /usr/local/bin/bubba -v plop
```

Ebben az esetben azokat a rendszerfájlokat, amelyekre a programnak szüksége van, be kell másolnunk egy olyan könyvtárba, amelyhez a program hozzáférhet. Ha például a fenti parancs olvasni szeretne a `/etc/passwd` fájlból, akkor ezt a fájlt át kell másolnunk a `/var/bubba/etc` könyvtárba. A másolatnak természetesen elég azokat az adatokat tartalmaznia, amelyekre valóban szükségünk van. Ha például a bubba parancsot csak az „anonymus” nevű felhasználó fogja futtatni, akkor a `/var/bubba/etc/passwd` fájlban csak egyetlen sort kell tartalmaznia (például `nobody::50:50:Anonymous user::/bin/noshell`).

Futtassuk a szolgáltatásokat a lehető legalacsonyabb jogosultsági szinten!

Vannak ugyan olyan démonok, amelyeket csak rendszergazdaként futtathatunk, manapság azonban egyre több olyan program létezik, amelyeket alacsonyabb jogosultsággal rendelkező felhasználók is futtathatnak. Például a Postfix (mely a sendmail szolgáltatást hivatott felváltani) általában egy postfix nevű, alacsony jogosultsági szinttel rendelkező felhasználóként fut.

A módszer előnye hasonló, mint a chroot esetében: ha az esetleges betörő ilyen szolgáltatáson keresztül jut be a rendszerbe, akkor alacsonyabb jogosultságokkal rendelkezik majd (remélhetőleg jóval alacsonyabbakkal), mint a rendszergazda.

Töröljük vagy tiltsuk le a felesleges azonosítókat!

Egyes Linux-változatok alapértelmezés szerint meglehetősen hosszú `/etc/passwd` fájlokat tartalmaznak olyan programcsomagok kedvéért, amelyeket többnyire még csak nem is telepítettünk. A hordozható

számítógépemre telepített SuSE Linux `/etc/passwd` fájljában például 22 felesleges bejegyzést találtam. Távolítsunk el minden olyan bejegyzést, amelyekre nincs szükségünk.

A naplózás beállítása és a naplók ellenőrzése:

Szintén olyan dolog, amiről tudjuk, hogy mennyire fontos, sokszor mégis megfeledekezünk róla. A nem létező naplókat nem is lehet átnézni, azokból a naplók közül pedig, amelyeket nem olvasunk át, semmit sem fogunk tanulni. Gondoskodjunk róla, hogy a fontos szolgáltatások működését megfelelő módon naplózzuk. Tudnunk kell, hogy hol tárolódnak a létrehozott naplófájlok és azt is, hogy mi történik velük, amikor túlságosan nagyra nőnek. Nézzük át rendszeresen az éppen használatos naplófájlokat!

Ez utóbbi feladatban a `grep` parancs nagy segítséget jelenthet, a `cat` pedig önmagában általában túlságosan sok adatot zúdít ránk. A naplók figyelését különböző parancsfájlok segítségével önműködően is elvégezhetjük. A parancsfájlok arra is alkalmasak, hogy figyeljük a fontosabb rendszerbeállításokat tartalmazó fájlok változásait. Ha több DMZ gép működését szeretnénk szemmel tartani, akkor nem kell egyesével végignézniük az egyes gépeken létrehozott naplófájlokat, mivel a `syslogd` segítségével egyetlen rendszerbe is összegyűjthetjük azokat. A `syslogd` démon a helyi folyamatok naplózása mellett képes távoli gépekről érkező naplóbejegyzések fogadására is. Ha például DMZ hálózatunk két gépből áll (Bobo és Rollo), és a naplófájlokat egyetlen helyen szeretnénk összegyűjteni, akkor módosítsuk a Bobo gépen a `/etc/syslogd.conf` fájlt úgy, hogy kizárólag a következő sort tartalmazza:

```
*.* @rollo
```

Ennek hatására a Bobo gépen futó `syslogd` minden naplóbejegyzést a Rollohoz továbbít majd.

Annak ellenére, hogy az imént bemutatott szolgáltatás nagyon hasznos, megvannak a maga kis biztonsági gondjai. Ha valakinek sikerül betörnie a Rollo rendszerbe, akkor hozzájut a Bobo naplófájljaihoz is, és így akár olyan adatok is megszerezhet, amelyek felhasználásával azután a másik rendszert is könnyedén feltörheti.

Használjuk a tűzfal biztonsági házirendjét és az IP-hamisításokat megelőző szolgáltatásait!

Teljesen természetes, hogy szeretnénk odafigyelni arra, hogy milyen adatok kerülnek be kívülről a szabad területre. Ugyanolyan fontos azonban az is, hogy megfelelően korlátozzuk a szabad területről a belső, illetve a külső hálózatba áramló adatforgalmat is. Az előzőre azért van szükség, hogy a szabad terület megtámadása esetén meg tudjuk védeni a belső hálózatot, az utóbbi pedig azért fontos, hogy elejét vehessük annak, hogy valaki egyetlen, a szabad területen található feltört gépről támadjon meg más hálózatokat. Természetes, hogy nem szeretnénk, ha bármi bejutna az Internetről a belső hálózatba. Bármi történjen is, a tűzfal biztonsági rendszere hatékonyabban működik majd, ha a tűzfal különbséget tud tenni a megbízható és a kétes eredetű IP-címek között. Ha erre nem képes a rendszerünk, akkor előfordulhat, hogy külső felhasználó az IP-cím meghamisításával csomagokat csempésszen be a rendszerünkbe. A legtöbb tűzfalon nincs alapértelmezés szerint engedélyezve ez a szolgáltatás. Még ha támogatja is a tűzfal ennek használatát, valószínűleg külön engedélyeznünk kell azt.

Igazán megéri az erőfeszítést.



*Mick Bauer* (mick@visi.com) hálózati biztonsággal foglalkozó szaktanácsadó. 1995 óta a Linux elkötelezett híve, 1997 óta pedig OpenBSD profétaként tevékenykedik. Mick szívesen fogad minden kérdést, és megjegyzi.

## Ellenőrzés pásztázókkal (1. rész)

Az nmap használatával felderítheted rendszered gyenge pontjait.

**T**alán hallottál már rémtörténeteket, hogy a rossz szándékú betörők milyen egyszerűen felderíthetik az áldozat rendszerének sebezhető pontjait az Internetről készen letölthető eszközök segítségével. Ezek a történetek általában sajnos igazak. Szerencsére sok ilyen program rendkívül jól használható törvényes célokra (néhányukat ilyen szándékkal tervezték), azaz felderítheted velük a rendszered fogyatékosságait.

Ez a cikk és a következő hónapban megjelenő folytatása bemutatja, hogyan használhatja az átlagos rendszergazda és a képzett biztonsági szakember az nmapot és a nessust, ezt a két kiemelkedő szabad forrású programot a rendszer biztonságának fejlesztésére. De ne feledjük, a tudás hatalom, rajtunk áll, hogy felelősséggel használjuk (és olyan módon, hogy ne kényszerítsük arra az egyenruhásokat, hogy elkobozzák kedvenc linuxos gépeinket)!

### Miért pásztázunk mi, a jó fiúk?

Miért pásztázunk? A betörő azért pásztáz, hogy megállapítsa, milyen szolgáltatások futnak a célba vett rendszeren, és milyen jól ismert támadási felületei vannak azoknak. A rendszergazda tulajdonképpen ugyanezért pásztáz, de az ő célja a rendszer hibáinak kijavítása (vagy legalább megértése), és természetesen nem akar betörni saját gépére.

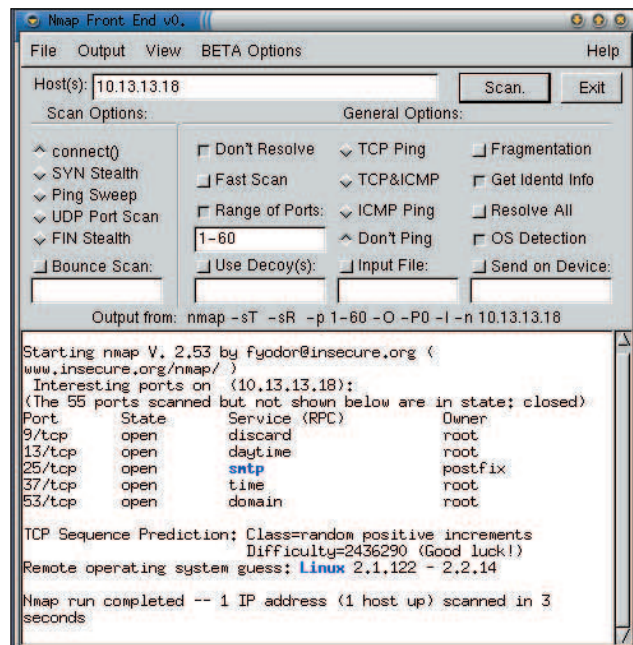
A jó fiúk furcsállhatják, hogy ugyanazokat az eszközöket kell használniuk, mint a rossz fiúknak, akiket meg akarnak fékezni. Végül is nem úgy szoktuk az ajtózárukat ellenőrizni, hogy a saját ajtókat rugdossuk. A rendszerbiztonság azonban sokkal bonyolultabb, mint a fizikai biztonság. Közel sem olyan egyszerű felmérni a hálózatba kapcsolt számítógéprendszer viszonylagos biztonságát, mint a ház ajtaját. Esményi esetben mindig az összes általunk üzemeltetett hálózatba kapcsolt gépről pontosan tudjuk, hogy milyen hálózati folyamatok futnak rajta, gyakorlatilag viszont nehéz nyomon követni az adatokat a mai behálózott világban.

Ezért minden biztonságért aggódó szakembernek kötelező komolyan vennie minden eszközt, annak ellenére, hogy ezeket vetik be a rossz fiúk is. Ezekkel a programokkal nem érhetünk el sem teljes körű, sem tökéletes védelmet, de ez természetes is, ha olyan állandóan változó célponttal van dolgunk, mint a rendszerbiztonság. Egyetlen ilyen program, eljárás vagy eszköz sem szavatolhatja a teljes biztonságot. Ismételjük fennhangon: a biztonság folyamat, nem termék! Az elmondottakon kívül még egy oka van, amiért tanulmányozzuk a biztonsági pásztázókat: a hírnév. Jó érzés s00p3r 3L33T HaX0rnek kiadni magunkat. Felülmúlhatatlan érzés, amikor az nmap, a nessus vagy valami hasonló program használata közben ártatlan arccal mondhatjuk: dolgozunk.

### A pásztázók típusai és használatuk

Kétféle rendszerpásztázó ismeretes. A kapupásztázók nyitott TCP- és UDP-kapukat keresnek, amelyek mögött szolgáltatások figyelnek. A biztonsági pásztázók még egy lépéssel továbbmennek, és a beazonosított szolgáltatásokat ellenőrzik az ismert gyenge pontok szempontjából. A hétköznapi életből vett példával úgy mondhatjuk, hogy a kapupásztázó megszámolja egy ház ablakait és ajtóit, a biztonsági pásztázó mindenhova becsönget és ellenőrzi a riasztót az ablakokon. Ó, majdnem elfeledkeztünk az általános pingről, mely a harmadik-

fajta pásztázásnak tekinthető! Ez a hálózat egy megadott IP-címtartományát ellenőrzi végig, és a működő (azaz a pingre válaszoló) gépeket megtalálja. Ez is hasznos lehet, főleg ha pont ilyesmire van szükségünk, de területi okokból ez alkalommal csak a kapupásztázókat és a biztonsági pásztázókat fogjuk bemutatni. Minden itt tárgyalt eset igaz akár öt, akár 65 500 gépet pásztázunk.



Jellegzetes nmapfe munkafolyamat

### Az nmap, a kapupásztázás világbajnoka

A kapupásztázás elve egyszerű: ha kapcsolódni próbálunk egy kapuhoz, megállapíthatjuk, hogy zárva van, vagy egy alkalmazás (pl.: webkiszolgáló, FTP démon stb.) fogadja a kapcsolatot. Könnyű írni olyan egyszerű kapupásztázót, amely a helyi connect() rendszerhívást használva csatlakozik különböző TCP-kapukhoz. A megfelelő modulok segítségével még Perlben is lehet ilyet készíteni. Csakhogy ez a módszer a legerőszakosabb és legnyilvánvalóbb módja a pásztázásnak, az eredmény nagyszámú naplóbejegyzés lesz a célrendszeren. Itt kerül a képbe *Fjodor* nmap programja. Az nmap tud egyszerű connect() pásztázást is, de az igazi erőssége a „lopakodó pásztázás”. A lopakodó pásztázás hamis TCP-csomagok használatával történik, amelyek anélkül készítenek választ a célpontot, hogy a TCP-kapcsolat létrejött volna.

Az nmap nem egy, hanem négy lopakodó módszert ismer azonkívül, hogy ismeri a TCP-kapcsolat pásztázását, az UDP-pásztázást, az RPC-pásztázást, az általános ping, és még az operációs rendszer „ujjlenyomatát” is le tudja venni. Van még csomó más tulajdonsága is, amelyeket inkább a sötét oldal tud kihasználni, például az FTP visszhang pásztázása, ACK-pásztázás és Window tűzfalpasztázás, viszont ezek aligha érdeklik a Linuxvilág törvénytisztelő, erkölcsös

olvasóit. Jelenleg az nmap messze a legnagyobb tudású és legsokoldalúbb kapupásztázó.

Foglaljuk össze az nmap által ismert legfontosabb pásztázási típusokat: *TCP-kapcsolat pásztázása* – az OS beépített connect() rendszerhívását használja. A kiszemelt kapun teljes, háromlépcsős TCP-kézfogást kísérel meg (SYN, ACK-SYN, ACK). A sikertelen kapcsolatot az jelzi, ha a kiszolgáló a SYN csomagra ACK-RST-csomagot küld vissza, ez azt jelenti, hogy a kapu zárva van. Ez a módszer nem igényel rendszergazdai jogokat, és az egyik leggyorsabb is. Azonban ne lepődjünk meg azon, hogy a legtöbb kiszolgálóalkalmazás naplózza a megnyitás után azonnal bezárt kapukat, ezért ez elég „zajos” pásztázástípus.

*TCP-SYN pásztázás* – kétharmada a TCP-kapcsolat pásztázásának. Ha a célpont ACK-SYN csomagot küld vissza, az nmap rögtön RST-csomaggal felel, ahelyett, hogy felépítené a kézfogást ACK-csomaggal. Az ilyen „félíg nyitott” kapcsolatokat sokkal kevésbé naplózza a rendszer, ezért a SYN pásztázás sokkal kevésbé észrevehető, mint a TCP-kapcsolat pásztázása. Ennek az az ára, hogy rendszergazdaként kell futtatni az nmapot az üzemmód eléréséhez, mivel ilyenkor nem a rendszermag állítja össze a csomagokat.

*TCP-FIN-pásztázás* – még csak nem is tesz úgy az nmap, mint ami szabványos TCP-kapcsolatot kezdeményez. A kiszemelt célpontnak elküld egy FIN (befejezés) csomagot. Ha a célpont TCP-, illetve IP-verme megfelel az RFC-793 előírásainak (MS-bármí, HP-UX, IRIX, MVS és Cisco IOS nem), akkor a nyitott kapuk eldobják a csomagot, a zárt kapuk pedig RST-t küldenek.

*TCP NULL-pásztázás* – hasonló a FIN-pásztázáshoz, de itt jelöletlen TCP-csomag megy ki (nullcsomag). Szintén az RFC-793 megfelelést használja ki.

*TCP-karácsonyfa pásztázás* – hasonló a FIN-pásztázáshoz, de itt a kimenő csomag FIN, PSH és URG jelölést kap (befejezés, adattolás és sürgős). Szintén függ a fent említett RFC-793 megfeleléstől.

*UDP-pásztázás* – az UDP-kapcsolat nélküli protokoll, azaz semmilyen függőség nincs meghatározva a protokollban az egyes csomagok között semelyik irányban. Ezért az UDP-nél nem játszhatunk a TCP-nél megismert kézfogással. A legtöbb OS TCP-, illetve IP-verme azonban „elérhetetlen kapu” ICMP-csomagot küld vissza, ha UDP-csomag zárt UDP-kaput vesz célba. Ebből következik, hogy az a kapu, amely nem küld vissza ICMP-csomagot, feltehetőleg nyitva van. Mivel sem a próbacsomag, sem a válaszként érkező ICMP-csomag nem ér biztosan célba (ne feledjük, az UDP-kapcsolat nélküli protokoll, és az ICMP is az), az nmap általában több UDP-csomagot küld a kiszemelt UDP-kapura, hogy a hamis eredményeket kiszűrje. Tapasztalatunk szerint az nmap UDP-pásztázásának pontossága különböző a célba vett operációs rendszerek függvényében, de a semminél jobb.

*RPC-pásztázás* – egyéb pásztázási típusokkal együtt használva, ezzel deríthető fel, hogy mely kapuk vannak nyitva RPC (távoli eljárás-hívás) céljára, mik a kapuk mögött figyelő szolgáltatások és mi a változatszámuk.

*Általános ping (söprés)* – lásd fenn a pásztázók típusai és használatuk részt.

Huhh! Szép kis lista, pedig kihagytuk az ACK-pásztázást és a Window pásztázást (ha érdekel, nézd meg az nmap(1) sűgőoldalt). Az nmapnak van még egy nagyon hasznos tevékenysége, az OS ujjenyomatának levétele. Az nmap a célpont a különböző csomagok adott válaszainak elemzéséből elég jól meg tudja határozni a célponton futó operációs rendszer típusát.

## Az nmap beszerzése és telepítése

Manapság olyan népszerű az nmap, hogy a legtöbb Linux-változat tartalmazza. A RedHat 7.0 és a Debian 2.2, ezeket jelenleg is használom, tartalmazzzák az nmapot (az Applications/System, illetve az

### 1. lista Az „alapértelmezett” nmap pásztázás

```
[root@sprecher /root]# nmap 10.123.123.9
Starting nmap V. 2.53 by fyodor@insecure.org
(www.insecure.org/nmap/)
Interesting ports on (10.123.123.9):
(The 1520 ports scanned but not shown below
are in state: closed)
Port      State      Service
80/tcp    open      http
139/tcp   open      netbios-ssn
1432/tcp  open      blueberry-lm

Nmap run completed--1 IP address (1 host up)
scanned
in 2 seconds
```

Extra/Net alá besorolva). A Linux-felhasználók ezért a legegyszerűbben rendszerük csomagkezelőjével (pl.: RPM, dselect, YAST) telepíthetik az nmapot a nekik tetsző forrásból (CD-ROM, FTP stb.). Ha azonban a legfrissebb változatra, illetve a forráskódra vágyunk, mindkettőt letölthetjük a <http://www.insecure.org/> webhelyről (Fjodor honlapja) RPM és TGZ formátumban. Ha a forrásból önállóan szeretnénk lefordítani az nmapot, adjuk ki a következő parancsokat (természetesen a forráskódot tartalmazó könyvtár neve más is lehet, mert az nmap 2.53 esetleg már elavul, mire e cikk eljut az olvasókhöz):

```
cd nmap-2.53
./configure
make
make install
```

## Az nmap használata

Az nmap két különböző módon futtatható. A leghatékonyabb és legrugalmasabb mód a parancssor. Emellett nmapfe néven létezik grafikus felület is, mellyel összerakhatók az nmap kapcsolói, és a program futtatható (lásd az 1. ábrát).

A GUI hasznos, ha csak gyorsan és felületesen akarunk pásztázni, vagy az nmap parancssori kapcsolóival ismerkedünk. Azonban nagyon ajánlott rendszeresen megtanulni az nmap parancssoros használatát, mert a GUI csupán a lehetőségek egy részét teszi elérhetővé. Arról nem is beszélve, milyen pazarló dolog X-et indítani a jó öreg nmap kedvéért... Az nmap parancsokat könnyű tanulni. Az egyszerű pásztázás így érhető el: `nmap -s pásztázás típusa -p kaputartomány cél`.

A `-s` kapcsolót azonnal követi a következők egyike:

- T: TCP-kapcsolat pásztázása
- S: TCP-SYN-pásztázás
- F: TCP-FIN-pásztázás
- N: TCP-NUL-pásztázás
- X: TCP-karácsonyfa pásztázás
- U: UDP-pásztázás (a fentiekkel együtt is használható)
- R: RPC-pásztázás (a fentiekkel együtt is használható)

A `-s` kapcsoló adja meg, hogy milyen típusú pásztázást kell futtatni. Ezt követően megadhatjuk bármelyik TCP-pásztázási típust, U-t az UDP-pásztázáshoz, R-t az RPC-pásztázáshoz, illetve azonosításhoz, vagy ezek tetszőleges kombinációját. Az egyetlen megkötés, hogy egyszerre csak egyféle TCP-pásztázási típus adható meg. Ha a `-s` kapcsolót elhagyjuk, az alapértelmezés a TCP-kapcsolat pásztázása. Például a `-sSUR` arra utasítja az nmapot, hogy a célpontra végezzen el SYN pásztázást, majd UDP-pásztázást, végül RPC-pásztázást,

illetve azonosítást. A `-sTSR` nem működne, mert a TCP-kapcsolat pásztázása és a SYN pásztázás egyaránt TCP-pasztázások. Ha megadod a kaputartományt a `-p` kapcsolóval, a vesszők és kötőjelek használatával pontosan szabályozhatod a pásztázandó kapuk csoportját. Például a `-p 20-23,80,53,600-1024` hatására az nmap 20-tól 23-ig és 600-tól 1024-ig pásztáz, valamint pásztázza a 8-as és 53-as kapukat. A kaputartományok felsorolásában nem lehet szóköz.

Hasonlóképpen a célpont kifejezése is lehet gépnév, gép IP-címe, hálózat IP-címe vagy IP-címtartomány. Például a `192.168.17.*` mind a 255 IP-címet magában foglalja (valójában írhatnánk `192.168.17.0/24-et` is); a `10.13.[1,2,4].*` jelentése `10.13.1.0/24`, `10.13.2.0/24` és `10.13.4.0/24`. Láthatjuk, hogy az nmap nagyon rugalmas, sokféle célpontkifejezést megért.

### Az életből ellesett példák – néhány egyszerű pásztázás

Mielőtt továbblépnénk, vizsgáljunk meg néhány egyszerű pásztázást, amelyek az eddig ismertetett kapcsolókat használják! Az ebben a fejezetben megadott példák az nmap 2.53 változatával (az írás pillá-

natában ez a legújabb) futnak RedHat 7.0-n. A célrendszer a példákban Windows 98-at futtat Samba kiszolgálóval.

Tegyük fel, hogy először „alapértelmezett” pásztázást szeretnénk végrehajtani az nmap programmal. Nem kell megadnunk kapcsolóját, ha nem szeretnénk. Ha csak a célpont IP-címét vagy egy IP kifejezést adunk meg, az nmap minden célgépet megpingel, azután végigpásztázza TCP-kapcsolat pásztázása üzemmódban az 1–1024 kaputartományt és a `/usr/share/nmap/nmap-services` fájlban (a fájl elérési útja különbözhet) felsorolt kapukat, összesen 1523-at. Az 1. lista megmutatja, hogyan néz ki egy ilyen alapértelmezett pásztázás, ha Windows 98 a célpont.

Csak két másodpercet vett igénybe az 1523 kapu lekérdezése, tehát nem a levegőbe beszéltünk, amikor gyorsnak neveztuk a TCP-kapcsolat pásztázását.

A következő pásztázási példában vegyük az UDP-t is az eddigiek mellé, és ha már ott vagyunk, nézzük meg, hogy vannak-e RPC-alkalmazásokat futtató nyitott kapuk! Mivel az UDP-pasztázást nem a TCP-kapcsolat pásztázása helyett szeretnénk, hanem vele együtt, ezért most már azt is ki kell írni a parancsorbán. A kiadandó parancs a 2. listában látható.

© Kiskapu Kft. Minden jog fenntartva

## Mi az a kapu?

A TCP/IP-verem (a rendszer mag TCP/IP támogatását megvalósító protokoll-meghajtók halmaza) az egyes hálózati alkalmazásokat és munkafolyamatokat a kapuszámok alapján különbözteti meg. Minden egyes alkalmazás/démon/folyamat, amelynek TCP-n vagy UDP-n keresztül kell kapcsolatokat kezelnie, rendelkezik kapuszámmal. A kapuszámok 0-tól 65.536-ig terjedhetnek, és két sorozat van, egy a TCP-hez és egy az UDP-hez.

Más szavakkal, a 2000. TCP-kaput használhatja egy folyamat, és a 2000. UDP-kaput egy másik, feltéve, hogy az első TCP protokollt használ, a második UDP-t. A 0-tól 1023-ig terjedő tartományt csak a rendszergazdai jogokkal felruházott folyamatok használhatják, ezért ezeket rendszerkapuknak vagy kivételezett kapuknak hívják. Az Internet Assigned Numbers Authority (IANA) fenntart egy listát a hivatalos kapuszám-hozzárendelésekről: például a telnet többé-kevésbé mindenütt a 23-as TCP-kaput használja, mert az IANA szerint azt kell használnia.

Ákár csak az Internet sok más vonatkozásában, ebben az esetben is inkább hagyományról van szó, mint nemzetközi szabványról. Senki sem akadályozhat meg egy programozót abban, hogy írjon egy olyan alkalmazást, amely nem telnet és mégis a 23-as TCP-kaput használja. Mégis, ha pásztázunk egy gépet, és a 23-as TCP-kaput működik, nagy

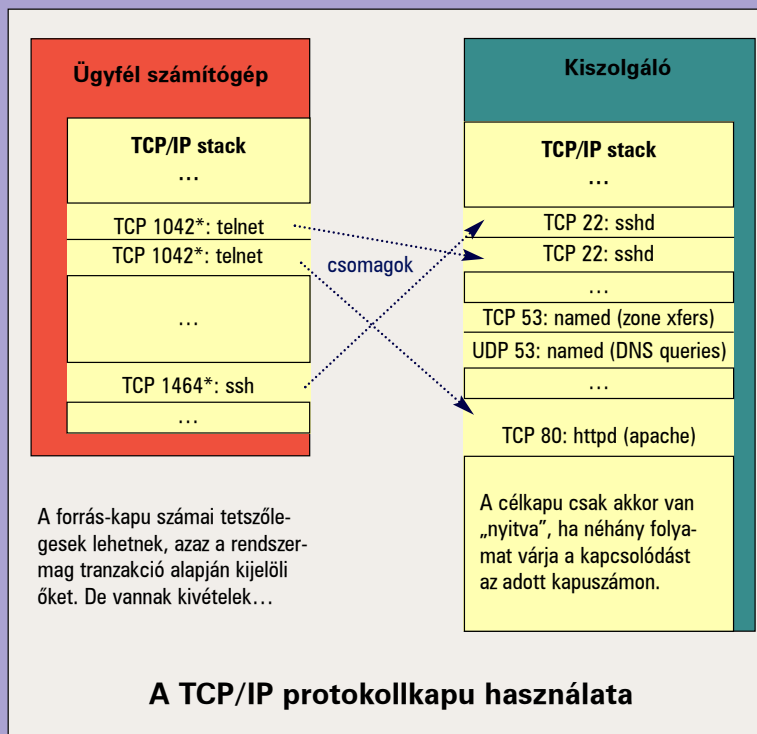
valószínűséggel a Telnet démon figyel mögötte.

Az 1024-től 49151-ig terjedő tartományt a rendszergazdai jogokkal nem rendelkező folyamatok használják, és az IANA szintén bejegyezte ezeket (a „közösség kényelme érdekében”). Ezeket a kapukat bejegyzett kapuknak hívják. A 49152-től

65535-ig terjedő tartomány használható olyan folyamatok számára, amelyek nem kapcsolódnak az Internet-hez, ezért ezeket a kapukat dinamikus vagy magánkapuknak hívják. Még egy fogalommal meg kell ismerkednünk a TCP-, illetve UDP-kapuk kérdéskörében, ez a forrás-, és célkapuk közötti különbség. Minden TCP- és UDP-csomag rendelkezik célkapuval (erre a kapura megy az elküldött csomag) és forráskapuval (innen küldték a csomagot, és a távoli szolgáltatás vagy démon ide küldi a választ).

A célkapukkal ellentétben a forráskapukat általában a rendszermag dinamikusan osztja ki, az alkalmazástól függően, minden egyes átvitelre külön.

A kapuszámok hozzárendelése a szolgáltatásokhoz a rendszer `/etc/services` fájljában csak a helyi figyelő-, illetve célkapukra vonatkozik (pl.: telnetd), és nem a helyi kifelé irányuló folyamatok forráskapuira (pl.: telnet). Bizonyos alkalmazások, mint az NFS és a NIS, a célkapukra is dinamikusan hozzárendelést alkalmaznak, de ez teljesen más téma.



### A TCP/IP protokollkapu használata

2. lista nmap pásztázás TCP kapcsolatra, UDP-re és RPC-re

```
[root@sprecher /etc]# nmap -sTUR 10.123.123.9

Starting nmap V. 2.53 by fyodor@insecure.org
(www.insecure.org/nmap/)
Interesting ports on (10.123.123.9):
(The 3075 ports scanned but not shown below
are in state: closed)

Port      State      Service (RPC)
80/tcp    open      http
111/udp   open      sunrpc (rpcbind V2)
137/udp   open      netbios-ns
138/udp   open      netbios-dgm
139/tcp    open      netbios-ssn
1026/udp  open      (rpcbind V2)
1432/tcp  open      blueberry-lm

Nmap run completed-1 IP address (1 host up)
scanned in 14 seconds
```

3. lista Kijelölt TCP- és UDP-kapuk pásztázása

```
[root@sprecher /root]# nmap -sTU -p
1-1024,12345,12346,31336 10.123.123.9

Starting nmap V. 2.53 by fyodor@insecure.org
(www.insecure.org/nmap/)
Interesting ports on (10.123.123.9):
(The 2049 ports scanned but not shown below
are in state: closed)

Port      State      Service
80/tcp    open      http
111/udp   open      sunrpc
137/udp   open      netbios-ns
138/udp   open      netbios-dgm
139/tcp    open      netbios-ssn

Nmap run completed-1 IP address (1 host up)
scanned in 7 seconds
```

A -sU és a -sR pásztázás (együtt -sTUR) különösen jól együttműködnek: az RCP sokat használja az UDP-t. Ha az nmap RPC-szolgáltatást talál egy nyitott kapun, kiírja zárójelben az alkalmazás nevét és változatszámát is (ha ki tudja találni).  
 Tegyük fel, hogy kicsit szűkíteni szeretnénk a pásztázás tartományát. Ez akkor lehetséges, ha sejtjük, hogy mit futtat a cél gép, illetve a pásztázási időt szeretnénk lerövidíteni. Megadhatjuk a használandó kapukat a -p kapcsoló utáni felsorolásban. Vesszők és kötőjelek használhatók ebben a listában, de szóköz nem. A 3. lista olyan pásztázást mutat be, amelyben minden kivételezett kaput ellenőrzünk, valamint néhány veszélyes kaput, a TCP 12345-öt és 12346-ot (NetBus alapértelmezett kapui) és az UDP 31337-et (BackOrifice alapértelmezett kapuja).  
 Végül lássuk, hogyan pásztázhatunk egyszerre több gépet! Könnyen. A gépeket még rugalmasabban adhatjuk meg az nmapnak, mint

4. lista Néhány lehetőség középhaladóknak

```
[root@sprecher]# nmap -sTUR -OIF -oN lamer.txt
1.12.123.4

Starting nmap V. 2.53 by fyodor@insecure.org
(www.insecure.org/nmap/)
Interesting ports on bookoosvr (1.12.123.4):
(The 2153 ports scanned but not shown below
are in state: closed)

Port      State      Service (RPC)      Owner
22/tcp    open      ssh                root
23/tcp    open      telnet             root
25/tcp    open      smtp               root
80/tcp    open      http               apache
111/tcp   open      sunrpc (rpcbind V2)  rpc
111/udp   open      sunrpc (rpcbind V2)
113/tcp   open      auth               nobody
443/tcp   open      https              apache
515/tcp   open      printer            lp
587/tcp   open      submission         root
999/udp   open      applix             root
1024/tcp  open      kdm (status V1)    rpcuser
1024/udp  open      (RPC (Unknown Prog #))
1025/udp  open      blackjack (status V1)
1241/tcp  open      msg                root
3001/tcp  open      nessusd            root
6000/tcp  open      X11                root

TCP Sequence Prediction:
  Class=random positive increments
  Difficulty=613547 (Good luck!)

Remote operating system guess:
  Linux 2.1.122 - 2.2.14

Nmap run completed-1 IP address (1 host up)
scanned in 959 seconds
```

a kapukat: használhatók helyettesítő karakterek, szögletes zárójelek (listák) és „per/álhálózat bites” jelölés. Így nézne ki a 3. listán megadott pásztázás az egész hálózatra (254 cím, az eredményt most elhagytuk):

```
nmap -sTU -p 1-1024,12345,12346,31336
10.13.13.0/24
```

**nmap középhaladóknak**

Az nmap rengeteg ijesztő képességgel rendelkezik, átoson tűzfalonon, vigyáz, hogy ne hozza működésbe a behatolást figyelő programokat, és más módokon is segíti a felhasználót, hogy elkerülje a lebukást. Nem érzek készletét arra, hogy ezeket a dolgokat itt tárgyaljam, bár kétségkívül ezeket is lehet törvényes célokra használni. A cikk hátralevő részét inkább olyan dolgoknak szenteltem, amelyek nem függenek össze ilyen egyértelműen a betöréssel.  
 Tegyük fel, hogy nagy hálózat rendszergazdái vagyunk, és valaki telepít egy kiszolgálót a gépterünkben, amely úgy tűnik, elérhető az Internetről is. Ez sértheti a szervezet biztonsági szabályzatát (vagy csak a mi önézetünket, mert nem kértek tőlünk engedélyt a telepítésre). Mielőtt dühödten érvényre juttatnánk jogainkat, előbb kicsit szimatoljunk körül, milyen veszélyeknek van kitéve a hálózatunk!  
 Szerencsére a titokzatos kiszolgálóra valaki felírta lila krétával az IP-címét. Az is szerencse, hogy nmap-tudásunkkal felvélteze

a bosszú igazságos angyalaként láthatunk munkához. Íme néhány lehetőség az nmap használatára ebben a helyzetben.

Először is, milyen operációs rendszer fut a kiszolgálón? Az OS ujjlenyomat megmondja, ehhez a `-O` kapcsolót kell megadni. A `-O` használatok az nmap mindenféle jelzéssel ellátott TCP-csomagokat küld, és a válaszokat összeveti az OS ujjlenyomat-adatbázissal (`/usr/share/nmap/nmap-os-fingerprints` a Red Hat 7.0 rendszeren). Tapasztalatunk szerint nagyon jól működik a felismerés, kivétel a MacOS 8 esetén, ez ugyanis összezavarja.

Következő kérdés, vannak-e rendszergazdai jogokkal rendelkező szolgáltatásokat futtató nyitott kapuk. Természetesen néhány szolgáltatásnak szüksége van ilyen széles körű jogosultságokra, a legtöbbnek azonban nincs. Ha a webkiszolgáló a rendszergazda nevében fut ezen a gépen, akkor valakinek ezért felelnie kell, az biztos. Használjuk a `-l` kapcsolót a célpont ident démonjának lekérdezésére, ennek egyetlen célja, hogy szétkürtölje a világban, hogy melyik felhasználó birtokában vannak az egyes szolgáltatások.

Lehetséges csökkenteni annak az esélyét, hogy erőszakos pástázásunkkal túlterheljük a célrendszert vagy a hálózatot? Természetesen. A `-T` kapcsolóval megadható az időzítés módja. A lehetőségek: Paranoid, Sneaky, Polite, Normal, Aggressive és Insane (üldözési mániás, lopakodó, udvarias, szokásos, erőszakos és örült). A sorrend a hálózat egyre nagyobb megterhelésének felel meg, és a háttérben az húzódik meg, hogy az nmap mennyi időt vár az egyes csomagok elküldése között, és hogy sorban vagy kötegekben küldi ki őket. A `-T Polite` jó választás, ha finoman szeretnénk bánni a célponttal, illetve a hálózattal.

Hogyan végezzünk gyorsan a pástázással? Csak a valószínű szolgáltatásokat akarjuk ellenőrizni, és nem szeretnénk az összes kiemelt kaput végigpásztázni? A `-F` kapcsoló arra utasítja az nmapot, hogy csak az nmap-services fájlban felsorolt kapukat pástázza. Így elkerülhető, hogy olyan kapukat is ellenőrizzünk, amelyek valószínűleg nem adnának érdekes eredményt.

### Kapcsolódó címek

Fjodor hivatalos nmap oldala

➔ <http://www.insecure.org/nmap/>

Fjodor „The Art of Port Scanning” című cikke az nmap utasításforma tekintetében elavult, de nagyszerűen leírja, hogyan működik a kapupásztázás általában, és bemutatja a lopakodó pástázás működési elvét

➔ [http://www.insecure.org/nmap/nmap\\_doc.html](http://www.insecure.org/nmap/nmap_doc.html)

Fjodor szórakoztató cikke az nmap OS-ujjlenyomatfelismerő képességéről ➔ <http://www.insecure.org/nmap/nmap-fingerprinting-article.html>

Rik Farror cikke a Network Magazine-ból következő módon írja le az nmap OS-ujjlenyomatfelismerő képességét

➔ [http://www.insecure.org/nmap/press/network\\_magazine-system\\_fingerprinting.txt](http://www.insecure.org/nmap/press/network_magazine-system_fingerprinting.txt)

Az IANA hivatalos listája a jól ismert, bejegyzett és magán, illetve a dinamikus kapukról ➔ <http://www.isi.edu/in-notes/iana/assignments/port-numbers/>

A hivatalos Internet Engineering Task Force (IETF) RFC tárolóhely. Fúrja az oldalad, hogy mit jelent az RFC-793-megfelelőség? Ne töprengj tovább! Különös figyelmet érdemel az RFC 793 (TCP protokoll), az RFC 768 (UDP protokoll) és az RFC 1413 (Ident protokoll)

➔ <http://www.ietf.org/rfc.html>

Végül, van arra egyszerű lehetőség, hogy a bénaság bizonyítékát szövegfájlként mentjük? A `-oN` fájlnev az eredményeket szövegfájlba menti. Ha a `V@gÁny DuMáT` részesítjük előnyben, használhatjuk a `-oS` kapcsolót (az „S” a „Script-Kiddie-Duma” rövidítése). A 4. listán láthatjuk, hogy az engedély nélkül telepített kiszolgáló sok más mellett fogadja a Secure Shell, a Telnet, a HTTP/SSL, az LPD, az X és a nessus kapcsolatokat. A nessus? Mi a baj vele, hiszen ez egy biztonsági pásztázó. Nem akarjuk, hogy a hálózatunkon található nessus-kiszolgáló látszódjék az Internetről – a következő hónap témája pont ez lesz.

Az nmap nagyon hatékony, de a nessus még egy lépéssel továbbvisz, és megszondázza azokat a kapukat, amelyeket az nmap talált, nincsen ismert gyenge pontjuk. Ismét arra fogunk törekedni, hogy ezeket a hatékony eszközöket a jóra használjuk, és ne rosszra.



Mick Bauer (mick@visi.com)

alkalmazott biztonsági vezető az ENRGI hálózatmérnöki és tanácsadó cég minneapolis-i részlegénél. 1995 óta Linux-rajongó, és 1997 óta vakbuzgó OpenBSD-s. Különös élvezetét leli abban, hogy ezeket az élvonalbeli operációs rendszereket rávegye, hogy elavult roncsokon fussanak. Mick szívesen vesz minden kérdést és hozzászólást.

## Typescript



Rögzítene a terminál kimenetét, de nem mindent lehet átírányítani?

Használd a `script` parancsot. A parancs kiadása után a terminálon megjelenő összes szöveget rögzíti a `typescript` nevű fájlban.

```
tux@coollinuxbox:/home/tux$ script
script: WARNING: script session is not secure
against eavesdropping/hijacking!
script: read /usr/doc/bsdutils/README.script
for details.
Script started, output file is typescript
tux@coollinuxbox:/home/tux$ python
python parancsok
Control-D
tux@coollinuxbox:/home/tux$ exit
Script done, output file is typescript
tux@coollinuxbox:/home/tux$ cat typescript
Script started on Thu Oct 12 12:03:22 2000
tux@coollinuxbox:/home/tux$ python
Python 1.5.2 (#1, Dec 15 1999, 11:15:06) [GCC
2.7.2.3] on linux2
Copyright 1991-1995 Stichting Mathematisch
Centrum, Amsterdam
>> 45+89+12.25+63.21
209.46
>> 70/12
5
>> 70%12
10
>>
tux@coollinuxbox:/home/tux$ exit
Script done on Thu Oct 12 12:04:43 2000
tux@coollinuxbox:/home/tux$
```

## A PostgreSQL és a PHP

**V**arázsoljuk PostgreSQL-lel tárolt adatbázisainkat a Webre, és egyúttal ismerkedjünk meg napjaink egyik leggyorsabban fejlődő programozási nyelvével, a PHP-vel. A cikksorozat első részében telepítjük a PHP-t (3-as vagy 4-es változat) Debian alá, képessé tesszük PostgreSQL adatbázisok elérésére, és megismerkedünk néhány egyszerű PHP utasítással is.

### Miért Debian?

A változtatválasztásom oka egyszerűen az, hogy én ezt használom. A programok és a telepítés is Debianon lett kipróbálva, ennek ellenére működni kell a többi változat alatt is. Elképzelhető, hogy a beállítófájlok helye más (például RedHat esetén), de a PHP programoknak és a beállító fájlokban levő beállításoknak ugyanígy kell működniük. Debian esetében a csomagok kiterjesztése .deb és a csomagkezelő program a dpkg, míg például RedHat esetében .rpm a kiterjesztés és rpm a csomagkezelő neve. RPM csomagokat az RPM -i csomagnév utasítással lehet telepíteni. RedHat esetén a php3.ini-t a /etc/httpd könyvtárban, az Apache webkiszolgáló-beállító állományait pedig a /etc/httpd/conf könyvtárban találjuk.

### A PHP működésének ellenőrzése

Írásomban feltételezem, hogy az olvasó működő Apache webkiszolgálóval rendelkezik. Ennek telepítését ugyanis nem írom le. Legyen egy felhasználónk, akinek a weboldalait meg tudjuk nézni, és akinek a nevében bejelentkezve weboldalakat tudunk készíteni. A szükséges beállítások elvégzéséhez természetesen ismernünk kell a rendszer-

gazda jelszavát is. Helyezzük el a **phpinfo.shtml** fájlt (*1. lista*) a felhasználónk webes könyvtárában (ez általában a felhasználó saját könyvtárának public\_html nevű alkönyvtára), adjunk olvasási jogot mindenkinek (chmod o+r phpinfo.shtml), és próbáljuk elérni egy böngésző segítségével!

Ha látunk egy színes oldalt (*1. kép*), akkor minden rendben, működik a PHP, átugorhatjuk a következő részt. Ha bármi más történik, akkor vagy nincs telepítve PHP, vagy beállítási gondok vannak. Ellenőrizzük, telepítve van-e a php3 csomag! Debianon ezt a dpkg -l 'php3' paranccsal

nézhetjük meg. Ha a képernyőn a legelső oszlopban nem ii (*2. lista*) van, akkor telepítsük fel.

Használhatunk php4-et is, a cikk összes példája működik mindkét változattal, a két változat beállítása is szinte ugyanaz.

### A php3 telepítése

A php3 csomagot Debian alatt többféleképpen telepíthetjük. Használhatjuk a dselect programot: a főképernyőnél „/”-t nyomunk, majd beírjuk: „php”. A kívánt csomagoknál pluszt nyomunk, majd a csomagválasztó menüből kilépve az „Install” pontot választjuk. Ha ismerjük a .deb fájl nevét és helyét, használhatjuk a dpkg parancsot is (*3. lista*):

*2. lista* A sor elején levő második betű (*n=Not installed*) mutatja, hogy nincs telepítve a php3. Ha ez i (*i=Installed*), akkor minden rendben

```
$ dpkg -l 'php3'
Desired=Unknown/Install/Remove/Purge
| Status=Not/Installed/Config-
files/Unpacked/Failed-config/Half-installed
| / Err?=(none)/Hold/Reinst-required/X=both-
problems (Status,Err: uppercase=bad)
| | Name Version Description
+++
=====
pn php3 <none> (no description available)
```

*3. lista* A php3 telepítése. A LoadModule-lal kezdődő sort kell megjegyeznünk

```
monkey:~# cd /cdrom/dists/slink/main/
binary-i386/web/
monkey:/cdrom/dists/slink/main/binary-i386/web#
dpkg -i php3_3.0.5-3.deb
Selecting previously deselected package php3.
(Reading database ... 28718 files and
directories currently installed.)
Unpacking php3 (from php3_3.0.5-3.deb) ...
Setting up php3 (3.0.5-3) ...
I see you have apache webserver installed and
so far you haven't
used the apache module version of php3 in your
apache. If you want to use it, you should
reconfigure the apache webserver and select to
load the php module. I can call the
apacheconfig script now for you to do it, or
you can insert the following line into
/etc/apache/httpd.conf manually:
LoadModule php3_module
/usr/lib/apache/1.3/libphp3.so
Do you want me to run the apacheconfig script
now [y/N] ?
Ok, not running apacheconfig. Please read the
docs in /usr/doc/php3
Press RETURN to continue
```

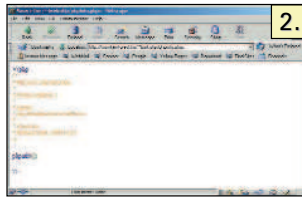
```
dpkg -i php3_3.0.5-3.deb
```

A kérdésre, hogy indítsa-e el a telepítő a webkiszolgáló beállítóprogramját, csak nyomjunk ENTER-t, és inkább állítsuk be mi kézzel! Ezután ki is írja, mit kell majd beszúrnunk. A PHP-hoz tartozó leírások külön csomagban találhatóak:

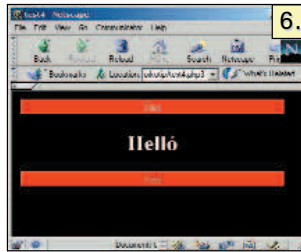
```
dpkg -i php3-doc_3.0.5-3.deb
```

Ha a négyes változatot kívánjuk használni, természetesen a php4 nevű csomagokat kell telepíteni.

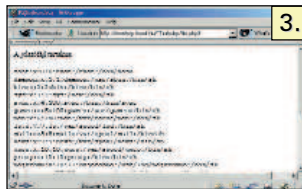




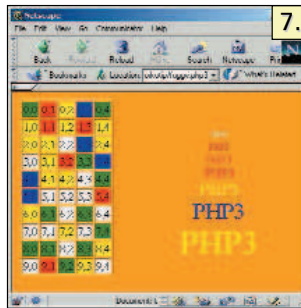
2.



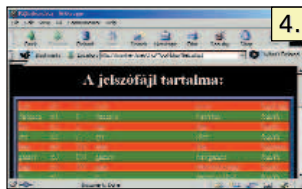
6.



3.



7.



4.



5.

Ha sikerült a telepítés, felkerül a gépünkre a *php3.ini* fájl, ami a php3 beállításait tartalmazza. Ezt Debianon a */etc/php3/apache* könyvtárban találhatjuk meg (vagy a *php.ini* fájlt a */etc/php4/apache* könyvtárban). Egyelőre hagyjuk változatlanul, és módosítuk az Apache beállítóállományait!

## Beállítás

A */etc/apache/srm.conf* fájlban keressünk rá a php3 szóra! Alapértelmezésként megjegyzésben vannak a php3-ra vonatkozó részek, szedjük ki a # jeleket. Mivel a következő php program példák kiterjesztése php3, ezért módosítuk is az

```
AddType application/x-httpd-php3
```

sort, hozzáfűzve a php3 szót (4. lista

➔ [www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)). Az

```
AddType application/x-httpd-php3-source .phps
```

sorból látható, hogy a phps kiterjesztésű fájlokat php3 forrásként kezeli, azaz meg tudja jeleníteni a tartalmukat (2. kép).

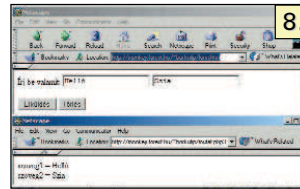
A */etc/apache/httpd.conf* fájlban keressük meg a LoadModullal kezdődő sorokat tartalmazó részt és írjuk a végére a libphp3.so modult betöltő sort (5. lista ➔ [www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)):

```
LoadModule php3_module
/usr/lib/apache/1.3/libphp3.so
```

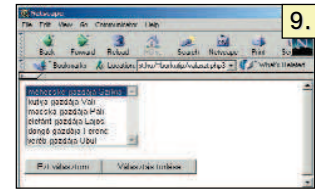
php4-hez az `AddType application/x-httpd-php` sorhoz fűzzük a `phtml php3 php4 php` szavakat, ugyanígy az `AddType application/x-httpd-php-source .phps` sor is kell. A `httpd.conf`-ba pedig a következő sort kell beírunk:

```
LoadModule php4_module
/usr/lib/apache/1.3/libphp4.so
```

Indítsuk újra a kiszolgálót a `/etc/init.d/apache restart` paranccsal és próbáljuk megnézni a `phpinfo.phtml` fájlt a böngé-



8.



9.

zőnkkel! Most már az 1. képhez hasonlót kell látnunk. Ha mégsem, ellenőrizzük a `php3` meglétét, az `srm.conf`-ot és a `httpd.conf`-ot!

## Ismerkedés a php3-mal

Helyezzük el a `public_html` könyvtárunkban a `phpinfo.php3` és a `phpinfo.phps` fájlokat! A `phpinfo.php3` betöltésekor ugyanazt kell kapnunk, mint a `phpinfo.phtml` megnézésekor (1. kép). Ha mégsem, akkor nem állítottuk be az `srm.conf`-ot. Ellenőrizzük az állományt a 4. lista alapján! A `phpinfo.phps` betöltésekor a böngészőben a forrásprogramot kell látnunk (2. kép).

A következő program a `file.php3` (6. lista) Tegyük ezt is a `public_html` könyvtárba és nézzük meg (3. kép).

A program megjeleníti a `/etc/passwd` fájl tartalmát a böngészőben (3. kép). Ez rengeteg adatot adhat egy betörő kezébe.

Gondolom, nem csak az én gyomrom rándul össze a `passwd` fájl tartalmát látva. A PHP igen erős eszköz a felhasználók kezében, jól gondoljuk meg, telepítjük-e éles rendszeren.

Vizsgáljuk meg a fájl tartalmát. Átlagos HTML-dokumentumot látunk, a

```
<? readfile('/etc/passwd'); ?>
```

sor kivételével, ami a `readfile()` php3 utasítást hajtja végre a kiszolgálóval.

A PHP kódot a HTML-oldalban helyezük el. Ezáltal a weboldalunk forrása áttekinthetőbb lesz, mint ha például Perl vagy C-alapú weboldal-előállító programot használnánk. Az utóbiainál ugyanis a programunk szerkezetét kell követnie a programba illesztett HTML-kódnak, míg a PHP-nál a PHP programkódot tesszük a HTML-oldal megfelelő helyére. Így nagyon könnyű módosítani a weboldal kinézetét. A PHP és a HTML-kód szétválasztására szolgál a `<? . . . ?>` tag. Tehát a PHP programunk részeit helyezzük el a `<? és a ?>` tagok között.

A PHP kódot a webkiszolgáló hajtja végre, ezért a forráskódunk nem jut el a böngészőhöz, csak a PHP program eredménye.

A `php3`-mal a további ismerkedéshez, csináltsuk az előző programot! A `file.php3` (7. lista ➔ [www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)) szintén a `passwd` fájl tartalmát írja ki, de most soronként olvassuk a fájlt, így lehetőség nyílik az egyes sorok színezésére és táblázatba helyezésére (4. kép). Nézzük át a fontosabb részeket! (A program-példa sorai számozottak, de ez csak a magyarázat kedvéért szükséges. Az eredeti fájlból természetesen hiányoznak a sorszámok.)

## A program szerkezete

Az 1–8 sorok normál HTML-fájl kezdősorai. A 9. sorban belépünk a `php3`-blokkba. Két változót hozunk létre értékadással. Minden változó neve \$ jellel kezdődik. Mindkét változó egy tömb. A `file()` utasítás a megadott fájl olvassa be, a tartalmát pedig tömbként adja vissza, ahol a tömb minden eleme a fájl egy-egy sorát tartalmazza. A 10. sorban létrehozuk a `$colors` tömböt, aminek két karakterlánc típusú eleme van. A 11. sorban kilépünk a `php3`-ból. A következő `php3` blokkot a 16–22-ik sorok mutatják. C-szerű számlálós ciklus a 16. sorban, a `count()` függvény egy tömb elemeinek számát adja meg. Mivel a tömbök első elemének sorszáma 0, így `$_i < count($lines)` következtében a tömb összes elemén végigmegyünk. A 17. sorban a `$color` változónak hol a



9. lista A php3.ini fájlba kerüljön bele az `extension=pgsql.so` sor

```

;;;;;;;;;;;;;;;;;;;;;;;;;
Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;;;;;;;;
; if you wish to have an extension loaded
; automatically, use the
; following syntax:
    extension=modulename.extension
; for example, on windows,
; extension=msql.dll
; or under UNIX,
; extension=msql.so
; Note that it should be the name of the
; module only,
; no directory information
; needs to go here. Specify the location of
; the extension with the extension_dir
; directive above.
; extension=pgsql.so
    
```

<TD BGCOLOR=red>, hol pedig <TD BGCOLOR=green> értéket adunk, attól függően, hogy \$i páros-e vagy sem. (Az a % b művelet az a és b között végzett osztás maradékát adja.) A tömbök elemeit \$tömbnév[index] hivatkozással kapjuk meg, a . (pont) művelet pedig karakterláncokat fűz össze). A 18–20. sor egyetlen kiíró utasítást tartalmaz, a kinézet kedvéért három részre osztva. A php3 macskakörmök közé tett karakterláncokban elhelyezett változóneveket az értékükkel helyettesíti. A 19. sor szorol bővebb magyarázatra, ugyanis itt vágjuk szét a jelszófájl egy sorát és ragasztjuk össze úgy, hogy a mezők között a </TD><TD BGCOLOR=...> szöveg legyen. Az eplode(határolójel, szöveg) függvény szétdarabolja a szöveget – jelen esetben a „:” a határolójel –, míg az implode(határolójel, szövegtömb) egyesíti egy karakterláncra a szövegtömb elemeit úgy, hogy minden elem közé a határolójelet teszi. A 20. sorban levő \n hatására a print utasítás új sor karaktert ír ki. Most tekintsük át röviden, hogyan vehetjük fel a kapcsolatot egy PostgreSQL adatbázissal!

### A PHP3 PostgreSQL moduljának ellenőrzése

Próbáljuk ki a test.php3 programot, (8. lista megtalálható: [www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)). Ez megpróbál kapcsolódni a `php_db` adatbázishoz a `php_user` nevű felhasználó nevében a `heureka` jelszóval. Ha a böngészőnk a *Fatal error: Call to unsupported or undefined function pg\_connect() in .../public\_html/test.php3 on line 2* hibaüzenetet adja, fel kell telepítenünk a `pgsql` modult a `php3`-hoz. Ez a `php3-pgsql_3.0.16-2.deb` fájl (`php4`-hez a `php4-pgsql_4.0b3-4.deb` fájl). A telepítéskor a `/usr/lib/php3/apache` könyvtárba bekerül a `pgsql.so` fájl. Ez az a modul, amit be kell állítanunk a `/etc/php3/apache` könyvtárban található `php3.ini` fájl segítségével. Szerkesszük ezt át! A `php3.ini` fájlban az `extension_dir`-rel kezdődő sor mutatja, hol kell lenniük a moduloknak. Ennek a `/usr/lib/php3/apache` értéket kell tartalmaznia. A *Dynamic Extensions* részbe írjuk be (9. lista): `extension=pgsql.so` (a „:”-vel kezdődő sorok megjegyzések). Ezután indítsuk újra az Apache-ot a `/etc/init.d/apache` restart utasítással! Éртеlemszerűen `php4`-hez a `/usr/lib/php4/apache` könyvtárban levő `pgsql.so` fájl és a `/etc/php4/apache` könyvtárban levő `php.ini` fájl

használjuk. Próbáljuk újra a `test.php3`-at. Ha a hibaüzenet nem változott, ellenőrizzük a `php3.ini` tartalmát és indítsuk újra az Apache-ot. Ha viszont az alábbi üzenetet kapjuk:

```
Warning: Unable to connect to PostgreSQL server:
FATAL 1: Database php_db does not exist in
pg_database in /home/.../public_html/test.php3 on
line 2
```

akkor minden rendben, a kapcsolat felépült. Ismerkedjünk kicsit a PostgreSQL adatbázis-kiszolgálóval. Kapunk hozzá egy konzolos segédprogramot, a `psql`-t. Ezzel hozzáférünk az adatbázisokhoz, táblákhoz és SQL-lekérdezéseket indíthatunk el. A programot a `psql` utasítással indíthatjuk el. A \? paranccsal segítséget kérhetünk. Minden SQL parancs végét pontosvessző zárja. A parancsok lehetnek többsorosak is, ekkor az utolsó sor végét zárjuk pontosvesszővel. A `psql` belső parancsai fordított perjellel kezdődnek és nem kell pontosvessző a lezárásukhoz. A \l kiírja az adatbázisokat, \connect *adatbázisnév* paranccsal kapcsolódhatunk a megadott adatbázishoz, \d kiírja az adatbázis tábláit. Ha még nem hoztuk létre saját nevünkhöz PostgreSQL-felhasználót, akkor tegyük a következőt rendszergazdaként:

```
su - postgres
createuser
```

Ezután be kell gépelnünk a bejelentkezési nevünket. A user ID-re csak nyomjunk ENTER-t. Saját magunknak érdemes engedélyezni az adatbázisok létrehozását, és hogy esetleg mi legyünk a PostgreSQL rendszerfelügyelői (akik felhasználókat is létrehozhatnak) úgyhogy a következő két kérdésre válaszoljunk y-nal. Következő lépésként hozzuk létre a `php_db` adatbázist. Ezt `psql`-ben a `create database php_db;` paranccsal tehetjük meg. Újra megpróbálva a `test.php3`-at, a *FATAL 1: SetUserId: user 'php\_user' is not in 'pg\_shadow'* üzenetet kell kapnunk, úgyhogy hozzunk létre a felhasználót is `psql`-ben:

```
create user php_user with password heureka
    ncreatedb ncreateuser;
```

utasítással. A `ncreatedb` hatására `php_user` nem tud adatbázisokat létrehozni, `ncreateuser` miatt pedig felhasználókat se. Ha a böngészőnk *Document contain no data* vagy hasonló üzenetet küld, helyben vagyunk. A `php3` képes elérni a PostgreSQL adatbázisokat (5. kép).

### Hogyan tovább?

A cikk folytatásában megismerkedünk a `php` PostgreSQL adatbázisokat kezelő függvényeivel, a PostgreSQL biztonsági beállításával. Közlünk egy webes alkalmazást és rövid útmutatót kapunk, hogyan mentjük át adatbázisainkat újabb változatú PostgreSQL alá.

Borkuti Péter (borkuti@freemail.hu)  
matematika-informatika szakos tanár, rendszergazda,  
informatikus, rendszerépítő és programozó.

#### Kapcsolódó címek

PHP levelezőlista ➔ <http://weblabor.hu/wl-phplista/>  
PHP tükör ➔ <http://hu.php.net> címről a Documentation részből letölthetjük a leírás magyar fordítását.

## Széles sávú internet-hozzáférés Magyarországon (2. rész)

Összevont számunkban megkezdjük a Magyarországon elérhető gyors internet-kapcsolatok és -szolgáltatások áttekintését. Akkor az otthoni, most az üzleti csomagok közül szemelgettünk.

**Ü**dvözlöm Olvasóimat a cikksorozat második részében, melyben a vállalkozások számára érdekes megoldásokat tekintem át!

Az Internet ma már a legtöbb szervezet működésében fontos szerepet játszik, legyen szó kis forgalmat bonyolító cégekről, vagy nagy ügyfélkörrel rendelkező vállalatokról. Az új évezred küszöbén a vásárlókkal, üzleti partnerekkel való kapcsolattartáshoz elengedhetetlen, hogy vállalkozásunk legalább levélküldési és -fogadási lehetőséggel rendelkezzen. Innen indulva a felső határ a csillagos ég – a következő lépcső egy saját honlap létrehozása lehet, melyen a látogatók a termékekről, szolgáltatásokról kaphatnak olyan tájékoztatást, melyhez máskülönbem nem, vagy csak körülményesen és drágábban juthatnának hozzá (például a postaládába dobott színes ismertetőkel). A legfejlettebb és egyben jelentős anyagi ráfordítást igénylő megoldás a webáruház létrehozása, ahol a vásárlók a felkínált készletből állíthatják össze megrendelésüket, melynek ellenértékét bankkártyával, csekken vagy más módon egyenlíthetik ki.

1. táblázat Az Axelero bérelt vonali szolgáltatásának árai

	Egyéves szerződés		Hároméves szerződés	
	Egyszeri díj	Havidíj	Egyszeri díj	Havidíj
64 kbit/mp	39 900 Ft	77 900 Ft	-	69 900 Ft
128 kbit/mp	44 200 Ft	111 900 Ft	-	99 900 Ft
128 kbit/mp felett	egyedi ajánlat			

Van már olyan egyszeri webkereskedelmi megoldás is, amely alacsony költséggel biztosítja a kisebb cégek elektronikus boltjainak megoldását. Amennyiben a céges internet-hozzáférést főként levelezésre kívánjuk használni, akkor mindenképpen olvassuk el az áprilisi számban talál-

3. táblázat A GTS-DataNet forgalomfüggő bérelt vonali szolgáltatása

Havidíj	Forgalmi díjak		Kapudíjak	
	Ft	Ft/GB	Sebesség	Ft
0-5 GB	-	25 830	64 k	25 900
5-10 GB	129 150	24 150	128 k	44 900
10-20 GB	249 900	7 350	256 k	69 900
20-40 GB	323 400	1 350	512 k	89 900
40-60 GB	350 400	1 890	1 M	99 900
60 GB fölött	700 000	-	2 M	109 900

ható, Stew Benedict tollából származó cikket (60. oldalon). Ez igazából egy trükk, melynek segítségével egyetlen, a szolgáltatótól kapott levélcímünket az iroda összes munkatársa használhatja saját levelezésére. A módszer bármilyen előfizetéssel (hagyományos kapcsolt vonali, ADSL, kábeltévé stb.) működik.

Ennek is vannak azonban hátrányai, hiszen egy tévesen megcímezett levél nem biztos, hogy a megfelelő felhasználóhoz érkezik. Ha biztosra akarunk menni, akkor komolyabb megoldás után kell néznünk. Az egyik legkézenfekvőbb választás valamilyen folyamatos internet-elérés, például bérelt vonal kiépítése.

### Bérelt vonal

A bérelt vonali kapcsolat lényege, hogy szolgáltatónk egy, csak a megrendelő által használt vonalat épít ki a megrendelő telephelye és a szolgáltató hálózata között, melyen közvetlen és állandó internet-kapcsolatot valósít meg. A bérelt vonalak sávszélessége általában 64 kbit/mp és 2 mbit/mp között mozog, ez 64 kbit/mp-es lépésként, illetve kétszereséssel (64, 128, 256, 512...) állítható be. A bérelt vonali internet-hozzáférés sok helyütt elérhető az országban.

2. táblázat Az UPC Broadband szolgáltatásának árai

	Privát	Profi	Max	SoHo	Business
<b>Adatforgalom</b>	1 GB	3GB	5 GB	15 GB	Egyéni megegyezés szerint
<b>Webtárhely</b>	3 MB	5 MB	10 MB	20 MB	Egyéni megegyezés szerint
<b>E-mail postafiók</b>	1-3	1-5	1-5	1-10	Egyéni megegyezés szerint
<b>Sávszélesség</b>	300/64 kb/mp mindegyiknél				
<b>Munkaállomások száma</b>	1	1	1	10	Egyéni megegyezés szerint
<b>IP-címek</b>	1 nyilvános cím	1 nyilvános cím	1 nyilvános cím	Minden munkaállomáshoz 1 nyilvános cím	Egyéni megegyezés szerint
<b>Kereskedelmi kiszolgáló üzemeltetés</b>	Nem	Nem	Nem	Nem	Egyéni megegyezés szerint
<b>Árak:</b>	7760 + Áfa	9840 + Áfa	13 200 + Áfa	44 000 + Áfa	Egyéni megegyezés szerint

4. táblázat A Tvnet Kft. bérelt vonali szolgáltatása

Díjcsomag	64 kbit/mp	128 kbit/mp	64 kbit/mp		
VIP				128 kbit/mp	
VIP					192 kbit/mp – 4 Mbit/mp
Havi előfizetési díj	69 000 Ft	109 000 Ft	89 000 Ft	129 000 Ft	egyedi ajánlat
Belépési díj	nincs	nincs	nincs	nincs	nincs
Forgalmi díj	nincs	nincs	nincs	nincs	nincs
IP-tartomány (db IP)	max. 16	max. 16	max. 32	max. 32	megegyezés szerint

5. táblázat Az Irisz Kft. bérelt vonali szolgáltatása

Havi díjak szerződéstípus szerint	64 k	128 k	256 k	512 k
Határozatlan idejű	79 000 Ft	123 000 Ft	257 000 Ft	484 000 Ft
Egyéves	72 000 Ft	112 000 Ft	234 000 Ft	440 000 Ft
Kétéves	64 000 Ft	100 000 Ft	208 000 Ft	392 000 Ft

6. táblázat Az Irisz Kft. WaveNet szolgáltatásának havidíjai

	64 k	128 k	256 k
Határozatlan idejű	89 000 Ft	128 000 Ft	256 000 Ft
Egy éves időtartam	75 000 Ft	118 000 Ft	244 000 Ft
Két éves időtartam	64 000 Ft	112 000 Ft	234 000 Ft

7. táblázat Az Irisz Kft. WaveFlex díjai

Időtartam	64 k	128 k	256 k	512 k
• Egy nap	80 000 Ft	100 000 Ft	130 000 Ft	170 000 Ft
• További napok	40 000 Ft	50 000 Ft	60 000 Ft	75 000 Ft
• Egy hét	200 000 Ft	250 000 Ft	300 000 Ft	375 000 Ft

Tájékozódjunk a szolgáltatók honlapján vagy ügyfélszolgálatán a szolgáltatási területekről! Egy 20-40 ezer fős városban általában már jó esélyünk van arra, hogy valamelyik szolgáltató rendelkezzen elérési alközponttal. A szolgáltatás ára több részből tevődik össze. Az első a vonal kiépítésének költsége, mely a megrendelő és a szolgáltató telephelyének távolságától és egyéb műszaki feltételektől függ.

A második a belépési díj, melynek fejében csatlakozhatunk a szolgáltató hálózatához. Korábban a bérelt vonali szolgáltatásért rendszerint havi alapdíjat és forgalmi díjat kértek, manapság azonban egyre inkább terjedőben van a havi átalánydíj, melynél magasabb havi díj ellenében megszabadulhatunk a forgalmi díj fizetésének terhetől. Végül, de nem utolsósorban a távközlési cégnek fizetendő alaphálózati szolgáltatói díjat kell megemlítenünk, melyet az internet-szolgáltatók általában elfelejtettek a hirdeteikben megemlíteni. Néhány cég már olyan csomagokat is árul, melyek árába ezt a tételt is beépítették. Az alábbiakban néhány piacvezető bérelt vonali szolgáltatást nyújtó cég ajánlatai között válogatok, illetve a kiegészítő szolgáltatásokra is kitérek (mindenhol a nettó árakat tüntettem fel). A korábban MatávNetként ismert szolgáltató immár Axelero néven folytatja működését. A névváltás komolyabb átalakulást is magával hozott: a cég nem hirdeti forgalomfüggő bérelt vonali csomagjait, a hangsúly a továbbiakban a korlátlan bérelt vonali szolgáltatá-

sokra helyeződött át (1. táblázat). A szolgáltatáshoz fix IP-címtartomány, egy .hu tartománynév, valamint elsődleges és másodlagos tartománynév-kezelés is tartozik. A csomagok árait 2001. augusztus 31-éig történő megrendelés esetén szavatolják. A GTS-DataNet legfrissebb akciója keretében egy 64 kbites vonalat a távközlési díjjal együtt összesen 77 ezer forintos havidíjért vásárolhatunk meg, s ehhez még egy Cisco 805-ös átjárót és tartománynév-szolgáltatást kapunk IP-címekkel. Mindössze kétezer forinttal többért, 79 ezerért pedig 128 kbites vonalat kínálnak kétszoros VoIP szolgáltatással. A többi csomag árait a 2. táblázatban találjuk. A Telnét Magyarország Rt. is értékesít átalánydíjas és forgalomfüggő csomagokat. Az átalánydíjas bérelt vonali szolgáltatásokhoz számos ingyenes kiegészítő szolgáltatás tartozik: átjáróbérlet, korlátlan telefonos előfizetés, csoportos levél, tartománynév-bejegyzés és karbantartás, 20 MB tárhely honlapok számára, valamint 16 IP-cím. A 64 kbites csomag ára távközlési díjjal, kiépítéssel együtt 119 000 forint egyszeri és ugyanekkora havi díj, míg a 128 kbites vonalé 199 000 egyszeri és havi díj. Nagyobb sávszélesség-igény esetén egyedi ajánlatot készítenek. A Tvnet Kft. csak forgalomfüggetlen csomagokat értékesít. Náluk 4 Mbites sávszélességet is rendelhetünk, s az árak az alaphálózati szolgáltatói díjakat is tartalmazzák. Minden csomaghoz jár átjáró, egy .hu tartománynév bejegyzése és karbantartása, elsődleges DNS szolgáltatás, postafiókok, illetve 4-16 IP-cím. Az árakat a 4. táblázatban találjuk.

8. táblázat A Tvnet Kft. ADSL szolgáltatásának

Szerződés	DSL 384/64 (magánszemélyeknek)			DSL 384/64 (üzleti)	
	Határozatlan	Egyéves	Kétéves	Határozatlan	Egyéves
Havidíj	17 250 Ft	13 500 Ft	10 400 Ft	83 000 Ft	72 000 Ft
Letöltés	384 kbit/mp	384 kbit/mp	384 kbit/mp	384 kbit/mp	384 kbit
Feltöltés	64 kbit/mp	64 kbit/mp	64 kbit/mp	64 kbit/sec	64 kbit
Levélcím	5 db	5 db	5 db	5 db	5 db
Tárterület	10 MB	10 MB	10 MB	140 MB	10 db
.hu tartománynév bejegyzés	nincs			1 db	1 db

A FlexCom vonalakon szolgáltató Irisz Kft. kínálatában (5. táblázat) is csak korlátlan csomagokat találunk. Minden bérelt vonali szolgáltatáshoz állandó IP-cím és .hu tartománynév tartozik, de vásárolhatunk náluk Linux-alapú, többfeladatos (tűzfal, levél, www, ftp, dns, dhcp stb.) kiszolgálót is.

A bérelt vonal kiépítésének azonban számos műszaki akadálya lehet (a szolgáltató legközelebbi csomópontjától való nagy távolság, nem megfelelő domborzati viszonyok stb.). Ilyen esetben elgondolkodhatunk a mikrohullámú kapcsolat kiépítésén is. Ennek nagy előnye, hogy megtakaríthatjuk vele a távközlési díjakat, hiszen az adatok nem vezetékek, hanem a mikrohullámú rádiójelek segítségével kerülnek továbbításra. A fent említett cégek közül több is vállalja ezen rendszerek kiépítését és működtetését, például az Irisz Kft. is.

A WaveNet szolgáltatás (6. táblázat) jelenleg Budapest területén vehető igénybe, amennyiben a telepítés műszaki feltételei (pl. rálátás az Írisz bázisállomáira) adottak. A szükséges mikrohullámú berendezést a szerződés időtartama alatt az Irisz Kft. díjmentesen biztosítja, megvásárlására nincs szükség. A cég határozatlan idejű, illetve egyéves szerződés esetén 35 000 forint telepítési költséget számít fel. Minden csomaghoz állandó IP-cím és .hu tartománynév is jár.

A WaveFlex (7. táblázat) alkalmi internet-szolgáltatás, különféle rendezvények, kiállítások, koncertek, tanácskozások, sajtótájékoztatók nagy sávszélességű ellátására bárhol Budapest területén. A szolgáltatás telepítését akár a megrendeléstől számított napon belül elvégzik. Természetesen nem minden üzleti ügyfél számára a bérelt vonal az igazi megoldás. Nem mindig van szükség kiszolgáló üzemeltetésére a cég telephelyén, sokszor csupán a szolgáltató kiszolgálótermében elhelyezett gépet kellene gyorsan, hatékonyan és takarékosan elérni. Erre kiváló megoldást jelent az ADSL.

Korábban csak a MatávNet kínált hasonló szolgáltatást, mára azonban több más cég is képes nagy sebességű telefonhálózati hozzáférést nyújtani. Az egyik ilyen a Tvnét, mely magánszemélyek és vállalkozások számára egyaránt kínál megoldásokat. A szolgáltatás (8. táblázat) jelenleg Budapest legnagyobb részén elérhető, hiszen a monopolhelyzetben lévő Matáv Rt. egyelőre itt képes ADSL-csatlakozást kiépíteni. Határozatlan idejű szerződés esetén a Tvnét 45 000 forint belépési díjat kér, ez legalább egyéves szerződéssel 22 500-ra csökkenthető. A legkisebb, 384/64 kbit-es csomagból kétféle létezik. Az elsőt magánszemélyeknek ajánlják: nem tartozik hozzá állandó IP-cím, és csak egy számítógép csatlakoztatását teszi lehetővé. Az összes többi csomaghoz állandó IP is jár, valamint hálózatokat is csatlakoztathatunk. Sőt, tízezer forint havidíjtöbbletért IP-címtartományt is igényelhetünk.

A valamivel drágább VIP csomagok ugyanilyen sebességadatokkal rendelkeznek, de itt a szolgáltató további kötelezettséget is vállal a kapcsolat minőségére (kisebb terhelés és adatátviteli késedelem). A VIP előfizetők egy hagyományos kapcsolt vonali előfizetést

**9. táblázat Az Axelero üzleti ADSL szolgáltatása**

<i>Egy éves szerződés esetén</i>	<i>egyszeri díj</i>	<i>havidíj</i>
BDSL 384	30 000 Ft	49 900 Ft
BDSL 768	45 000 Ft	89 900 Ft
BDSL 1500	60 000 Ft	169 900 Ft
<i>Három éves szerződés esetén</i>	<i>egyszeri díj</i>	<i>havidíj</i>
BDSL 384	30 000 Ft	39 900 Ft
BDSL 768	45 000 Ft	69 900 Ft
BDSL 1500	60 000 Ft	129 900 Ft

kapnak tartalékba, és kedvezményesen bérelhetnek tűzfalat, átjárót. Az Axelero BDSL névvel illeti új, üzleti ADSL csomagjait (9. táblázat). A csomagokhoz számos kiegészítő (ingyenes vagy fizetős) szolgáltatás jár: egy ethernetet keresztül PPP-t megvalósító átjáró (melyhez négy hálózati kapcsoló vagy jelelosztó közbeiktatásával tíz számítógép csatlakoztatható), Linux vagy Windows 2000 operációs rendszer, 15 levélcím, 40 MB webtárhely, egyedi honlaptervező-rendszer CGI/PHP/ASP-támogatással, elsődleges és másodlagos DNS-szolgáltatás.

Az ADSL-alapú internet-hozzáférés vidéken jelenleg nem érhető el. Több cég is tervezi a vidéki terjeszkedést, ez azonban eddig nem következett be (általában műszaki akadályokra hivatkoznak). Magyarországon több kábeltéves internet-szolgáltató is működik, de nem mindegyik értékesít üzleti csomagokat.

A Tvnétnél egyépes előfizetést is köthetnek az üzleti megrendelők 48 000 forint belépési díjért és tízezer forint havidíjért, de ebben az esetben hálózat nem csatlakoztatható a végpontra. A cégek számára így valószínűleg a többgépes előfizetés a megfelelőbb, melynek egyik fajtája egy IP-címmel (10. táblázat), a másik saját IP-címtartománnyal (11. táblázat) rendelkezik. Az utóbbi csomagokhoz szükséges RIPE engedélyeket a Tvnét díjmentesen beszerzi. Az 5–13 gépes csomagtól fölfelé az előfizető a kapcsolt ethernetszakasz (pl.: kábelmodem) kizárólagos használója, tehát más felhasználók nem lassíthatják a kapcsolat sebességét. A 14–30 gépes csomagtól fölfelé minden előfizetéshez elsődleges DNS-szolgáltatás és .hu tartománynév bejegyzése is jár.

Mindkét szolgáltatás rendelhető VIP csomaggal is, melynek lényege, hogy 25 000 forint havidíjtöbbletért megbízhatóbb működést és valóban gyors külföldi kapcsolatot kapunk.

Az Axelero kábeltéves internetszolgáltatása jelenleg Budapest XVI. kerületében (Sashalom) és Budapest II. kerületében (Pesthidegkút) a MatávkábelTV Kft. hálózatán, illetve Kaposváron a Kaposkábel Kft. hálózatán érhető el. Az Üzlet nevű csomaggal 320/64 kbit/mp-es le- és feltöltési sebességet érhetünk el, kapunk hozzá 8 levélcímet, 100 MB webtárhelyet, egy állandó IP-címet, valamint tartalékba évi tíz óra kapcsolatot lehetővé tevő hagyományos modemes előfizetést. A csatlakozási díj 25 000, a havidíj 31 250 forint. Nagyobb sávszélesség vagy más igények esetén egyedi ajánlatot készítenek. A UPC Magyarország Kft. a legnagyobb kábeltéves internetszolgáltató Magyarországon, s egyben az egyetlen olyan cég, amely a vidéki terjeszkedést is a legfőbb célkitűzései között tartja számon. Az ország számos pontján (Budapest, Miskolc, Nyíregyháza, Debrecen, Szolnok stb.) elérhető valamelyik szolgáltatásuk (Broadband vagy Chello), azonban a vállalkozások számára jelenleg nem kínálnak csomagokat. Többször kinyilatkoztatták, hogy tervezik az üzleti csomagok megjelentetését, ez azonban a mai napig nem következett be.

A többi vidéki – egy-egy városban (pl. Szegeden) működő –

© Kiskapu Kft. Minden jog fenntartva

**havidíjai**

<i>DSL 768/128 (üzleti)</i>		<i>DSL 1500/384 (üzleti)</i>	
<i>Határozatlan</i>	<i>Egyéves</i>	<i>Határozatlan</i>	<i>Egyéves</i>
182 000 Ft	165 000 Ft	318 000 Ft	287 500 Ft
768 kbit	768 kbit	1,5Mbit	1,5Mbit
128 kbit	128 kbit	384 kbiz	384 kbit
10 db	10 db	10 db	10 db
20 MB	20 MB	30 MB	30 MB
1	1	1	1

10. táblázat A Tvnet Kft. kábeltéves internetszolgáltatása üzleti megrendelők számára

Csatlakoztatott számítógépek száma:	3-4	5-13	14-30	31-62
Belépési díj (egyszeri)	120 000	200 000	280 000	350 000
Belépési díj részletfizetéssel (12 hónapra)	12 000/hó	20 000/hó	28 000/hó	35 000/hó
Havi előfizetési díj	30 000	50 000	80 000	120 000
Webtárhely	5 MB	5 MB	5 MB	5 MB
Saját IP-cím	1	1	1	1
Levélcím (2 MB/postafiók)	30 db	50 db	80 db	120 db

11. táblázat A Tvnet Kft. üzleti kábelnet-csomagjai IP-címtartománnyal

Csatlakoztatott számítógépek száma:	3-4	5-13	14-30	31-62
Belépési díj (egyszeri)	120 000	200 000	280 000	350 000
Belépési díj részletfizetéssel (12 hónapra)	12 000/hó	20 000/hó	28 000/hó	35 000/hó
Havi előfizetési díj	40 000	60 000	90 000	130 000
Webtárhely	5 MB	5 MB	5 MB	5 MB
Saját IP-cím	1+8	1+16	1+32	1+64
Levélcím (2 MB/postafiók)	30 db	50 db	80 db	120 db

kábelszolgáltató egyelőre tartja magát a vetélytársaival szemben, azonban az üzleti megrendelők számára ezen cégek ajánlatai sem jelentenek semmit. A vidéki kábeltéves internet-hozzáférés elterjedésére és „felnőtté válására” még egy pár évet várnunk kell. Amint láthatjuk, azért van miből választani. Cikkem végén most ehhez szeretnék egy kis segítséget nyújtani.

A bérelt vonal valóban az egyik legnépszerűbb „széles sávú” megoldás cégek számára. Megbízható kapcsolatot és gyors hibaelhárítást kapunk a pénzünkért. Nyilván feltűnt két mondattal előbb az idézőjel. Nem végeztem kutatást ez irányban, de valószínűnek tartom, hogy a magyarországi vállalkozások legnagyobb részénél a 64-128, esetleg 256 kbit/mp-es bérelt vonalra mutatkozik a legnagyobb igény, hiszen ennél nagyobb összeget nem tudnak az intézmény, telephely, iroda internetkapcsolatára fordítani. A nagy döntés előtt tehát tegyük fel magunknak a következő fontos kérdéseket:

1. Szükség van-e arra, hogy a vállalkozás telephelyén nyilvános webkiszolgálót üzemeltessünk?

A már itthon is megjelent internet data centereknek köszönhetően könnyen, költségtakarékosan és hatékonyan megoldható saját honlap, vagy webbolt elindítása, üzemeltetése. Az internet data centerek korszerű, biztonságos hátteret nyújtanak saját kiszolgáló elhelyezésére, illetve bérlésére.

Ez általában azt jelenti, hogy a szolgáltató kifejezetten erre a célra szolgáló termében elhelyeznek egy általunk megvásárolt és mindvégig a tulajdonunkban maradó gépet, melyen a webkiszolgáló és az összes ehhez kapcsolódó alkalmazás futhat. A kiszolgálótermek legtöbbször nagyon gyors kapcsolattal rendelkeznek a külvilág (belföldön a BIX, illetve a külföldi vonalak) felé. Az egyes kiszolgálókra jutó sávszélesség ez általában 10, 100 vagy 1000 Mbit/mp-esek, e különbségek a szolgáltatás árában is jelentkeznek.

Ilyen esetben járjunk utána, hogy telephelyünk internet-csatlakozásának kiépítésére milyen más lehetőségek vannak. Tájékozódjunk a szolgáltatók honlapján arról, hogy az adott területen elérhető-e az általunk választott szolgáltatás. Anyagi szempontból mindenképpen jobban járunk kábeltéves hálózaton, vagy ADSL-en működő megoldással, mint a bérelt vonallal. Már a kisebb

csomagokhoz is legalább 64-128 kbit/mp-es feltöltési sebesség tartozik, mely legtöbbször elegendő a kiszolgálóteremben lévő gép karbantartására, a legújabb adatok feltöltésére – ugyanakkor ezek ára nagyságrenddel kisebb a hasonló sebességű bérelt vonalakénál. Emellett természetesen nagy letöltési sebességet is biztosítanak.

2. Szükség van-e a bérelt vonal által kínált többletszolgáltatásokra (nagyobb biztonság, megbízhatóbb működés, állandó IP stb.)?

Ezek valóban megfontolandó tényezők, azonban ne felejtsük el, hogy több szolgáltató kínál a kábeltéves és ADSL csomagokhoz is állandó IP-címet és más többletszolgáltatásokat (gondolok itt például a Tvnet VIP csomagjaira)! A helyi hálózatokat kábeltéves vagy ADSL előfizetés esetén is mindenképpen átjáróval, tűzfalal célszerű védenünk az esetleges támadások ellen. A Linux erre a feladatra is kiválóan alkalmas.

## Mi várható a jövőben?

A bérelt vonali szolgáltatások árai ma is a nyugati árak környékén mozognak, előre lépésre tehát valószínűleg nem e területen számíthatunk. Az igazán nagy sávszélességű

bérelt vonali kapcsolat minden bizonnyal a jövőben sem a kis- és közepes méretű vállalkozások számára jelenti majd a megfelelő választást. Az ADSL vidéki térhódítása nemcsak a magánszemélyek, de a vállalkozások érdeke is, és nagyjából ugyanez mondható el a kábeltéves hozzáférésről. Az üzleti élet – és hangsúlyozottan, a vidéki vállalkozások – fellendítésében komoly szerepet játszhat az Internet, s e folyamat felgyorsításában a távközlési piacon oly régóta várt egészséges verseny kialakulása hozza majd a legnagyobb változást.



Borai János (borai.janos@linuxvilag.hu) az ELTE amerikanisztika szakos hallgatója, 1997-ben ismerkedett meg a Linuxszal. Szabadidejében zenél, jelenleg egy otthoni stúdió kiépítésén fáradozik.

## Kapcsolódó címek

### ISDN, ADSL

- ➔ <http://www.isdn.matav.hu/>
- ➔ <http://www.matav.hu/>
- ➔ <http://www.matavnet.hu/>
- ➔ <http://www.datanet.hu/>

### Kábelnet

- ➔ <http://www.tvnet.hu/>
- ➔ <http://www.broadband.hu/>
- ➔ <http://www.chello.hu/>
- ➔ <http://www.matavkabel.hu/>

### Műholdas kapcsolat

- ➔ <http://www.europeonline.hu/>
- ➔ <http://www.infotechna.hu/>
- ➔ <http://www.weyland-yutani.hu/>

## GIMP a gyakorlatban (3. rész)

Tovább építjük weboldalunkat, és a GIMP segítségével létrehozzuk az Aqua stílusú gombokat, egyedi logót is tervezünk.

**A** cikkek során szép lassan elkészül a weboldal, és remélem a GIMP használatában is sikerül elmélyedni. Megpróbálom olvasóinkat példákon keresztül rávezetni, hogyan lehet egyedi grafikákat szerkeszteni. Ez sokszor csak az első látásra tűnik megoldhatatlan feladatnak. Ha jobban megnézzük a lehetőségeket, szinte mindig találunk több módot a megvalósításra. A megoldás kulcsa – mint más területeken is – a feladat kisebb részekre bontása. Tagoljuk olyan kisebb részekre a feladatot, melyeket már könnyedén megoldhatunk a GIMP segítségével! Az ilyen kis részekből fog összeállni a kép. A cikkben levő grafikák elkészítése első látásra nehéznek tűnik, de meg fogjuk látni, hogy milyen egyszerűen készíthetünk látványos képeket Linuxon.

A cikk következő részében elkészítjük az Aqua stílusú gombot. Ehhez először szükségünk lesz lekerekített terület létrehozására, és csíkos háttér készítésére. Elsőként nézzük meg, hogyan alkothatunk egyszerűen lekerekített gombokat kedvenc grafikai programunk segítségével.

### Lekerekített keretek készítése

Számos grafikai feladathoz és a weboldalakhoz szükségünk lesz lekerekített keretekre, gombokra. A megoldás egyszerűbb, mint gondolnánk:



1. kép A gomb mérete lekerekítés nélkül

1. Először készítsünk képet fekete alapon a kívánt méretben. A mintaképet 800x300 képpont méretben készítettem el, ez természetesen nagyobb méret, mint amire szükségünk lesz. A képeket, gombokat, rajzokat ajánlatos a felhasználó méretben készíteni, mivel az átméretezés során a kép veszít a minőségéből. Jelöljük ki a keretünk számára szükséges területet (1. kép)!
2. A következő lépésben a kívánt lekerekítés mértékében alkalmazzuk a

Gauss-elmosást (Szűrők->Elmosás->Gauss-elmosás IIR). Minél nagyobb lekerekítést szeretnénk, az elmosást annál nagyobb értékre állítjuk. A kép nagy mérete miatt az elmosás értékét százra állítottam. A Weben használatos gomb elmosásánál 10–30 közötti érték a megfelelő, ezt természetesen az elérni kívánt hatás függvényében adjuk meg. Rövid használat után már mindenki rá fog érezni, hogy ötletei megvalósításához milyen mértékű elmosást kell alkalmaznia (2. kép).



2. kép Az elmosás utáni állapot, már jól látszik a lekerekítés



3. kép A lekerekített gomb alapja

3. Nincs más hátra, mint az elmosott képből éles körvonalú, legömbölyített gombot készíteni. Ezt a szintet (Kép->Colors->Szintek) állításával tudjuk megtenni. A bemeneti értékeket (Input Levels) állítsuk 120, 1.0, 130-ra! Már is láthatjuk a kívánt végeredményt a 3-as képen.
4. A tökéletes hatás eléréséhez a területet pontosan határoljuk körül. Jelöljük ki a fehér területet az Egybefüggő területek kijelölése eszközzel! Invertáljuk a kijelölést (Kijelölés->Invertálás), és színezzük ki feketével (Kijelölés->Feltöltés az előtérszínnel)!

Ugye nem is volt olyan nehéz? A legömbölyítésre nem csak gombok, feliratok készítésnél lesz szükségünk, használhatjuk maszkolásra, élek finomítására, területek világosítására stb.

Miután elkészítettük a legömbölyített gombot, ha tetszik akár használhatjuk ebben a formában is. Az előző cikkben ismertetett módon írhatunk bele szöveget.

A kívánt eredmény eléréséhez használhatunk szűrőket, Script-Fu függvényeket (például árnyék készítése – drop shadow). Az általam elkészített egyszerű gombot láthatjuk a 4. képen.



4. kép Az egyszerű lekerekített gomb

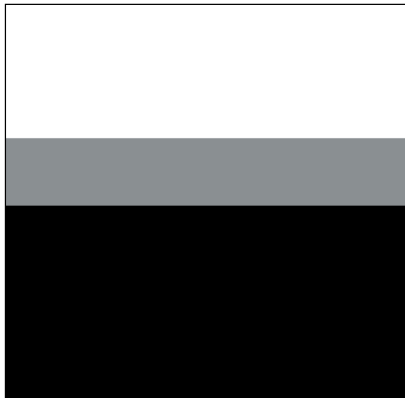
Ha nem elégedünk meg az egyszerű, lekerekített feliratok készítésével, az Aqua stílusú gomb megalkotásához szükségünk lesz csíkos kitöltő mintára. Mivel csíkos kitöltő mintát a GIMP-ben nem találunk, ezért nekünk kell elkészíteni. A legegyszerűbb és legtöbb célra felhasználható megoldás szerintem a kitöltő minta.

### Kitöltő minta készítése

Egyedi hátterek készítéséhez, betűk, grafikai elemek mintával való kitöltéséhez használhatjuk a programmal adott kitöltő mintákat, de magunk is készíthetünk kedvünk szerint. A kitöltő minták kisméretű képek, amivel a GIMP a kívánt területet feltölti, „ismétli” a képeket. A kitöltő minta készítésekor általában ügyelnünk kell arra, hogy a mintánk jól „ismételhető” legyen.

Készítsünk csíkos kitöltő mintát! Ezt a mintát nemcsak gombunk háttéréhez, hanem például tévészerű hatásként is lehet használni.

1. Mivel a csíkos hatás eléréséhez nem szükséges nagy területet ismételni, ezért kis méretű kép is megteszi. Hozzunk létre egy 6x6 képpont méretű képet, átlátszó háttérrel, a kép típusa RGB legyen!
2. Jelöljük ki a felső két sort, ezt töltsük ki fehérrel! Az alatta levő sort töltsük fel harminc százalék szürkével (lásd az 5. képen). A szürke kitöltés ad majd némi térbeli hatást.
3. Mentsük el a mintánkat a saját könyvtárunkban található .gimp-1.2/patterns könyvtárba csikoshatter.pat néven, majd a mentéskor a párbeszédablakban a képünknek adjuk a „Csíkos háttér” nevet! A GIMP két helyen tárolja a



5. kép A csíkos kitöltő minta, mentés előtt

kitöltő mintákat, a közös minták a telepítés során a /usr/X11/share/gimp könyvtárba kerülnek, ezekhez csak olvasási jogunk van. Az első indítás után a program a felhasználó saját könyvtárában létrehoz egy könyvtárat .gimp-1.2 néven. A felhasználó ide minden korlátozás nélkül bemásolhatja az elkészített kiegészítéseket. Rövid felsorolás a könyvtárakról:

- Brushes – ecsetek
- Curves – a Colors->Görbék menüben készített és elmentett beállítások
- Fractalexplorer – a fraktálkészítőhöz alkotott saját kiegészítések
- Gfig, gflare, gimprc, gimpressionist – függvényekhez írt kiegészítések
- Gradients – saját színátmenetek (az előző részben elkészített „Piros-Fehér-Zöld” színátmenet is ide kerül)
- Levels – a Colors->Szintek menüben készített és elmentett szintbeállítások
- Modules – kiegészítő modulok (Kit->Modules)
- Palettes – színpaletták
- Patterns – kitöltő minták (ezt használtuk most)
- Plug-ins, scripts – saját szűrők és parancsfájlok.

A .gimp-1.2 könyvtárban található a program beállításait tartalmazó rc fájlokat. Ezek szerkesztésére nincs szükségünk. A könyvtárak nagy részének kezelése a GIMP programból önműködően történik.

Próbáljuk is ki gyorsan az új alkotásunkat! Nyissuk meg egyik kedvenc képünket, hozzunk létre egy új réteget, és jelöljük ki az egész képet! Ha a mintaböngészőben nem látjuk az új elemeket, a frissítés gombbal vehetjük rá azok beolvasására. Válasszuk ki az új, Csíkos háttér nevű képet, és töltsük ki az új rétegünket! Ne ijedjünk meg, az egész kép kis fehér csíkokból fog állni! A réteg módját állítsuk *Kivetítés* (Screen)



6. kép Használatban az új kitöltő mintánk

típusúra, a láthatóság értékét pedig húszra! Ha a csíkosságot szeretnénk jobban hangsúlyozni, játsszunk kicsit a láthatóság értékével. A kitöltő minta használatával kicsit később még foglalkozunk (6. kép).

### Aqua stílusú gomb készítése

A következő részben elkészítjük gombjainkat, ezek kinézetét az egyik legszebb felületű ablakkezelőtől kölcsönözzük. Remélem, hogy hamarosan az Apple mellett PC-ken is használhatjuk ezt a gyönyörű ablakkezelőt (☞ [osxonintel.com](http://osxonintel.com)). Még mielőtt elmélyednénk az operációs rendszerek és géptípusok közötti különbségek vizsgálatában, készítsük el a gombot! Most már rendelkezünk minden olyan ismerettel, ami szükséges a sikeres munkához.

1. Először is hozzuk létre a kívánt méretű képet! A cikk elején bemutatottak alapján készítsünk egy fehér lekerekített gombot, fekete háttérrel. Nevezzük el a réteget „alapnak”! Induljunk onnan, ahol a lekerekített gomb leírásában befejeztük (7. kép).



7. kép Az ismert lekerekített gomb

2. Hozzunk létre egy új réteget, töltsük ki bordó színnel, majd egy újabb réteget töltsük ki a „Csíkos háttér” mintával! Alkalmazzuk az ismertetett „kivetítés” (screen) módot és a láthatóság értékét állítsuk húszra!
3. Hozzunk létre maszkot ezen a rétegen, jelöljük ki a képet, és az alapréteget másoljuk a vágólapra, majd illesszük be, és a „réteg rögzítése” szolgáltatással adjuk hozzá a maszkhoz a korábban kimásolt képet! A maszkok használatával későbbi részben fogunk foglalkozni, addig is rövid összefoglalás: a maszk esetében a fekete szín az átlátszóságot, a fehér a „színt” jelenti, tehát ahol fekete színű a réteg,

- ott átlátszik a kép. Nagyon jól használhatjuk ezt a szolgáltatást kitakarásokhoz.
4. Most a gomb alsó részének világítását fogjuk elkészíteni. Hozzunk létre egy új réteget, töltsük ki színárnyalattal a 9. képen látható módon! Alkalmazzuk az előbb megismert maszkolást erre a rétegre is. A réteg módját válasszuk „kivilágításra”! A kitöltés fehér részének a mérete adja majd a gomb „magasságát”, minél kisebb a fehér terület, a gomb annál magasabbnak tűnik.

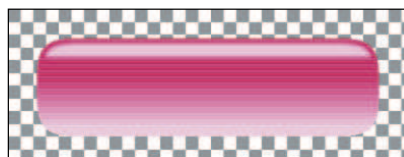


8. kép Már alakul a gomb



9. kép A világosításra használt színátmenet

5. Majdnem készen is vagyunk a gombbal. Most készítsük el a gomb felső részének világosítását! Válasszuk ki az alapréteget, jelöljük ki a fehér területet, majd csökkentjük a kijelölést a *Kijelölés->Csökkentés* pont segítségével – a 800 képpont széles képen én tíz képponttal csökkenttem. Ezek után a négyszög kijelöléssel a CTRL gomb nyomva tartása mellett csökkentjük a kijelölést a gomb felső öt-tíz százalékára! A kijelölt területet



10. kép A gomb felső részének világosítása



11. kép Az elkészült gomb felirattal

- töltsük ki fehér színnel, majd Gauss-elmosással kissé homályosítsuk el! A réteg módját állítsuk „kivetítésre”, láthatóságát pedig 80-ra (lásd 10. kép).
6. Már csupán a gomb végső csiszolása maradt hátra. Írjuk a szöveget a gombra, finomítsunk rajta tetszés szerint! Ha szeretnénk, készítsünk neki vékony

## GREEN LINUX

12. kép A kitöltések sok mindenre használhatók

keretet. Jelöljük ki az alaprétegen a gomb területét, növeljük a kijelölést, annyira amilyen vastagságú keretet szeretnénk! Hozunk létre egy új réteget, majd a kijelölésünk szerint töltjük fel fekete színnel! Az erősségét csökkenthetjük a láthatóság tulajdonsággal. Kísérletezzünk a leírás alapján az új megjelenés egyszerű elkészítésével! A kész gomb a 11. képen látható. Ennek a gombnak az elkészítésére több más lehetőség is van. Mindenkitől örömmel fogadok további ötleteket!

### Készítsünk természetbarát feliratot

Próbáljuk meg a kitöltő minták használatát más megközelítésben is. Alkalmazzuk ismét a cikk első részéből ismert gombot. Most kicsit részletesebben más megköze-



13. kép A Green Linux felirat háttérrel

lítésből fogjuk használni a kitöltő mintákat. Ezeket nemcsak egyszerű szűrőként, hanem hatásnövelőként is használhatjuk. Készíthetünk falevelekkel kitöltött feliratot is, például Green Linux felirattal (lásd 12. kép). Nézzük meg, hogyan!

Jelöljük ki a feliratunkat a *Kijelölés szín szerint* menüponttal *Kijelölés->Kijelölés szín szerint!* A kitöltések közül válasszuk ki a zöld leveleket tartalmazó mintát (leave #2), majd a kitöltés eszközbeállításai *Fájl->Párbeszédablakok->Eszköztárak* ablakban válasszuk ki a *Kitöltés mintával* típust! Természetesen nemcsak szöveget, hanem bármilyen területet, általunk kijelölt képrészletet kitölthetünk ezzel a módszerrel. Készíthetünk egyedi háttereket a kedvenc ablakkezelőnkhez is. A kitöltő minták használatakor ügyeljünk arra, hogy a

részletes minták esetén az „illesztések”, a minta ismétlődése nem mindig tökéletes. Ha nagy, összefüggő területet szeretnénk kitölteni, válasszunk kevésbé részletes mintát. Magunk is készíthetünk kitöltő mintát. A zöld levelekkel kitöltött feliratot felhasználva készítettem egy kis képet, ezt láthatjuk a 13. képen.

Az ilyen feliratokat használhatjuk a weboldalunkhoz, egyedi feliratokhoz, de akár a weboldalunk háttéréül is. A programba épített kitöltő minták változatos lehetőségeket teremtenek, ezeket az igényeink szerint tovább bővíthetjük. Ennek menetét a következő részben fogom bemutatni. Addig is mindenkinek jó kísérletezést és gyakorlást kívánok! A cikkben szereplő mintákhoz használt képek megtalálhatók a [www.gimp.hu/linuxvilag/](http://www.gimp.hu/linuxvilag/) oldalon.



Süveg Gábor  
Régóta használ Linuxot és BSD-t. Hobbija a bűvárkodás, vitorlázás és a számítógépes grafika. Elérhető a [gsuveg@sgmobil2000.hu](mailto:gsuveg@sgmobil2000.hu) címen.

© Kiskapu Kft. Minden jog fenntartva

## Szerkesztők háborúja II. – A Pico szerkesztő

Nagyon sokan kedvelik a karakteres alapú Pine levelezőprogramot, hiszen könnyen használható, okos és gyors is. A program szerkesztő része külön programként is elérhető, ez a *pico*. A szerkesztő nagy előnye, hogy ugyanazok a parancsok használhatók, amik a levelezőben, így ha valaki megtanulta azt a néhány szolgáltatást, amivel könnyen összehozza leveleit, akkor már nincs elveszve.

De nézzünk egy kicsit a *pico* körmére! Indítása a szokásos, kezelése is nagyon hasonlít például a *joe* használatára. A hosszú szövegben szintén a CTRL+V-vel tudunk lapozni, a sor elejére és végére ugyancsak a CTRL+A és a Ctrl+E kombinációkkal juthatunk. Ha a HOME, END és hasonozórú billentyűk nem működnek, ezekkel jelentősen gyorsíthatjuk munkánkat. Nagyon hasznos lehet még, hogy a program ismeri a különböző kétpaneles fájlkezelőkben (például a Midnight Commander) szokásos Fn billentyűket is. Ha indításkor a -f parancssori kapcsolót is megadjuk, akkor az alsó két sorban is ezeket a billentyűket láthatjuk (lásd a képen).

A fájlműveletek területén a program két fő paranccsal rendelkezik: CTRL+R (F5): fájl tartalmának beszurása (read) és CTRL+O (F3): fájl mentése (output). A kilépésre a CTRL+X-et (vagy az F2-t) használhatjuk, amikor is a program megkérdezi, hogy mentse-e a változásokat.

A blokkműveletek területén nem túlzottan jeleskedik a drága. Röviden összefoglalva tud kivágni és beilleszteni. Le tudunk rakni egy jelet a CTRL+^ billentyűvel (magyar

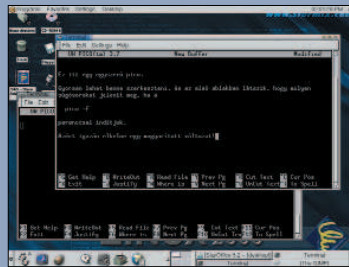
kiosztásnál CTRL+ALTGR+3, angolnál CTRL+SHIFT+6), innen től a jel és a kurzor közötti részt jelöljük ki. A kijelölt részt azután egy CTRL+K-val kivágjuk. Ha nincs jelölés, akkor az adott sort vágja ki a CTRL+K (vagy az F9). Kényelmes megoldás, hogy az egymás után kivágott sorok egy blokknak számítanak, tehát a háromszori CTRL+K ugyanazt eredményezi, mintha kijelöljük a három sort, majd kivágjuk. A kívánt helyen pedig egy CTRL+U (F10) segítségével már be is illeszthetjük a kivágott részt.

Egy egyszerű keresője van, ami a kis- és nagybetűk különbségére nem érzékeny, viszont a magyar ékezetes betűkre igen. Nos, nem tökéletes, de legalább gyors. A keresőt a CTRL+W-vel (F6) érhetjük el.

Az utolsó szolgáltatás, amit érdemesnek tartok megemlíteni, igen hasznos, főleg, ha többször átírunk egy bekezdést. Egy idő után a négy-öt sor össze-vissza hosszúságú lesz, és tördelhetjük újra kézzel. Erre megoldás a CTRL+J (F4), mely újrendezi a bekezdés szavait a margó éppen érvényben lévő értéke szerint (a margót a -r-n parancssori kapcsolóval állíthatjuk be). Ha egyáltalán nem szeretnénk, hogy a szerkesztő magától tördelje a sorokat, kikapcsolhatjuk ezt a szolgáltatást a -w parancssori kapcsolóval.

A szerkesztő rendelkezik saját helyesírás-ellenőrzővel, de külső programot is tud használni. Ebben az esetben a SPELL környezeti változóban kell megadnunk a használni kívánt programot (ennek a szabványos bemenetről kell olvasnia, és a szabványos kimenetre kell írnia).

A Pico magyar súgóoldala *Hercegkuti László* munkája.





## A nyílt forrású MPEG

Az MPEG egyik szülőatyja elmeséli a Moving Picture Experts Group történetét és beszámol a nyílt forrású fejlesztésekről is.

**N**éhány évszázaddal ezelőtt az Alpok lábánál, Torinóban élő őseim rájöttek: ha a hegyekben kanyargó hágók útjait zúzalékkővel fedik be, akkor mindenki „jobban jár”.

Soha nem derült ki, hogy ez hihetetlen szorgalmukról ad tanúbizonyságot, vagy inkább egyszerűen csak az történt, hogy a helyi hatóságok munkára kötelezték a télvíz idején mezőgazdasági munka nélkül tengődő hegyi telepeseket. Végül is a gazdák akkoriban nem arról voltak híresek, hogy szívesen megosztották volna készleteiket másokkal, és azt se feledjük, hogy azok az idők – felvilágosodás ide vagy oda – az önkényuralom évszázadai voltak.

Néhány évvel ezelőtt a számítógépes guruk rájöttek, hogy mindenki számára hasznos, ha a processzorok és az alkatrészek „járhatatlan hágóit” közös erővel kifejlesztett és mindenki számára elérhető operációs rendszerrel kövezik ki.

A felhasználók sokszor olyan adatokkal dolgoztak, amelyekhez eleve digitális formában fértek hozzá, vagy amelyeket nagyon egyszerűen lehetett digitálissá alakítani (gondoljunk csak a szövegekre). Más adat-típusok, például a látvány és a hang, analóg természetüknél fogva sokkal nehezebben digitalizálhatók. S ami még tovább bonyolítja a helyzetet, hogy ezek „széles sávú”, azaz adott időegység alatt viszonylag nagy mennyiségű adat átvitelét igénylő források (a „széles sávú” kifejezés természetesen mindig a módszer éppen időszerű állapotától függ). A hang- és képadatok feldolgozása mindig is fontos szakterület volt, de sokáig csak félmegoldások születtek e téren. A Moving Picture Experts Group, vagy MPEG által kezdeményezett munkának köszönhetően a hangos mozgókép a jelenleg használatos adatsatornákon egyszerűen továbbítható.

Az MPEG-12, az MPEG-23 és az MPEG-44 szabványok fejlesztésével párhuzamosan az MPEG-kiegészítő programokat is készítette, a nyílt forrású programokhoz hasonló elvek alapján (bár a Nyílt Forrású Kód Közössége biztosan találna kivétneivalót a részletekben). Hozzá kell tennünk, hogy a folyamatot az ISO szabványokhoz kellett igazítaniuk, hiszen az MPEG e szervezet keretein belül működik. Cikkemben felidézem a hang és kép digitalizálásának kezdeteit, a Moving Picture Experts Group megalakulásának célját, összefoglalom a ma használatos MPEG szabványok elemeit, majd ismertetem az MPEG nyílt forrású fejlesztés jelenlegi állapotát.

### Hang és kép digitalizálása

A nyomda feltalálása után 400 évnek kellett eltelnie ahhoz, hogy a kapcsolattartás területén hasonló jelentőségű találmányok lássanak napvilágot. Az 1830-as évektől kezdődően az emberiség számára egyre több, a hang és a kép rögzítését, feldolgozását forradalmasító eljárás állt rendelkezésre: a fénykép, a telegráf, a telefon, a telefax, a lemezjátszó, a filmfelvevő, a rádió, a televízió és a mágneses rögzítés. E módszerek egyik fő hátránya, hogy együttes használatuk általában nehézségekbe ütközik. Minden egyes típus esetében önálló készülékre van szükség. Mennyire más ez a számítógépek világában, ahol minden adat ugyanazon elv felhasználásával továbbítható és tárolódik! A képi és a hangadat egyesítésének elméleti alapjait tizenöt évvel az első számítógép megépítése előtt fektették le. Ugyanis ekkor jöttek rá, hogy egy S sávszélességű jelet 2S frekvenciával mintavételezve minőségvesztés nélkül tárolható az adat.

Bár az elméleti kutatásokat végző Bell Laboratórium a tranzisztor feltalálásával megtette az első lépést a gyakorlati felhasználás irányába, ennek ellenére még rengeteg munka maradt. Még az olyan, viszonylag keskeny sávszélességet igénylő adatfolyam is, mint a beszéd, mely a telefonvonalon 0,3–3,4 kHz-et foglal el, 8 kHz-en, nyolc biten mintavételezve az akkor igencsak széles sávnak számító 64 Kbit/másodperces sebességet igényelte.

Tizenöt év kísérletezés után a bitek végre készen álltak arra, hogy szerepet játsszanak az emberi beszéd továbbításában. Az 1960-as években a CCITT (ma ITU-T-nek hívják) elfogadta a beszéd digitális rögzítésének irányelveit. Valójában két szabványt fogadtak el, melyek mindegyike 8 kHz-es rögzítést ír elő: az egyik a hét bites  $\mu$ -law, a másik pedig a nyolc bites A-law. Mindkettő nonlineáris, azért hogy minél tökéletesebben igazodjanak az emberi hallás logaritmikus természetéhez. Az egész gyakorlati haszna azonban nem volt túl jelentős, hiszen csak a telefon gerinchálózatán alkalmazták, ahol a csatornák többszörösét jóval kényelmesebb volt digitálisan, mint analóg módon megoldani. A végfelhasználóknál tehát nem változott semmi. A Group 3 faxrendszere (Gr. 3 fax) sokkal érdekesebb volt. Egy letapogatógépséggel (mely 1728 érzékelőt tartalmazott) nagy felbontású üzemmódban beolvasott A4-es lap átvitele 4 Mb/mp sebességet igényelt. Az akkori idők „nagy sebességű” (9,6 Kb/mp) modemjeivel egyetlen oldal húsz perc alatt jutott célba, azonban egy egyszerű tömörítési eljárás segítségével (az egyszerű fekete és fehér jelzők, valamint a méretek meghatározása helyett változó adatblokkok és „ismétlődés-hosszok” használatával) az átviteli időt sikerült 2,1 percre csökkenteni.

A gerinchálózatokon tehát nagyszerű újítást jelentett a digitális hangtovábbítás, de a végpontok környéke reménytelenül analóg maradt. Az 1980-as években az ISDN megjelenése komolyabb megoldásokat tett szükségessé, 7 kHz-es sávszélességű, 16 kHz-es mintavételezési frekvenciát és a  $\mu$ -lawnál, A-lawnál magasabb (pl.: 14-es) mintánkénti bitszámot használó tömörítési eljárásokat. Erre azért volt szükség, mert egy ekkora adatfolyam átvitele 200 Kb/mp sebességű hálózatot igényelt volna. A négyszeres vagy nagyobb tömörítési aránnyal elérhető legfeljebb 64 Kb/mp-es adatfolyam még megfelelő hangminőséget biztosított volna. Az ekkoriban kifejlesztett eszközök digitális jelfeldolgozókat (Digital Signal Processorokat – DSP-eket) használtak, de soha nem terjedtek el széles körben.

A mozgókép még ennél is nagyobb kihívás elé állította a tervezőket, hiszen itt a hangnál használt sávszélesség többszöröse szükséges, és egyszerre több jelet kell átvinni. A digitális televízió lényege, hogy az Y világosságértéket 13,5 MHz-en, az R-Y és B-Y színkülönbségeket pedig 6,75 MHz-nél, nyolc biten mintavételezzük. Az így kialakuló 216 Mb/mp-es adatfolyamot a nem látható minták eltávolításával körülbelül 166 Mb/mp-re csökkenthetjük. Ekkora sávszélességű adatokat nem lehet a hagyományos hálózatokon továbbítani, így csak a stúdiókban vagy digitális szalagokon használhatók. Először azzal próbálkoztak, hogy a sebességet 1,5-2 Mb/mp-re csökkentik, és így az adatfolyam illeszkedhet az Egyesült Államokban és Európában használt többszörösök csatornaszámához (24, illetve 32). Ez azonban a mai napig hatalmas kihívást jelent, ezért a bemeneti bitsebességet a felére csökkentették, majd alumintavételezték a vízszintes

és függőleges (a váltott soros üzemmód miatt valójában változó) irányokat, illetve a szinkronizációkat. Ezt követően két egyszerű eljárás, a DPCM-et (Different Pulse Code Modulation) és a feltételes kitöltést alkalmazták. A második nemzedékbeli tömörítők (kodekek) összetettebb eljárásokkal dolgoztak, például DCT-vel (digitális koszinusz átalakítás) és mozgáskiegyenlítésrel, így még 384 Kb/mp-nél is elfogadható minőséget érthetünk el. Végül a videojel bitebességének felére csökkentésével (mely ismét a vízszintes és függőleges irányok alulmintavételezésével oldható meg) a mozgókép bitebessége az ISDN csatornasebességére, 64–128 Kb/mp-re csökkent. Nézzük meg, hogy közben mi történt a hangtechnikában. Az 1980-as évek elején a Philips és a Sony kifejlesztette a kompakt lemezt (CD), a lézerrel olvasható háttértárat (ezzel összevethető minőségű rendszert az RCA is kidolgozott, de tisztavág-életű volt). A CD kifejlesztését a csúcsminőségű sztereó zene tárolása iránti igény vezérelte: két hangcsatornát tartalmazott, melyen 44,1 kHz-es, 16 biten mintavételezett hang található.

A digitális hang- és képtechnológia összeolvadásának legjobb példája a közeljövőben az Egyesült Államokban és Európában egyaránt elterjeszteni kívánt HDTV (High Definition Television), azaz a nagyfelbontású képpel és kristálytisztá hanggal bíró digitális televízió.

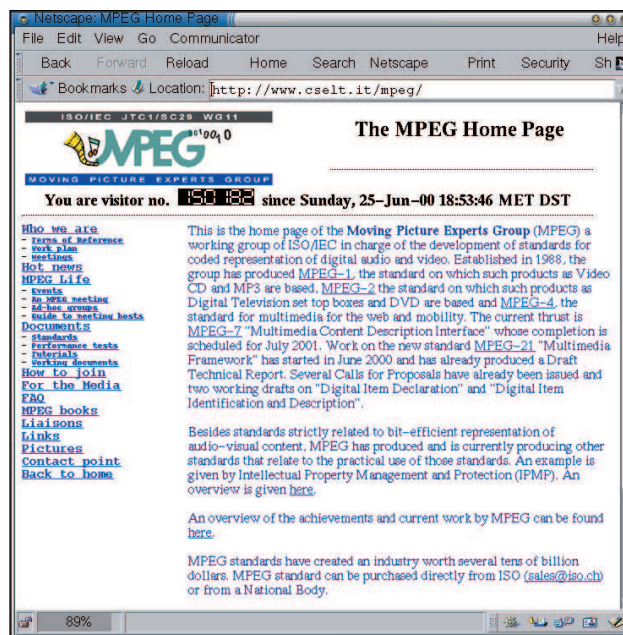
## Az első MPEG szabványok

Jómagam a távközléssel kapcsolatos kutatási területen szereztem tapasztalatokat. A távközlési iparra sokáig jellemző volt, hogy a hálózatokat igen gyorsan, anyagiakat és fáradságot nem kímélve fejlesztették, a végpontok kialakítására azonban sajnálták a pénzt. Ennek egyszerűen az az oka, hogy a távközlési ipar óriásai sokáig hidegletést kaptak a lakásokba és az irodákba szerelhető digitális végpont gondolatától is – pedig még egy kisiskolás is megállapíthatja, hogy a digitális, csúcsteljesítményű hálózatokra ezek nélkül nincsen szükség –, másrészt a végpontok kiépítése kívül esett e cégek tevékenységi körén. Ők azt szerették volna, ha a felhasználóknál elhelyezhető eszközöket más cégek gyártják le. A távközlési berendezéseket gyártó cégek sajnos a megszokott utat járták, így eszük ágában sem volt belevágni a jó öreg CCITT szabványokkal szembeszálló újdonság gyártásába. Bár a fogyasztói elektronikai cikkek gyártó vállalatok sokkal jobban érzékelték a végfelhasználók igényeit, és hozzászoktak ahhoz, hogy döntéseiket termékeik sikerességére és a vásárlók szükségleteire alapozzák, ennek ellenére ők sem érezték úgy, hogy nekik kellene lépni ez ügyben. Így történhetett meg az, hogy az 1980-as évek végén a faxgép kivételével gyakorlatilag egyetlen távközlési eszköz sem használt tömörítő és kibontó eljárásokat.

Az olcsó és kisméretű végberendezések gyártásához a készülék tulajdonságaihoz igazított integrált áramkörökre lett volna szükség (ezeket ASIC-nak nevezik), melyek a tömörítési eljárásokhoz szükséges bonyolult jelfeldolgozást végzik.

Azokban az években figyelemmel kísértem a Philips és az RCA fejlesztéseit, melyek a mozgókép digitális formában, CD-n való táro-

lására irányultak, és a céljuk az volt, hogy az interaktív alkalmazások hozzáférhessenek ezekhez (ilyen rendszer volt például a CD-i vagy a DVI). Ezek a megoldások a távközlési rendszerekben használható képtömörítő lapkákra támaszkodtak. Azonban hiányoztak a széles körben elfogadott szabványok, melyek véget vethettek volna az érdekelt cégek marakodásának.



www.cselet.ip/mpeg/

## MPEG-1

Ilyen előtörténettel indult 1988. januárjában az MPEG csoport, mely néhány hónap múlva megkezdte a hangjelek tömörítésének és a két adatfolyam összehangolásának megvalósítását is. Négy éven belül megszületett az első szabvány, az MPEG-1. Érdekes, hogy a két fő célcsoport (az interaktív CD és a digitális hangszolgáltatás) manapság nem használja e szabványt, de a mozgóképátvitel népszerűsége sem növekedett a várt mértékben. Másrészt az MPEG-1-et több millió VideoCD és MP3-lejátszó alkalmazza. A szabvány egyik legnagyobb újítása, hogy az MPEG-1 volt az első olyan multimédiás szabvány, melynek kifejlesztésénél a szimuláció is szerepet kapott.

A laboratórium, ahol dolgoztam, az 1,5-2 Mbit/másodperces videokonferencia tömörítőjének kidolgozásában is részt vett, néhány alapvető berendezés és némi számítógépes szimuláció segítségével.

A jövőbeli alkalmazások szempontjából még fontosabb, hogy az öt részből álló MPEG-1 szabvány programbeli megvalósítása az ISO/IEC 11172-5-ös szabvány ötödik részében található.

## MPEG-2

1990. júliusában az MPEG második kutatását, az MPEG-2-t is elindította. Míg az MPEG-1 pontosan meghatározott alkalmazási területtel rendelkezett, addig az MPEG-2 egy olyan gondot helyezett a középpontba, mely iránt mindenki érdeklődést tanúsított: hogyan alakítható át az ötvenéves analóg televíziózás tömörített, digitális formába oly módon, hogy az új rendszer minden lehetséges alkalmazási területen megállja a helyét. Ezt végül két rendszerre tagozták meg. Az egyik az MPEG-2 Transport Stream (TS), ezt a műholdas, kábeles műsorsugáráshoz használják. A másik az MPEG-2 Program Stream (PS), melyet számítógépes felhasználásra és a DVD-hez szántak. A cél az volt, hogy az MPEG-2 váljon a digitális televíziózás elfogadott formátumává. Ez meg is valósult, hiszen ma már az MPEG-2 TS formátum világszerte több adatot továbbít, mint az Internet protokollja, az IP. A szabvány címe a mozgókép és a hozzá tartozó hangjelek általános kódolása is ezt a célt sugalmazza. Mire az MPEG-2-t elfogadták (1994. novemberében), addigra széles körben elterjedtek az MPEG-1 valós idejű visszafejtésére és visszajátzására képes eszközök. A siker arra ösztönözte a fejlesztőket, hogy további programokat hozzanak létre az új szabványhoz (ISO/IEC 13818-5).

## MPEG-4

1993. januárjában az MPEG harmadik kutatását is elkezdte, mely az MPEG-4 nevet kapta. A terv célja, hogy kis sávzélességű, jó minőségű hangos mozgóképet valósítson meg. Bár akkoriban a tömeges

alkalmazásról még nem esett szó, ennek ellenére világossá vált, hogy a telefonhálózatokhoz hasonló keskeny sávú rendszerekhez olyan formátumra van szükség, melynek bitsebessége jóval az MPEG-1 és MPEG-2 által túlhaladott 1 Mbit/mp alatt van. Ehhez a sebességhez a programból megvalósított visszafejtő bizonyult a legmegfelelőbbnek. Ezenkívül a tervezők szinte biztosak voltak abban, hogy a születtendő felhasználások döntő többsége programalapú, és nem alkatrész-alapú megoldást tartalmaz majd.

Az MPEG-4 összefoglaló szabvány az audiovizuális adatok átvitelére. A szabvány támogatja az egyes audiovizuális adatok kódolt átvitelét, melyek térben és időben elfoglalt helyét a fogadó kapja meg. A jelenetet alkotó tárgyak több forrásból származhatnak: lehetnek természetesen vagy mesterségesek.

Ez azonban nem jelenti azt, hogy a szabvány megvalósítása túl bonyolult lenne. Az alkalmazásfejlesztők választhatnak a többféle eljárás közül, melyek mindegyike az MPEG-4 teljes eszköztárának részét képezi. Mindezek fényében arra számíthatunk, hogy az MPEG-4 hamarosan a széttöredezett multimédiás világ egységesítő erejévé válik.

### Miért szükséges az MPEG-4 szabvány?

Bizonyára elgondolkodnak azon, hogy ha a kódoló eljárás programból dolgozik, akkor miért van egyáltalán szükség a szabványra?

Miért nem megfelelő az a módszer, hogy az adott eljárás visszafejtését végző kódot egyszerűen letöltjük a Világhálóról?

Az MPEG-4 fejlesztésének első napjaiban gyakran föltettük ezt a kérdést, de manapság, amikor az MP3 rohamos elterjedésének vagyunk tanúi, már érthető a szabvány szükségessége: a lejátszó ugyanis nem feltétlenül csatlakozik a hálózatra. Lehet, hogy egy műsorsugárzó csatornához csatlakozik, vagy akár önálló, esetleg hordozható eszközről is szó lehet. Az eszközök többféle processzort használhatnak, és az túl drága lenne, hogy mindegyikhez kifejlesztենek a megfelelő lejátszót. Vagy mi történik akkor, ha az alkatrész netán olyan ASIC-ot használ a visszafejtéshez, mely nem frissíthető, vagy éppen annyi memória található benne, mely csak az alapértelmezett eljárás végrehajtásához elegendő. Más szóval: sokkal egyszerűbb olyan „forrásszabványt” alkotni, melyet a gyártók szabadon „továbbgondolhatnak”, mintsem állandóan a program-gépi környezet összeférhetlenségével bajlódni.

Végül megjegyeznénk, hogy a tömörítési kódolás jelentős változásokat idéz elő a mozgóképben. Általában elmondható, hogy minél kisebb a bitsebesség, annál gyengébb lesz a végeredmény minősége. Az egyik eljárásról a másikra történő átkódolás is rengeteget ronthat a minőségen. Így tehát az az elmélet, mely szerint a tömörítési eljárások fejlődnek, sajnos nem állja meg a helyét. Az MPEG annyi év után csak most ért el odáig, hogy a videótömörítési eljárások újrarendelését szorgalmazza, hiszen a csoport úgy érzi, hogy e területen lenne még mit figyelembe venni. A hangtömörítésnél még nagyobb az elmaradás, mert a csoport szerint ezt a tárgykört nem érdemes tovább erőltetni.

### A nyílt forrású MPEG-4

A nyílt forrású MPEG-4 szabvány fejlesztése olyan méretet öltött, hogy a referenciaprogram előállítását új alapokra helyezte. Éppen ezért érdekes megfigyelniünk, hogy egy ekkora méretű munkát miként hangoltak össze a tervezők. Csak a legfontosabbakat emelném ki:

- Az vitán felül állt, hogy a szabvány mindkét alkotórészét (a kódolást és visszafejtést) programból kell megvalósítani. Az is világos volt, hogy a forráskódot ki kell adni és a szabványt az ISO-nál kell bejegyeztetni.
- A szabvány minden részéhez kódfelelőségeket neveztek ki: a C és C++ részekhez a Microsoft és a MoMuSys egy-egy emberét, a természetes hanghoz a Fraunhofer Intézetet, a strukturált hanghoz az MIT-t (Massachusetts Institute of Technology) a beszédrendszer-

erhez a szöveg, illetve átalakításához az ETRI-t, a kód magjához az Optibase-t, az MPEG-4 fájlformátumhoz az Apple-t stb.

- Minden részterülethez kísérleti felügyelő is kijelöltek, akinek az elfogadott eszközöket alkotó kódoknak a létező kódalapba történő bedolgozása volt a feladata.
- A hagyományos nyílt forrású fejlesztésekkel ellentétben csak az MPEG tagjai vehettek részt a munkában. A megbeszéléseket levelezőlistákon folytatták, ahol a tervezők a külső segítők véleményét is meghallgatták.

Az MPEG csoportban állandóan új ötletek születnek. Az egyik ötlet a referenciaprogram továbbfejlesztése volt, melynek munkálatai 1999. decemberében kezdődtek. Ez a javított kód többek között mozgásvektorokkal is dolgozik, mely az MPEG-4 szabvány leginkább számítási-igényes része. Ezt a kódot bárki felhasználhatja saját programjában, jogi és egyéb díjak fizetése nélkül. Abban megegyeztünk, hogy ezek a fontos kódrészek nem igényelhetnek szabadalmakat. Tavaly októberben döntés született egy új vonalról az *MPEG-4 referenciagép leírás*-áról. Reményeink szerint ez tovább népszerűsíti az MPEG-4 szabványt mind a programok, mind az alkatrészek világában. Az alábbiakban közlöm az összes MPEG-4 programelemhez mellékelt szerzői jogi adatokat.

Az úgynevezett „szerzői jog” szövege az alábbi:

Ezt a programelemet eredetileg <vezetéknev1> <keresztnev1> (<cégnév1>) fejlesztette és <vezetéknev2> <keresztnev2> (<cégnév2>), valamint <vezetéknev3> <keresztnev3> (<cégnév3>) szerkesztette, az <MPEG szabvány> részeként. Ez a programelem az <MPEG szabvány> által meghatározott <MPEG szabvány> eszközök egy vagy több részének megvalósítása. Az ISO/IEC az <MPEG szabvány> felhasználói számára lehetőséget ad e programelem ingyenes használatára és módosítására, amennyiben a végtermék az <MPEG szabvány>-nyal való megfeleléségre törekszik. E programelem eszközökben vagy programokban felhasználni kívánók figyelmét felhívjuk arra, hogy annak használata bejegyzett szabadalmakat sérthet. E programelem eredeti tervezője és annak vállalata, a későbbi szerkesztők és azok vállalatai, valamint az ISO/IEC nem vállal felelősséget ezen programelem használatából és módosításából eredő jogi következményekért. A szerzői jog nem vonatkozatható az <MPEG szabvány>-nak meg nem felelő termékekre. A <cégnév1> minden jogot fenntart a kód bármilyen használatára, továbbadására, illetve annak megakadályozására, hogy egy harmadik fél a kódot az <MPEG szabvány>-nak meg nem felelő termékben használja föl. Ezt a szerzői jogi nyilatkozatot a programelem minden példányával és annak származékaival együtt kell terjeszteni. Copyright (évszám).

### Jelen és jövő

Jelenleg az MPEG csoport az MPEG-7 nevű, a multimédiás tartalom leírására szolgáló eszköz kifejlesztésének utolsó szakaszában tart.

Ennek a szabványnak célja a mozgókép és hang leírása, függetlenül attól, hogy az egész filmről vagy csupán egyik képkockájáról van-e szó. A szabványt 2001. júliusában fogadják el. Az MPEG-7-hez máris hatalmas mennyiségű referenciaprogram tartozik, melyeket az MPEG-4-nél alkalmazott elvek alapján fejlesztettek. 2000. júniusában az MPEG elindította legújabb, MPEG-21 névre hallgató tervét, amit más szakértői csoportokkal közösen fejleszt.

A munka célja, hogy a multimédiás tartalom hálózati továbbításához használt eljárásokat összefogja és egységesítse.

A munka legfontosabb lépései a következők:

1. A „digitális tárgyak” világos meghatározása.
2. A tartalom megformálása.
3. A digitális tárgyak azonosítása és leírása.
4. Tartalomkezelés és -használat: a tartalom létrehozására, módosítására, keresésére, elérésére, tárolására, átvitelére és (újra)felhasználására alkalmas csatlók és protokollok létrehozása.

5. A szellemi tulajdon kezelése és védelme: a hálózaton terjesztett és tárolt multimédiás anyagokhoz kapcsolódó jogok védelme.
6. Végpontok és hálózatok: a tartalmat a hálózatok és végpontok egymás között megoszthatóvá tehetik.
7. Események jelentése: minden olyan eszköz, mely ahhoz szükséges, hogy a felhasználók állandóan friss adattal rendelkezzenek a hálózat elemeinek működésével kapcsolatban.

Mіндеzek közül az 5. pont a legérdekesebb. Az MPEG-2 óta a csoport különös gondot fordít arra, hogy a tartalmat pénzért szolgáltató vállalatok számára is megfelelő megoldások szülessenek. Eddig az MPEG által kínált megoldások lehetővé tették a cégek saját védelmi eljárásainak használatát. Ennek azonban az a hátránya, hogy a titkosított adatok nem érhetők el olyan egyszerűen a felhasználók számára, még akkor sem, ha a felhasználók tiszteletben tartják a szerzői jogokat. Az MPEG éppen emiatt fáradozik olyan megoldás kifejlesztésén, mely a könnyen felhasználható védett tartalmat teszi elérhetővé.

### Az MPEG szabványokhoz kapcsolódó szabadalmak

A XV. századi Velencében és Firenzében már alkalmaztak szabadalmakat, de Mainzban a fogalom ismeretlen volt. Így Gutenberg számára szellemi alkotásai védelmére az egyetlen megoldás az volt, hogy mindent elrejtett ismerősei, üzletfelei elől, ami természetesen bukással végződött. A XIX. században minden, mozgóképpel vagy hanggal kapcsolatos találmányt szabadalmakkal védtek le. Ez a gyakorlat a XX. században is folytatódott, bár ekkor már magánszemélyek helyett inkább cégek foglalkoztak a szabadalmak bejegyzésével. Amikorra a digitális eljárások használatának előnyei nyilvánvalóvá váltak, csaknem minden érintett vállalat, szervezet támogatta a hang- és mozgóképkódolással kapcsolatos kutatásokat. Ma több ezer ilyen jellegű szabadalom létezik.

Amikor az MPEG megkezdte munkáját e szakterületen, nyilvánvaló volt, hogy vagy beáll a sorba, és szabadalmakkal védi kutatásait, vagy egyszerűen képtelen lesz azokból bármi hasznosat és általánosan elfogadottat létrehozni. Többek között ezekkel a nehézségekkel kellett szembenéznie az eleve tőke nélkül induló csapatnak.

A szabványokhoz kapcsolódó szabadalmak gondja természetesen a három legnagyobb nemzetközi szabványügyi hivatal, az IEC, az ISO és az ITU előtt sem volt ismeretlen. Ők az alábbi általános elveket alkalmazzák:

1. Semmilyen szabadalomhoz nem lehet szükség szabvány megvalósításához, vagy
2. a jogtulajdonosnak el kell engednie a jogokat, vagy
3. a jogtulajdonosnak nyilatkozatban kell állást foglalnia abban a kérdésben, hogy kinek engedélyezi a szabadalom használatát, és ezt igazságos és megindokolható feltételek alapján kell megtennie.

Az MPEG éppen ezért a szabványok kifejlesztése során nem foglalkozik a szabadalmakkal, és csak a minél hatékonyabb teljesítményre és minél szélesebb körű alkalmazhatóságra törekszik. Ennek természetesen az lett az eredménye, hogy az MPEG szabványokhoz általában nagy számú szabadalom kapcsolódik.

Egy MPEG-2 visszafejtő megvalósításához állítólag úgy száz szabadalom szükséges. Mivel a cégek igencsak érdekeltek abban, hogy e szabadalmakat lehetőleg egy helyről szerezzék be, ezért külön szervezet alakult az MPEG-2-höz kapcsolódó szabadalmak engedélyeztetésére. Bár ez így nyilván kényelmesebb, az azért mégis érdekes, hogy az MPEG-2 szabadalmiáért fizetett díjak ennek ellenére nem csökkentek, a szabadalmak száma viszont nőtt.

Hasonló események zajlottak le az MPEG-4 körül is. Az MPEG-4 Industry Forum (<http://www.m4if.org/>) az MPEG-4-hez előre összeállított szabadalmi csomagok megalkotására jött létre. Természetesen az MPEG-4 helyzete jóval összetettebb, hiszen számos üzleti modell esetén szükség van arra, hogy a lejátszót a felhasználó letöltse a cég weboldaláról. Az MPEG-7-hez is tervezik hasonló szervezet felállítását.

### Összegzés

Bár teljesen más irányból, de az adatfeldolgozással foglalkozó cégekhez hasonlóan a multimédiás világ egyik fő képviselője, az MPEG is eljutott arra a pontra, hogy szükségesnek ítélje a nyílt forrású fejlesztést. A hatalmas különbség azonban az, hogy míg az előbbi csoport teljesen nyílt eljárásokat hirdet, addig az MPEG kénytelen alkalmazkodni a multimédiás világ törvényeihez, ahol, tetszik vagy nem, szabadalmakkal kell „megküzdeni” minden nap. Így tehát a referenciaprogram (és a hasonló célú és jellegű gépi leírás) szerzői jogoktól mentes, de általában nem szabadalommentes.

Az MPEG-21 esetében, mely a hálózati tartalom általános rendszerleíró szabványa, a szabványosítás egy magasabb szintre kerül, mint amit eddig tapasztalhattunk a többi fejlesztésnél. Mivel a referenciaprogramok mára teljesen az MPEG szabványok részét képezik, ezért arra számíthatunk, hogy az MPEG csoportnak hatalmas nehézségekkel kell megküzdenie, ha majd a szabadelvű javaslatokat kell a régimódi, valós körülményekhez igazítani. Ennek ellenére azt hiszem, a gondok megoldásához azonban sokkal egészségesebb környezetet biztosít egy szakértőkből álló csoport, mint egy bírósági tárgyalóterem. Mindenki együttműködésére szükség van.



Leonardo Chiariglione

1971-ben csatlakozott a Telecom Italia csoport vállalati kutatóközpontjához, a CSELT-hez, ahol ő lett a műsorsugárzási eljárások kutató-részlegének vezetője. 1988-ban indítványozására jött létre az ISO-MPEG szabványcsoport.

Leonardo az Image Communications, az EURASIP képviselteti eljárásokkal foglalkozó magazinjának alapítója és főszerkesztője. 1994-ben alapítója volt a Digital Audio-Visual Council (DAVIC) csoportnak, amelynek egy évig elnöke volt. 1999-ben a Secure Digital Music Initiative (SDMI) igazgatójává nevezték ki, ahol a biztonságos digitális zenéhez kapcsolódó üzleti modellek kialakításának feladatával bízták meg.

### Kapcsolódó címek

MPEG ➔ <http://www.csel.it/mpeg/>

Az MPEG-1 rövid ismertetése

➔ <http://www.csel.it/mpeg/standards/mpeg-1/>

Az MPEG-2 rövid ismertetése

➔ <http://www.csel.it/mpeg/standards/mpeg-2/>

Az MPEG-4 rövid ismertetése

➔ <http://www.csel.it/mpeg/standards/mpeg-4/>

Az MPEG-4 hivatalos ismertetése ➔ [http://www.csel.it/mpeg/public/mpeg-4\\_procedures.htm](http://www.csel.it/mpeg/public/mpeg-4_procedures.htm).

Az MPEG megalakulásának körülményeiről részletesebben is olvashatunk a ➔ <http://www.csel.it/leonardo/paper/conftele99/conftele99.htm> címen

Az MPEG-4 referenciaprogramjának megvásárlásához írjunk a [sales@iso.ch](mailto:sales@iso.ch) címre, vagy töltsük le a

➔ <http://www.iso.ch/ittf/weboldalrol>.

ISO ➔ <http://www.iso.ch/>

Az MPEG-7 rövid leírása ➔ <http://www.csel.it/mpeg/standards/mpeg-7/>

A még hivatalosan el nem fogadott MPEG-21 összefoglaló ismertetése ➔ [http://www.csel.it/public/mpeg-21\\_pdtr.zip](http://www.csel.it/public/mpeg-21_pdtr.zip)

## JavaServer Pages

Bemutatjuk a JSP-ket, amelyek mind a programozók, mind a laikusok számára lehetővé teszik servletek és dinamikus lapok készítését, Java és HTML kód keverésével.

**E**lőző írásunkban vetettünk egy pillantást a kiszolgálóoldali Javára, és éppen csak lábujjainkat mártva a mélyvízbe, írtunk pár servletet. Ezek olyan Java programok, amelyek dinamikus webtartalmat készítenek. A CGI programok olyan végrehajtható állományok, amelyek nem tartoznak közvetlenül a webkiszolgálóhoz, és minden meghíváskor a lelejtől kezdve végrehajtnak. Ezzel szemben a Java servletek a Java virtuális gépben (JVM), a servlet-tárolóban találhatók, amely közvetlenül csatlakozik a http-kiszolgálóhoz. Amikor a webkiszolgálónak dinamikus tartalomra van szüksége, a servlettárolóhoz fordul.

A Java servlet készítés sok szempontból hasonlít egy mod\_perl kezelő írásához: jókora előnyöket nyújt, de megkövetel bizonyos fegyelmeztséget is. Lehangoló lehet olyan servletet írni, amely kilencven százalékgig statikus HTML kódot és mindössze tízs százaléknyi Java kódot tartalmaz. Ilyen sokszor meghívni az out.println()-t, már egyenesen őrlítő. Egyre népszerűbb megoldás erre a gondra a JavaServer Pages azaz a JSP. Ez hasonló szellemiségben készült, mint a Microsoft ASP, vagy akár a nyílt forráskódú PHP nyelv és Mason kiegészítő rendszer. A JSP lehetővé teszi, hogy a Java és a HTML nyelvet számos módon keverjük. Tulajdonképpen a JSP, akárcsak a legtöbb Java program, figyelemreméltóan felületfüggetlen, ez azt jelenti, hogy írhatunk JSP-ket windowsos gépen, amit aztán fejlesztés közben linuxos kiszolgálón alkalmazunk, végül pedig Solarisra telepítjük. Ebben az írásban gyors pillantást vetünk a JSP-re, ez jó alkalmat teremt arra, hogy egy kis Javát tanuljunk, és különösebb erőfeszítések nélkül, könnyű servletkészítési lehetőséget tárjunk a Java-programozók elé.

### Hogyan működik a JSP

A JSP-k alapötlete igen figyelemreméltó módon egyszerű: tulajdonképpen servletek áruházában. Amikor a JSP-t első ízben hívják meg önműködően servletté alakul. Ez a servlet fordítódik le aztán egy Java .class fájlba, amely ezután a servlettároló belsejéből hajtódik végre. A JSP első meghívásakor kicsit lassabban fognak az adatok a felhasználóhoz jutni, hiszen a színpalak mögött az alábbiakban részletezett folyamatok játszódnak le. JSP alatt, minden alapvetően statikus tartalomnak tekintendő, ami nincs különleges zárójel közé helyezve: <% és %>. Az ilyen tagok közé zárt kódrészletet „scriptlet”-nek nevezik a JSP tolvajnyelvén. A következő HTML-fájl – amit main.jsp-nek neveztünk el – is teljesen megfelel a JSP szabványoknak:

```
<HTML>
  <Head>
    <Title>Static JSP Title</Title>
  </Head>

  <Body>
    <P>Static JSP Content</P>
  </Body>
</HTML>
```

Természetesen a fenti JSP meglehetősen unalmas, hiszen kizárólag statikus tartalommal bír. De a JSP-motor egyáltalán nem törődik

azzal, hogy egy JSP mennyi dinamikus tartalmat foglal magába; az egész dolgot servletté alakítja, függetlenül attól, hogy mennyire összetett. A fenti JSP esetében az eredményül kapott servlet nem sokkal lesz több, mint egy hosszú karaktersorozat és egy out.println() utasítás egy doGet() eljárás belsejében.

A saját rendszeremen, a fenti HTML kódot a main.jsp fájlba mentetem, a Tomcattal együtt érkező /usr/java/jakarta-tomcat-3.2.1/webapps/examples/jsp/ könyvtár példakönyvtárába.

Ez nyilvánvalóan nem a legmegfelelőbb hely, de ez volt számomra a legegyszerűbb megoldás.

Ha a JSP-t már feltelepítettem, nem is kell többet tennem; a rendszer önműködően fogja servlet forrássá (.java) alakítani, és Java .class fájlba fordítani.

Végre is hajthatjuk és megtekinthetjük az új JSP-eket a Tomcat kiszolgálón keresztül, amely az alapértelmezés szerint a 8080-as kapura figyel ☞ <http://localhost:8080/examples/jsp/main.jsp/>.

Ha az Apache-t és a mod\_jk-t úgy állítottuk be, hogy a továbbítsa a servlet és a JSP kérélmeket a Tomcat kiszolgálóhoz, akkor a main.jsp-t megtekinthetjük a

☞ <http://localhost/examples/jsp/main.jsp/> címen is.

Az én gépemen a JSP-rendszer által a main.jsp-hez készített .java és a .class fájlok a következő könyvtárban helyezkednek el: /usr/java/jakarta-tomcat-3.2.1/work/localhost\_8080%2Fexamples. Ha kilistázom a könyvtár tartalmát, a következőket látom:

```
_0002fjsp_0002fmain_0002ejspmain.class
_0002fjsp_0002fmain_0002ejspmain_jsp_0.java
_0002fjsp_0002fmain_0002ejspmain_jsp_1.java
_0002fjsp_0002fmain_0002ejspmain_jsp_2.java
_0002fjsp_0002fmain_0002ejspmain_jsp_3.java
_0002fjsp_0002fmain_0002ejspmain_jsp_4.java
_0002fjsp_0002fmain_0002ejspmain_jsp_5.java
_0002fjsp_0002fmain_0002ejspmain_jsp_6.java
```

Amint látható, jó néhány különféle .java fájl helyezkedik itt el, mind egyik az eredeti JSP egy-egy változata. Akárhányszor csak megváltoztatom a JSP-t, a rendszernek új .java fájlt kell készítenie. A Tomcat alapértelmezés szerint megtartja az előző JSP-alapú servleteket, de egyszerre csak egyetlen .class fájl létezhet, akárcsak ebben az esetben. A .java és a .class fájlnevek meglehetősen hosszúak és nem arra szánták, hogy a böngészőbe közvetlenül begépeljük őket. A JSP varázslat része ez is. A Tomcat képes arra, hogy az adott URL-lel összerendelt servletet értelmesen és önműködően megtalálja, és elkészítse a Java forrásfájlokat, ha szükséges.

Érdemes egy pillantást vetni a JSP által létrehozott Java forráskódra, hogy képet alkothassunk arról, milyen komoly munka megy végbe a háttérben. Az egyszerű, statikus JSP kódunk servletté alakult, amely több mint száz soros Java forráskódból áll. Ha vesszük a fáradságot, és visszafejtjük a JSP kódunkat a lefordított servlet forrásból – ami igen nagy kihívás, legalább akkora mintha a Perl HTML::Mason-t próbálnánk visszafejteni – a servletben megjegyzéseket találhatunk, amelyek az eredeti JSP sorok sorszámai és a lefordított kód közötti alapvető kapcsolat feltérképezésében nyújthatnak segítséget.

## 1. lista A &lt;% %&gt; tagok használata

```
<HTML>
  <Head>
    <Title>Mini-dynamic JSP Title</Title>
  </Head>
  <Body>

    <P>You are connecting from the host

    <% if (request.getRemoteHost().equals(""))
      { %>

      <%= request.getRemoteHost() %>.</P>
    <% } else { %>

      <%= request.getRemoteAddr() %>.</P>
    <% } %>

  </Body>
</HTML>
```

## 2. lista Az Include irányelv használata

```
<HTML>
  <Head>
    <Title>Mini-dynamic JSP Title</Title>
  </Head>
  <Body>

    <%@ include file="menubar.jsp" %>

    <P>You are connecting from the host

    <% if (request.getRemoteHost().equals(""))
      { %>

      <%= request.getRemoteHost() %>.</P>
    <% } else { %>

      <%= request.getRemoteAddr() %>.</P>
    <% } %>

  </Body>
</HTML>
```

## Dinamikus tartalom

Kicsit érdekesebbé tehetjük a dolgokat, ha egy különleges JSP tagot szúrunk be a main.jsp-be. A tag valamilyen Java kód eredményét szűrja be a felhasználóhoz küldendő kimenetbe:

```
<HTML>
  <Head>
    <Title>Mini-dynamic JSP Title</Title>
  </Head>
  <Body>
    <P>You are connecting from
      <%= request.getRemoteHost() %>.</P>
  </Body>
</HTML>
```

A <%= és a %> jelek közötti kérelem végrehajtható, és visszatérési értéke a kimenetre kerül. Mivel a JSP tulajdonképpen egy álrúhás servlet, minden olyan objektumhoz hozzáférhet, amihez egyébként a rendes servlet elér. Ilyenek például a „request” és a „response”. Figyeljük meg, hogy a <%= és a %> között elhelyezkedő Java kódot nem zárja le pontosvessző! Saját tapasztalatból tudom, hogy igen nehéz leszokni arról, hogy pontosvesszőt tegyünk ide, de a JSP kiakad, ha ragaszkodunk hozzá.

Amennyiben egy vagy több Java műveletet szeretnénk elvégezni anélkül, hogy az eredményt elküldenénk a felhasználó böngészőjére, használjuk az egyszerű <% és %> tagokat. Ezek összevegyíthetők a HTML-lel, ami lehetővé teszi, hogy feltételes szövegek jelenjenek meg a válaszban (lásd az 1. listát).

Ha a felhasználó gépének neve elérhető, kiírjuk azt, egyébként a gazdagép IP-címét jelenítjük meg. Figyeljük meg, hogyan lehet az if/then/else blokkokat összeegyeztíteni a statikus HTML kóddal. A request.getRemoteAddr() hívás csak akkor kerül végrehajtásra, ha a request.getRemoteHost() üres karaktersorozatot adott vissza (""). Számos JSP irányelvet (directive) a <%@ és %> tagpárral adunk meg. Minden ilyen irányelv a JSP-ből servletté való alakítás során kerül feldolgozásra. A directive kulcsszó közvetlenül a @ jel mögött helyezkedik el, amit aztán nulla vagy több elem követ. Például, tételezzük fel, hogy van egy honlapszerzte előforduló menü-sorunk, a menubar.jsp nevű JSP fájlban:

```
<table>
<tr>
  <td><a href="one">Option 1</a></td>
</tr>
<tr>
  <td><a href="two">Option 2</a></td>
</tr>
</table>
```

Ezt a részletet könnyen beépíthetjük a dokumentumunkba az „include” irányelv használatával (lásd a 2. listát).

Fontos azonban tudni, hogy az irányelvek a JSP fordításakor hajtódnak végre és nem futásidőben. Így a fenti példa csak addig fog kifogástalanul működni, amíg meg nem változtatjuk a menubar.jsp fájlt. Mivel a menubar.jsp tartalma már akkor beépült a main.jsp-be mikor servletté alakult, az <%@ include %> tag pedig többé már nem létezik, így aztán, a dolgok nem úgy fognak frissülni, ahogy azt elvárnánk. A megoldás későbbiekben ismertetett, futás idejű JSP művelet használata. Van még két különleges JSP tag. Az egyik közülük a <%-- --%>, mely megjegyzésként használható. Bár feleslegesnek tűnhet JSP megjegyzések léte, mikor HTML megjegyzések már régóta léteznek, de egy alapvető különbségre mindenképpen szeretném felhívni a figyelmet: a JSP megjegyzéseket a JSP-motor távolítja el, a servlet készítése közben. Ezzel szemben a HTML megjegyzéseket érintetlenül továbbengedi, így azok minden végfelhasználó számára láthatók, ha a „view source” menüpontot választják a böngészőjükben. Mivel lehet választani, többnyire magam is inkább JSP tagokat használok a megjegyzéseimhez, kivéve azokat, amelyek a JSP kódomban HTML forrás kimenetének hibakereséséhez használhatók. Az utolsó JSP-tag a <%! %>. Ennek segítségével a futó servlet számára példányváltozókat (úgynevezett mezőket) vezethetünk be. Habár igen csábító lehetőségnek tűnik, hogy a JSP-ben használt összes változót előre bevezessük, emlékezzünk viszont arra is, hogy a mezők alkalmazása azt is jelenti, hogy magunknak kell ügyelnünk a szálak biztonságos kezelésére is. Mivel a szálkezeléshez többnyire fejfájás társul, ahol lehet, célszerűbb elkerülni alkalmazását. A <%! %> tagot helyi eljárások meghatározására is lehet használni, ám nem vagyok meggyőződve arról, hogy ez olyan jó ötlet.

5. lista showblog.jsp

```

<%@ page language="java" contentType="text/html"
%>
<%@ page import="java.sql.*" %>

<HTML>
  <Head>
    <Title>Weblog</Title>
  </Head>
  <Body>

    <H1>Weblog</H1>

  <%
    // Néhány saját változóbevezetés az
    // adatbázis használatához

    String loginJdbc = "org.postgresql.Driver";
    String loginUser = "reuven";
    String loginPasswd = "";
    String loginUrl = "jdbc:postgresql:
//localhost/atf";

    // A PostgreSQL vezérlő betöltése
    Class.forName(loginJdbc);
    Connection con = DriverManager.getConnection
        (loginUrl, loginUser, loginPasswd);

    // A kifejezés megadása
    Statement statement = con.createStatement();

    // Bizonyosodjunk meg afelől, hogy egy időben
    // csak egy szál kezeli ezt a kifejezést

    String query = "SELECT entry_date, ";
    query += "entry_headline, ";
    query += "entry_text ";
    query += "FROM BlogEntries ";
    query += "ORDER BY entry_date; ";
    query += "DESC; ";

    // Lekérdezés végrehajtása
    ResultSet rs =
    statement.executeQuery(query);
  %>

  <table>

  <% while (rs.next()) { %>

    <tr>
      <td <%= rs.getString("entry_date") %>
      </td>
      <td <%= rs.getString("entry_headline") %>
      </td>
      <td <%= rs.getString("entry_text") %></td>
    </tr>

  <% } %>

</table>

  </Body>
</HTML>

```

## JSP eljárások

Az irányelvek igen hasznosnak bizonyulhatnak, ha a servlet felépítésének folyamatát szeretnénk befolyásolni. De mi történik, ha a servlet eljárásait futásidőben szeretnénk megváltoztatni?

Természetesen belefoglalhatnánk (include) egy Java kódot, ami végrehajtja a szükséges eljárásokat. A JSP azonban számos különleges tagot tartalmaz, amelyek a servletben Java kódba fordítódnak, így bárki képes a kód megírására, még azok is, akik egyáltalán nem ismerik a Javát.

A JSP eljárások tulajdonképpen XML-nek megfelelők és tagkönyvtáraknak nevezett különleges XML dokumentumokban találhatóak. Így, habár külsőre HTML-nek tűnnek, valójában nem azok, ami többnyire azt jelenti, hogy különösen nagy figyelmet kell fordítanunk az olyan elemekre, mint a bezáró tagok vagy a perjelek.

A beépített JSP tagkönyvtár számos függvényt is tartalmaz, amelyek közül az egyik gyanúsán hasonlít a második listában megismert include irányelvhez. A 3. lista a main.html

([www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)) egy olyan változatát mutatja be, amelyik a menubar.jsp irányelv felhozásához a <jsp:include> műveletet alkalmazza.

Az eltérés a két változat között apró, de igen jelentős: a beépítés irányelv ugyanis akkor fejt ki hatását az adott lapra mikor a JSP servletté alakul át, az include művelet ellenben futásidőben hajtódik végre. Ha a menubar.jsp-t megváltoztatjuk a main.jsp két futtatása közt, az irányelvet alkalmazó változat figyelmen kívül hagyja az új menüt, a műveletet használó változat viszont a legutóbbi változatot

jeleníti meg. Természetesen mindennek megvan az ára; a <jsp:include> művelet futásidejű kérelmet hajt végre, ami lassabbá és kevésbé hatékonyá teszi irányelvalapú testvérénél.

Mivel a <jsp:include> által beolvasott kód a legfelső szintű JSP minden kérelemadatához hozzáférhet, lehetőség nyílik dinamikusan változó menürendszerek, igény szerint alakítható rendszerek és adatbázis-elérést megvalósító könyvtárak készítésére.

Vannak más JSP műveletek is. Az egyik közülük a <jsp:forward>, amely a kérelmet egy másik JSP-hez irányítja. Akárcsak a <jsp:include> esetében, ez a művelet is a servletmotorban hajtódik végre, ami azt jelenti, hogy a http-kérelm és a felhasználói kapcsolatok továbbra is elérhetőek maradnak.

A 4. lista ([www.linuxvilag.hu/magazin/cikkekhez.html](http://www.linuxvilag.hu/magazin/cikkekhez.html)) például, az eredeti JSP-nk egy olyan változatát mutatja be, amelyik másik lapra irányítja át a felhasználót, ha gazdagépének neve nem állapítható meg. Ha a kiszolgálónk esetleg nem tartalmaz no-reverse.jsp nevű lapot, a felhasználók 404-es (a fájl nem található) hibával találkozhatnak a böngészőjükben. Ugyanakkor a böngészőjük továbbra is az eredetileg kért, main.jsp lapra mutató címet írni ki forrásként. Ez azért van így, mert az átirányítás a motor belsejében megy végbe, külső http-átírányítás nélkül.

## Egyszerű webnaplózó JSP

Előző cikkemben (64. oldal) írtunk pár egyszerű servletet, amelyek segítségével webnaplókat (blogokat) készíthettünk, és nézhettünk meg. Mivel a JSP-k servletekké fordítódnak, semmi sem szól az

ellen, hogy olyan JSP-eket készítsünk, amelyek ugyanazt tudják, mint azok a servletek. Nyilvánvalóan kicsit másképp fognak kinézni, de a hatásuk ugyanaz lesz.

Az 5. lista JSP-t mutat be (showblog.jsp), amely ugyanazt a feladatot hajtja végre, amit az előző cikkben ShowBlog servlet. Azaz, kiírja a PostgreSQL adatbázistáblában tárolt webnapló tartalmát, a legfrissebbtől a legrégebbi bejegyzésig rendezve.

Már most felhívom rá a figyelmet, hogy a showblog.jsp meglehetősen borzalmas példa arra, hogyan írunk JSP-eket; a feladata egyszerűen az, hogy bemutassa, mi mindent lehet megvalósítani, és nem törekszik választékos megoldásokra. Ezekkel a dolgokkal a két következő részben fogunk foglalkozni, amikor a JavaBeans és a saját tagkönyvtárak kerülnek terítékre.

Menjünk végig lépésenként ezen a JSP-n, hogy pontosan megvizsgáljuk, miképpen működik. Két „page” irányelvvel kezdünk. Ezek teszik lehetővé számunkra a JSP-nk alapvető beállításának elvégzését, kezdve a lap által visszaadott MIME Content Type fejléc beállításával. Sőt, azt is lehetővé teszi, hogy meghatározzuk, milyen programozási nyelvet keverünk a nem programozott szövegbe. Bár ilyen megoldás egyelőre nem létezik, elméletileg lehetséges olyan JSP-t írni, amely Perl-t használ az XML előállításához, vagy Python-t a PNG képek létrehozásához.

Figyeljük meg, hogy a page irányelvben egy vagy több értéket is megnevezhetünk. A showblog.jsp első sora egyszerre állítja be a nyelvet és a Content Type értéket. A második sor azt mutatja, hogy az eredményül kapott servletnek kell bejuttatni majd a java.sql-ben található csomagokat, melyek segítségével JDBC-n keresztül összekapcsolódhatunk a relációs adatbázis-kiszolgálókkal (jelen esetben a PostgreSQL-lel).

Egy rövid bevezető HTML kód után, mélyen elmerülünk a Javában. Készítünk egy SQL kapcsolat-objektumot, és segítségével hozzákapszolódunk a PostgreSQL kiszolgálónkhoz. Letöltjük az adatokat az adatbázisból, majd sorról sorra végiglépkedünk az eredménykészleten (ResultSet).

## Valami nincs rendben

Amint azt az imént említettem, ez csúnya JSP írási mód, ezenkívül rettenetesen rossz teljesítményt érünk el a rengeteg, ismételt felépített és lebontott adatbázis-kapcsolat miatt, ráadásul még bizonyos mennyiségű HTML és programkód zagyvalékot is készítettünk. Tulajdonképpen az egyetlen, amit nyertünk a dologgal, hogy a HTML kiírásához nem kell out.println() eljárásokat használnunk.

Továbbá, a JSP-k alapja pont az lenne, hogy a HTML lapokból eltávolítsuk a kódot, és a nem programozók számára is lehetővé tegyük, hogy kevés erőfeszítéssel dinamikus HTML lapokat készítsenek. Ha ilyen sok kódot kell beszúrniuk a dinamikus lapok létrehozásához, elég kicsi az esélye, hogy egy nem programozó rászánja magát a webfejlesztésre.

Ezenkívül, miközben a ShowBlog servletünket JSP-vé alakítottuk, jó néhány kivételkezelő eljárást eltávolítottunk. A servletünk elég értelmes volt ahhoz, hogy a PostgreSQL kiszolgáló eltűnését is kezelje, és erről megfelelő hibajelzést adjon. A JSP-nk, ezzel szemben, mindössze egy hibaüzenetet tartalmazó backtrace vonalat ad vissza. Ez a nyomvonal hasznos lehet a fejlesztőknek, de egyáltalán nem barátságos vagy hasznos a végfelhasználóknak, akik meglátogatják a lapunkat (az igazság kedvéért el kell mondani, hogy a lap irányelvei közt beállíthatunk egy errorPage értéket, így ezek a hibák egy másik JSP-hez irányíthatók).

A jó megoldás az lenne, ha annyi kódot távolítanánk el a JSP-ből, amennyit csak bírunk, így téve lehetővé a nem programozó számára, hogy hagyományos módon használhassa a kódot, illetve, hogy elválasszuk a programozott és a nem programozott tartalmat.

És valóban, a JSP együtt érkezik a JavaBeans támogatással, ahol minden egyes „bean” (babszem) tulajdonképpen egy számos tagfüggvénnyel rendelkező objektum, amelyet JSP-nkben a különleges <jsp: > művelettel használhatunk ki. A sikeres JSP fejlesztés titka, nem kétséges, a JavaBeans okos felhasználása. Továbbá a JSP lehetővé teszi számunkra, hogy saját műveleteket meghatározó tagkönyvtárakat fejlesszünk ki, így a megfelelő <jsp: > művelethez képest még több kódot tudunk tagokkal helyettesíteni.

## Következtetés

A JavaServer Pages, azaz a JSP, egy új írásmódot von a servletek köré, amit egyébként talán kicsit nehéz lenne egy nem programozónak megtanulnia. Ugyanakkor e hónap legbonyolultabb példája azt is megmutatta, hogy könnyen rossz útra tévedhetünk, ha majdnem annyi kód begépelésével érjük el célunkat, mint amennyi egy hagyományos servlet megírásához kellene. Bár a JSP-kkel általában véve könnyebb dolgozni, a HTML és kód szétválasztásából származó előny semmivé foszlik, ahogy a JSP egyre összetettebbé válik.

A következő hónapban belepillantunk a JavaBeans titkaiba, amely lehetővé teszi számunkra, hogy a sok programozást kiváltsuk a JSP-n kívül meghatározott és karbantartott osztályok segítségével. Ezután megismerkedünk az egyéni JSP tagkönyvtárak készítésének lépéseivel, amelyek lehetővé teszik, hogy saját kis nyelveinket használjuk a JSP-nken belül.



Reuven M. Lerner

(reuven@lerner.co.il) kisebb, webes és internetes módszerekkel foglalkozó tanácsadó cég tulajdonosa és vezetője. A cikk megjelenésének időpontjában valószínűleg már végleg elkészült Core Perl című könyvével, melyet idén jelentet meg a Prentice-Hall. Az ATF honlapon érhetjük őt el ☞ <http://www.lerner.co.il/atf/>

## Források

Ahogy a JSP-k egyre népszerűbbekké válnak, újabb és újabb források állnak rendelkezésre annak, aki tanulni szeretne belőlük.

A Jason Hunter (O'Reilly and Associates kiadónál napvilágot látott Servlet Programming szerzője) által vezetett ☞ [servlets.com](http://servlets.com) nevű honlap sok fontos adatot tartalmaz a servletekről, a legtöbb igen hasznos lehet JSP készítő számára is. Talán még érdekesebb, hogy található itt két cikk a „JPS-k gondoljai” témakörben. Bár néhány pontban nem értek egyet, kétségtelenül van ott néhány ötlet és megoldás, amit érdemes megfontolni, különösen, ha nem vagyunk biztosak benne, hogy JSP-eket vagy inkább servleteket használjunk-e.

Nemrégiben jelent meg az O'Reilly and Associates kiadásában Hans Bergsten kitűnő, JSP-kről szóló munkája, mely meglepő módon a „JavaServer Pages” címet viseli. A saját példányomat cikkem leadása előtt néhány nappal kaptam meg, és a mű teljesen lenyűgözött mélységével, világosságával és részletességével.



## Repíts engem a boros pincébe!

Marcel bemutatja, hogyan lehet a Flight Gear szimulátorprogrammal felszállni.

**N**em rossz, ugye, François? De hogy érted azt, hogy minek? A Linuxvilág e havi száma éppen az oktatással foglalkozik, én pedig most pontosan ezt a témát tanulmányozom. De hiszen ez egy repülőgépszimulátor, nemde? Senki sem szólt arról, milyen képzést és bizonyítványt kellene kitalálnom. Istenem, csodálatos pincér vagy, drága barátom, viszont olyan földhöz ragadt tudsz lenni. Nézd csak a hordók sorát, François! Nem csodálatos? Micsoda? Á igen, már itt is vannak a vendégeink. Isten hozott benneteket, drága barátaim! Helyezkedjete el kényelmesen! Az ímént éppen a Linux-rendszeretek segítségével kivitelezett repülésoktatásról és az ahhoz kapcsolódó bizonyítványokról beszélgettünk. A képzetelbeli repülőgépszámnyakat természetesen már a számítógép mellett ki lehet terjeszteni, ha azonban valóban repülni szeretnétek, akkor fel kell keresni a helybéli repülő iskolát és beiratkozni, ahogyan azt a ti főszakácsotok is tette. Bizony, barátaim, jó néhány éve kimentem a helyi repülőtérré és megtettem az első lépéseket a repülési engedélyem megszerzéséhez. Csodálatos élmény a repülés, ezt bátran ajánlhatom mindenkinek. A legtöbb repülőiskola csupán egyszerű bemutatórepülést kínál, hogy ízelítőt adjon az érzésből, mit várhat az ember a felhők között járástól. François, miért ácsorogással töltöd az időt? Gyorsan, siess a borpincébe! Tekintettel a szóban forgó téma nemzetközi természetére, azt javaslom, hozz fel egy palack 1997-es évjáratú chilei Petite Syraht a vendégeinknek. Ha a világ megnyílik neked, úgy helyes, hogy te is nyitott legyél a világra, nem igaz? A pilóta számára szükséges képességeket úgyis a pilótaülésben kell csiszolgatni, viszont addig is használhatjuk Linux-rendszerünket a tanulási folyamat elősegítésére, például letölthetünk és kipróbálhatunk egy programot, mellyel előmozdíthatjuk repülési terveink megvalósulását.

Az egyik repülőtérről eljutni a másikra többet jelent az utak egyszerű követésénél. Mindjárt elsőként azért, mert út sincs. Nahát, François, már itt is vagy! Kérlek, tölts a vendégeinket! Miközben élvezitek a pompás Syraht, talán szeretnétek pillantást vetni a menü első fogására. De milyen fogás lesz az!

Még a bejegyzett pilótáknak is sok időt kell szánniuk készségeik és képességeik karbantartására. A polgári repülésnél ezt szimulátorok segítségével érik el, hiszen mindez élesben rengeteg pénzbe kerülne! Így természetesen sokkal olcsóbb, mint – ahogy mondani szokták – egy kört tenni az egyik 747-essel. A Linux-felhasználók számára is rendelkezésre áll már néhány repülőgépszimulátor. Mi most egyet fogunk részletekbe menően megvizsgálni. A Világhálón talán a legnagyobb program a Flight Gear. E csodálatos program fejlesztői mély benyomást keltő szimulátort alkottak. Maga a helyszín lélegzet-elállító, az ég és föld színezése a fényképhűség határán mozog. A Flight Gear program szerzői annyira komolyan veszik a fényképhűség fogalmát, hogy vállalták az FAA-bizonyítvány és -tanúsítvány megszerzésének fáradságos folyamatát is. Ámde az említett valóság-hűség költségekkel jár. Bármilyen 3D-s képernyővezérlő-kártyát nélkülöző gépen a Flight Gear futtatására tett próbálkozás nem lesz több lassú és lehangoló tapasztalatnál. Ellenben, ha szerencsések vagyunk és rendelkezünk ilyen nagy sebességű grafikus gyorsítókártyával, akkor feltétlenül ki kell próbálnunk a Flight Geart. Ne is csak rohanj, hanem egyenesen repülj a <http://www.flightgear.org/> címre, hogy letölthessd a saját példányodat. Amikor ehhez a cikkhez

letöltöttem magamnak a szimulátorprogramot, a program változat-száma a 0.7.6.-nál tartott. A telepítési folyamat előfeltételezi néhány további könyvtár előzetes telepítését, ezek a PLIB, a SimGear és a Mesa. A gépen meg kell lenniük a GLUT-könyvtáraknak is. Ezek a legfrissebb Mesa 3D grafikus könyvtárának is részét képezik, és ezzel megtakarítanak számodra egy lépést a telepítés során. A Mesát számos Linux-változat korongja tartalmazza, ezért mielőtt nekikezdenél keresgélj, először is ellenőrizd lemezeid tartalmát! A Világháló erre vonatkozó oldalaira mutató hivatkozások a cikk végén lévő *Kapcsolódó címek* részben találhatóak.



1. kép Marcel éppen megkísérli a felszállást a Torontói Nemzetközi Repülőtérről

Ennek a programnak a lelke a 3D-s képkalkotás, tehát világos, hogy a Mesával kell kezdenünk. A legfrissebb változatot a <http://www.mesa3d.org> címről szerezheted be. Amint ellenőriztük a Mesa meglétét, létre kell hoznunk a PLIB-könyvtárakat. A telepítést jobbnak találtam, mint a legutóbbi változatnál (1.3.1.). Először bontjuk ki a forráskódot egy ideiglenes könyvtárba! Erre a célra a `usr/local/src` könyvtárat szeretem használni:

```
cd /usr/local/src
tar -xzf plib-1.3.1.tar.gz
cd plib-1.3.1
./configure
make
make install
```

A fentiekhez hasonlóan ezután ki kell csomagolni és összeépíteni a SimGear könyvtárakat. E cikk írásakor a legfrissebb csomag neve `SimGear-0.0.14.tar.gz` volt. Az alább leírt módon kell felépítenünk:

```
tar -xzf SimGear-0.0.14.tar.gz
cd SimGear-0.0.14
./configure
make
make install
```



2. kép A Sikorsky gépem készülődik a felszállásra a Los Angeles-i Nemzetközi Repülőtéren

A következő lépés az alapsomag kibontása. A főkönyvtár számára az alapértelmezés szerinti hely `/usr/local/lib`, ezt a mappát választottam a tömörített fájl kicsomagolására. Meg kell jegyezni, hogy ebben az esetben nem kell fordítanunk. Ezek a programcsomagok alapállományai, amelyekben repülőgépmoდეllek, helyszínek és még sok minden más található:

```
cd /usr/local/lib
tar -xvzf fgfs-base-0.7.6.tar.gz
```

Most jön létre a Flight Gear nevű könyvtár. Végül itt áll előttünk maga a programcsomag. Bontsuk ki a Flight Gear-0.7.6.tar.gz nevű állományt és építsük fel a programot!

```
tar -xvzf FlightGear-0.7.6.tar.gz
cd FlightGear-0.7.6
./configure
make
make install
```

A számítógéphez már eleve beépített gyorsítási lehetőség kihasználásához a Flight Gear végrehajtható moduljának (fgfs) SUID tulajdonsággal (a futtató a rendszergazda) kell futnia.

```
chmod +s /usr/local/bin/fgfs
```

Amennyiben az összes telepítés hibátlanul zajlott le, máris indítható a repülőgépszimulátor az íróasztalod mellől, a kényelmes karosszékből. Töltöttem magamnak egy pohárával és beindítottam a repülőgépmoდეim hajtóműveit. Igen, barátaim, én is tudom, hogy az ember nem ihat és vezethet egyszerre, de ez mégiscsak egy szimulátor. Jól van, talán mégis inkább gyömbéret iszom.

```
fgfs --fg-root=/egy-könyvtár/FlightGear
```

Hohó, most jut eszembe, hogy használni kell az fg-root kapcsolót, ha a szimulátorprogram alapállományait (helyszínek stb.) az alapértelmezés szerinti könyvtárról eltérő helyre telepítetted. Vess egy pillantást az 1. képen látható Cessna 172-esem pilótafülkéjére! Most pedig, barátaim, beavatlak benneteket egy titokba. A ti főszakácsotok is bejegyzett pilóta, és itt, mindenki előtt kész vagyok kijelenteni, hogy a jó repülőgépszimulátor elég félelmetesnek tűnik, amikor az ember megpróbálja megérteni a működését. Amint hozzászokunk a vezérlő-

berendezések kezeléséhez – jómagam a billentyűzetet használtam a repülőgépi irányítására –, elérkezik az ideje annak, hogy úticélt keressünk magunknak. Végül is elég nagy ez a bolygó, nem igaz? Szerencsére a Flight Gear nincs híján a felkereshető helyeknek. A program honlapján további helyszíneket is találhattok. Pár kattintás a 10x10 fokos szeletekre darabolt világtérképhez repít. A magam részéről a Torontói Nemzetközi Repülőtérre szerettem volna látni, amely a w080n40.tar.gz állományban található. Ahhoz, hogy ezt használni tudjam, a Flight Gear helyszíneket tartalmazó könyvtárára kellett váltanom. Ne feledjük, hogy az alapsomag alapértelmezés szerinti telepítéséhez a `/usr/local/lib` könyvtárat használtuk:

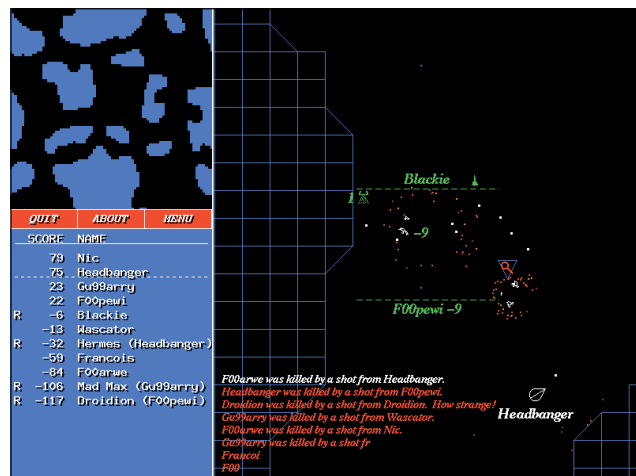
```
tar -xvzf w090n40.tar.gz
```

Ettől a ponttól újraindítottam a programot – most már a torontói repülőtérrel – megadva a repülőtér koordinátáit:

```
fgfs --airport-id=CYYZ
```

Ebből a kis Flight Gear bemutatóból sajnos nem fogunk mindent megtanulni a repülésről, de azért melegen ajánlom, szánj rá időt és szórakozz vele!

Ha inkább a helikopterekhez vonzódsz, akkor más fejlesztés alatt álló programot, a SearchAndRescue-t ajánlhatom. Nem csak kipróbálni lehet a helikopterek egész tárházát, de ezekkel kutató–mentő küldetéseket is lehet teljesíteni – innen származik a program neve –, miközben a szárnyú végzettől kell megmenteni az áldozatokat. Nézz csak a 2. képre, ahol a Sikorsky Jayhawkom a Los Angeles-i Repülőtér indítóállványán látható.



3. kép Marcel űrhajóját a csillagközi porba taszítják az xpilot programban

A program telepítésének folyamata lényegében a korábban bemutatott példakéval azonos. Bontsuk ki a SearchAndRescue0.7b.tgz forrásomagoat az ideiglenes könyvtárunkba! Fontos megjegyezni, hogy a hibátlan programfordításhoz az xvview könyvtárakra is szükség van. A hanggal és botkormányval történő vezérléshez még további programkönyvtárak szükségesek, amelyek nem szerepelnek a Világháló e lapján, ám ezekre a program működéséhez nincs is feltétlenül szükség.

```
tar -xvzf SearchAndRescue0.7b.tgz
cd SearchAndRescue0.7b/sar
make -f Makefile.Linux
```

```
make install
The default installation directory for the binary
is /usr/
```

A futtatható modulok számára a telepítés alapértelmezés szerinti könyvtára a /usr könyvtár. A SearchAndRescue program indításához géped be az alábbiakat. Figyelj a nagybetűkre!

```
/usr/games/SearchAndRescue
```

Tapasztalataim szerint a SearchAndRescue szintén meglehetősen valóságos képet ad, ezenkívül grafikus gyorsítás nélkül is elég tűrhetően játszható. Ha az égnek már minden szegletét kiismerted, akkor valószínűleg csak a csillagok maradtak elérhetetlenek a számodra. Vagy talán még azok sem? A következő héten még valószínűleg nem tudod bejárni az egész Világegyetemet. François éppen azon dolgozik, hogy kiszűrje a hibákat a maga fejlesztette „Gyorsabban a fénynél” nevű programból. Lazíts, François, ez csak tréfa! Mindamelllett élvezheted a csillagközi repülést (vagy akár a harc) izgalmát a Linux-rendszered egyik leginkább magával ragadó játékaival – egy olyan játékkal, amelyről esetleg nem is tudod, hogy már a te Linuxodban is szerepel. A játék neve xpilot, és ha netalán még nem lenne telepítve a gépeden, keresd a telepítőlemezeken. Nos, ha ott sem találnád, akkor irány a <http://www.xpilot.org/> cím, ahonnan letöltheted magadnak. Ha érdekel, itt az xpilot új felhasználói számára összeállított kézikönyvbe is bepillantasz. Mielőtt azonban továbblépnénk, úgy érzem, figyelmeztetnem kell téged, hogy a borhoz hasonlóan, az xpilotot is csak mértékkel szabad „fogyasztani”. Éppen csak kiejtettem e szavakat és máris elérkezett a játék ideje. Használd a teln programot, hogy megkapd a kiszolgálók listáját, amelyekhez csatlakozhatsz:

```
teln meta.xpilot.org 4401
```

Figyelj a parancs végén levő kapuazonosítóra: 4401! Előfordulhat, hogy állományba szeretnéd irányítani a parancs kimenetét: `teln meta.xpilot.org 4401 > xpilot.servers`, különben sok adat kifut majd a képernyőről. A telnben a 4402-as kapun megkaphatjuk a játékra vonatkozó gyakran ismétlődő kérdéseket. A 4400-es kapun keresztül interaktív kapcsolatot kezdhetünk a kiszolgálóval. Innen segítséget is kérhetünk (Help), listát (List) készíthetünk, vagy hozzáférhetünk a gyakran ismétlődő kérdésekhez. Az alábbiakban bemutatom, hogy hogyan néznek ki ezek a listák.

```
4.2.1 :marach.dd.chalmers.se :15345 :3:26d
      ↪:Tourmination
4.2.1 :tigger.skaro.bt.co.uk :15345 :1 :0.03
      ↪:The Globe
```

Az első rész a program változatszáma, amelyet a számítógépnév, majd a kapuazonosító követ, végül a számítógép folyamatos üzemben töltött idejét és a választott felhasználói nevet látjuk. Ez ám az igazi! Ezek a játékok mind élő játékok, vagyis az egész világon valós időben történik a programvégrehajtás! Sőt, még azoknak sem kell aggódnia, akik a nagy sebességű 3D-s grafikus gyorsító-kártyák hiánya miatt mellőzve érzik magukat. Ha fut az X környezet a gépeden, akkor játszhat az xpilottal, és összemérheted tudásodat a földkerekség legjobb pilótáival. Válasszuk a Tourmination nevű résztvevőt, minthogy máris hárman jelentkeztek be a játékba, és kapcsolódjunk be mi is! Ezt a következő paranccsal tehetjük meg:

```
xpilot -name francois -join marach.dd.chalmers.se
```

Becenevet választok magamnak a -name kapcsolóval a játék idejére, amely csak néhány pillanatig tart, mivel valaki eléggé aljasan a csillagközi porba taszítja a harci repülőgépet. Amint bizonyára el tudod képzelni, rengeteg tulajdonságot lehet beállítani. Ezek lekéréséhez géped be az `xpilot -help` paracsot. A 3. képen pedig a rövid úrcsatámról készült képet láthatod.

Tessék, François? Már ilyen késő van? Mielőtt elmentek, hadd adjak nektek még egy utolsó repüléssel kapcsolatos ajándékot. Valószínűleg mindnyájan ismeritek a képernyővédő csomagot a tűzijátékkal, a cikázó vonalakkal, a fraktálokkal és az űrutazást, amelyben sziklák száguldoznak a világűrön át, a képernyő meg ide-oda himbálódzik. Ha ez még nem lenne telepítve, akkor itt is a Linux-változatod telepítőlemezeit ajánlom figyelmedbe, vagy nézd meg a hivatkozásokat a *Kapcsolódó címeknél*. Jó, akkor most nézz körül a házban és ásd elő a régi jó 3D-s szemüvedet! Tudod azt, amelyekben piros meg kék (vagy zöld) lencsék vannak. Rajta, tedd fel a 3D-s szemüvedet, csatold be a biztonsági övedet és géped be a következő paracsot:

```
/usr/X11R6/lib/xscreensaver/rocks -3d
```

Ugye egészen olyan, mintha ott lennél? Talán nem is tudtad, de ezeket a programokat mind külön-külön is végre lehet hajtani – amint ezt a szikláknál is láttuk –, és meg sem kell várnunk, amíg bekapcsol a képernyővédő. Egylőre ennyi, kedves barátaim.

François, legyél szíves, tölts még egy utolsó pohárával, mielőtt elmennek a vendégeink! Kedves barátaim, be kell vallanotok, hogy tényleg élvezetes olyan operációs rendszert birtokolni, amely „képes repülni”. Mi több, ezzel ti magatok is tudtok repülni. Csak tréfáltam, de jó tréfa, igaz? François, barátom, tölts még az én poharamba is, légy szíves! Legközelebb is találkozunk itt a Chez Marcelnél! Az asztalunk már vár bennünket. Egészségetekre! Jó étvágyat!



*Marcel Gagné* (mggagne@salmar.com) Mississaugaban (Ontario, Kanada) él, a Salmar Consulting Inc. cég elnöke. A cég rendszerépítéssel és hálózati tanácsadással foglalkozik. Marcel egy személyben pilóta és író tudományos-fantasztikus regényeket ír, és társszerzője egy sci-fi, fantázia és horror antológiának, a TransVersionsnak. Kedveli a Linuxot, a Unix minden változatát. Mostanában a *Linux System Administration: A User's Guide* című művén dolgozik. A világhálón elérhető honlapján sok hasznos dolgot találhatunk.

## Kapcsolódó címek

FlightGear ➔ <http://www.flightgear.org/>  
Mesa 3-D Libraries ➔ <http://www.mesa3d.org/>  
Search and Rescue  
➔ <http://wolffpack.twu.net/SearchAndRescue/>  
The WINE Headquarters ➔ <http://www.winehq.com/>  
The XPilot Web Site ➔ <http://www.xpilot.org/>  
XScreenSaver Web Site  
➔ <http://www.jwz.org/xscreensaver/>  
The XView Toolkit  
➔ [http://step.polymtl.ca/~coyote/xview\\_main.html](http://step.polymtl.ca/~coyote/xview_main.html)

## SimCity 3000 Unlimited Linuxra

**A**zt hiszem, majdnem mindenki, aki szereti a számítógépes játékokat, találkozott már a SimCity valamelyik változatával – bár némelyek számára nem túl csábító egy város feletti istenszerű hatalom a kedvenc lövöldözős játékanak vérző-néhez képest. Ugyan miért fárasztaná magát bárki is azzal, hogy megszervezze a város működését és gazdaságát, ha könnyűszerrel kiiktathatja a rosszfűkát plazmafegyverével bármelyik ürarénában? Ha azonban leülünk játszani valamelyik SimCity-változattal, fél nap múlva a játék gerjesztette kábulatból arra ocsúdhatunk, hogy negyedmillió boldog ember lakta város polgármestereként ünnepelnek. Számunkra ezek, a felülről lefelé alakuló, széles látószög „világszimulátor” játékok kábítószerré válhatnak, és nincs többé menekvés! A SimCity 3000 Unlimited – e műfajban a legjobbak legjobbika – különösen veszélyes.

### A történet

A SimCity 3000 Unlimited a legutóbbi folytatása annak a játéksorozatnak, ami a Maxis zászlóshajójává vált, és az első, amely megjelent Linuxon. Önmagában nincs igazi cselekményvonala, bár lehet rövid küldetéseket is játszani különleges, elérendő célokkal. A legerősebb SimCity-szenvedély abból a hőstett érzésből fakad, amelyet a „vég nélküli játékok” nyújtanak azáltal, hogy egy üres térképre egy egész világvárost lehet építeni. A játékot kevéske pénzzel és egy telekkel kezdheted, ez utóbbit a saját ízlésed szerint módosíthatod. Ezután elkezdheted építeni a városodat: ki kell jelölnöd a kívánt típusú fejlődési övezeteket, meg kell szervezned egy sokoldalú mérnöksapatot, akik elvégzik a városvezetési feladatokat, úgymint a költségvetés egyensúlyban tartását, a városlakók megadóztatását, a határozathozatalt stb. Ahogy egyre több bevétel lesz az adóból, a városod ezzel arányosan nő (több pénz lesz, úgy fejlesztetd a városodat, hogy vonzza az embereket, több ember költözik be, több a bevétel az adóból). Más, közeli városokkal kereskedve, olyan kapcsolatokat alakíthatsz ki, melyek elősegítik városod fejlődését. A játékmenet e módjában, a játék elsősorban akkor ér véget, ha úgy gondolod, hogy megvalósítottál minden célt, amit kitűztél magad elé. Aztán újból előlről kezdheted az egészet, felhasználva azokat a tapasztalatokat, amelyeket az előző városokból szereztél, és egy új, sokkal jobb várost építhetsz.

A SimCity3000 Unlimitedben végre ismét lehet úgynevezett „scenario”-kat (előre megírt forgatókönyveket) játszani. Ezeket a SimCity2000 tartalmazta, viszont a SimCity3000 első (a nem Unlimited) kiadásából bosszantó módon hiányoztak. Ebben a tizenhárom új forgatókönyvben alapvetően olyan helyzetekben találhatod magad az elkészített városokban, ahol kiváló vezetői képességeidet

felhasználva kell bizonyítanod. Például a „Criminalville” forgatókönyve szerint csőcselék lepte el Moszkvát, és az a feladatod, hogy rendet rakj a zűrzavarban, és mindenkinek megadj mindent, amit csak kíván. A „Fal leomlásában” Kelet- és Nyugat-Berlint kell egyesíteni,

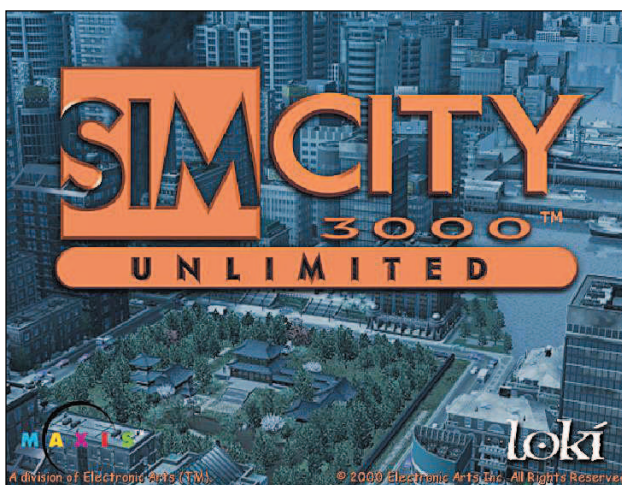
és helyrebillenteni a gazdasági egyensúlyt. A forgatókönyvek egy része meglehetősen könnyű, hozzávetőleg 45 perc alatt végig lehet játszani, viszont a többi olyan nehéz, hogy kedden hajnali négy órakor családottan harapsz az ajkadba, hogy még mindig nem értél a végére.

### A játék menete

Mint ahogy azt elvárhatjuk egy olyan játéktól, melynek témája egy hatalmas világváros pénzügyi és politikai irányítása, a SimCity 3000 Unlimited játékmenete első pillantásra meglehetősen bonyolult. Ha már játszottál valamely SimCity-változattal, akkor ismerős lesz a játék hangulata, és bár a kezelőfelületet itt-ott továbbfejlesztették, egy veterán játékosnak természetesnek fog tűnni ez a változás. Ha még soha nem játszottál a SimCity-vel (Melyik bolygóról származol?), akkor első pillantásra a vezérlőeszközök száma soknak tűnhet.

Azonban csekély szorgalommal is olyan természetessé válik majd a játék, mint a lélegzvétele. A Maxis nagy erőfeszítéseket tett azért, hogy a SimCity kezelőfelülete olyan felhasználóbarát és érthető legyen, amennyire csak lehetséges.

A játék elején a körülbelül száz négyzetkilométernyi terület rengeteg jellemzőjét te állíthatod be. Mostantól nem vagy arra ítélve, hogy egyszerű, véletlenszerűen előállított pályán kezdj. A SC3U-ban továbbá meg lehet határozni az általad választott térkép minden jellemzőjét (hegyeket, tavakat, erdőket vagy bármi mást), vagy választhatsz olyat is, amely valamelyik híres város térképén alapul (Bonn, London és Boston). Ezenkívül azt is eldöntheted, hogy milyen legyen a táj a kiválasztott térképen (a sivatagostól a havasig), milyen legyen az épületek stílusa, illetve az öt lehetséges változat közül milyenfajta növényzet burjánozzon rajta. A rács megjelenítése is igény szerint állítható. Nem csak az alapépületeket választhatod meg, hanem az SC3U's Building Replacement Manager segítségével lecserélheted a régi épületeket új, általad választottakkal. Nem tetszenek a látszólag végtelen számú, előre elkészített épületek? Használd a SimCity 3000's Building Architect Plus-t, hogy a régieket megváltoztasd, vagy készíts újakat a semmiből a BAP 3-D fejlesztőkörnyezet segítségével. De szögezzük le rögtön, ez nem maga a játék, hanem csak egy lehetőség, hogy még érdekesebbé tegyük azt! Bár az összetettség ilyen magas foka nem mindenkinek tetszik, jó tudni, hogy adott a lehetőség!



### Adatok

Gyártó: Loki Entertainment Software  
 E-mail: sales@lokigames.com  
 URL: <http://www.lokigames.com/>  
 Ár: kb. 10000 Ft

© Kiskapu Kft. Minden jog fenntartva



1. kép A Maxis nagy erőfeszítéseket tett azért, hogy a SimCity kezelőfelülete olyan felhasználóbarát és könnyen érthető legyen, amennyire csak lehetséges



2. kép Teljes ellenőrzés: az SC3U terepszerkesztője

A játék valós időben játszható (bár meg is lehet állítani). Te készítesz és irányítasz mindent: a majdan híres épületek építésétől kezdve a közlekedés (metró, vasút, közút, repülőtér, kikötő stb.) megszervezésén, a rendőrség fenntartásán és irányításán, valamint a városod vízrendszerének vezérlésén át a szomszéd városokhoz való megfelelő kapcsolat kialakításáig.

Feladataid közé tartozik ezenkívül a környezetszennyezés megfékezése, illetve foglalkoznod kell a városi hulladék elhelyezésével is. Megfelelő övezetekre kell osztanod a várost, hogy a lakosoknak legyen elegendő ipari létesítményük. Továbbá kereskedelmi és üzleti negyedeket is kell építened, hogy nehezen megkeresett pénzüket el is tudják költeni. Mindennek természetesen egyensúlyban is kell maradnia. Az összes övezetnek három típusa van: kis, közepes és nagy népsűrűségű, ezeknek hasonlóan kell lennie az általad építeni kívánt üzleti, illetve lakóhely-típusú övezeteknek az adott területen. Felügyelned kell az adóztatást is, de nem ez az egyedüli bevételforrásod. Megfelelő rendeletek kiadásával pénzhez is juthatsz (például a szerencsejáték és kaszinók engedélyezésével), és még lakályosabbá teheted a városodat. Különleges üzleti szerződéseket is köthetsz a szomszéd városokkal, ez szintén pénzt hoz a konyhára. Vízrendszert többet termel, mint a városod fogyasztása? Miért nem adsz el ebből a vízből a szomszédaidnak?

Hogy a játék folyamán a legfrissebb adatok eljussanak a polgármes-

terhez, a képernyő alján található táviró folyamatosan kopogja a híreket. Ez tulajdonképpen üdvözlendő változás az előző változatok bosszantóan előugró újságpapír-stílusú adataihoz képest. Tanácsadóik állnak rendelkezésedre, akik időről időre szabadon tudatják veled a véleményüket, és akikkel megtárgyalhatsz a gondjaidat, ha bizonytalan vagy a dolgok folytatását illetően.

Az SC3U grafikája szuper! Minden épület tulajdonképpen 3D környezetben készült, pontosan kidolgozott részletességgel. Az épületek több mint négyszáz különböző fajtájával találkozhatasz (százakkal



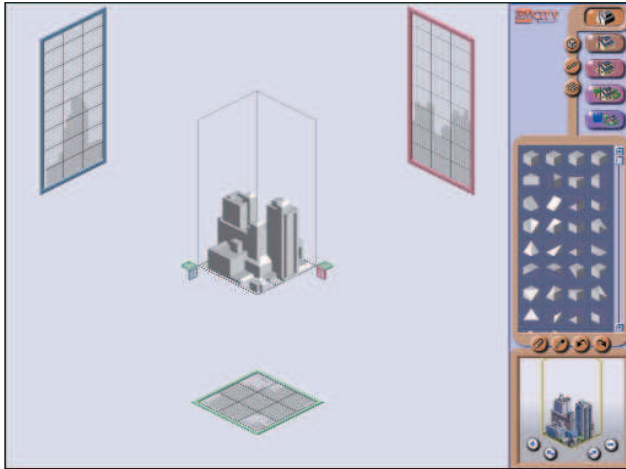
3. kép Tanácskozás a tokiói költségvetés kiegyensúlyozásáról

több, mint az előző SimCity-változatokban). Mi több, ha úgy gondold, játék közben egyetlen egérgattintással átválthatsz gyökeresen eltérő építészeti stílusra (mondjuk európairól ázsiaira)! Több tucat, a valóságban is létező látványosságot építhetsz, például a World Trade Centerrel, az ENSZ épületével vagy az Eiffel-toronyral csinosíthatod a városodat. A játékban az animációk lélegzetelállítón szépek. A Hálaadás ünnepi díszfelvonulás választható elem, és ha a rákattintás után ráközelítesz, ámulatba fog ejteni a látvány: emberek mindenhol, menetelő zenekarok, ünneplők kis autókban, fények, sőt, óriási léggömbök. Mindenki bolyong az utcákon fel s alá. A részletek kidolgozottságának e foka tényleg a fő oka annak, hogy az SC3U ilyen jól néz ki.

### Saját készítésű épületek

A SimCity 3000 Unlimited testreszabhatósága itt természetesen nem ér véget. Az SC3U-felhasználók ezrei mindenhol a világon rendszeresen készítenek épületeket, forgatókönyveket, neves épületeket és városokat, melyeket beilleszthetsz a játékba. A Maxis sajnos úgy döntött, hogy nem támogatja ezen kiegészítőket hivatalos kereskedelmi felületén, a SimCity Exchange-en. Csúnya Maxis, nem szeretsz bennünket? Szerencsére, a Loki biztosít egy olyan választható listát, ahol elérhetjük a testreszabhatóság ugyanezen fokát (anélkül, hogy szükségünk lenne a Redmondból származó operációs rendszerre). Elérhető a <http://faqs.lokigames.com/faq.php3?view=index&product=SC3K> címen.

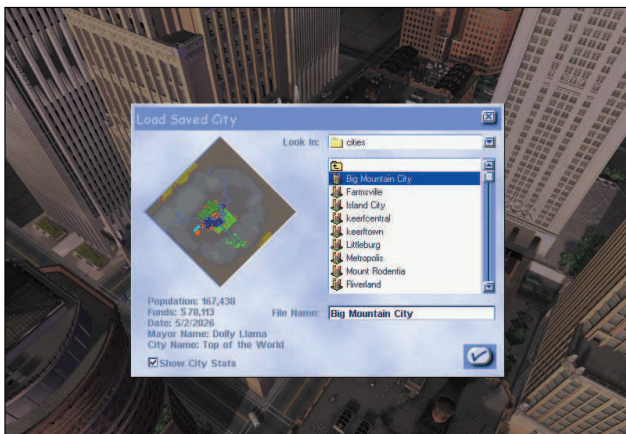
Ez a kitűnő forrás választ ad még tucatnyi SC3U-val kapcsolatban felmerülő kérdésre, mint például hogyan állítsuk be az esound-ot, vagy miként lehet autópályafeljárót készíteni.



4. kép Saját épületek a Building Architect Plusban



5. kép Tanácsok játék közben



6. kép Várnak az elmentett városok

Úgy találd, hogy a városod már elég jó és más kihívásra vágysz? Miért nem szabadítasz rá a városodra valamilyen csapást, hogy utána rendet rakhass a zűrzavarban? Négy olyan új szerencsétlenség van az SC3U-ban, ami az előzőekben nem szerepelt, valamint megtalálhatók a régi kedvencek gyűjteménye is, a zavargástól a sáskajáráson keresztül, az Armageddon-szerű űrszemétzáporig. Ha érdekesebbé akarod tenni a játékot vagy csak rombolhatnéd van, a Maxis emberei

megadják az eszközöket, hogy ezt is megtehesd. A zene, nos igen, ez a Maxis, s ahogy lehetett rá számítani – hátborzongatóan jó. Lány és nyugtató és hosszabb idő alatt hajlamos hihetetlen boldogságot árasztó, önkívületszerű állapotba ringatni, de sajnos egy idő után unalmas. Időről időre azon kaptam magam, hogy kikapcsoltam a hangszórókat, hogy összpontosítani tudjak. Természetesen ez nem azt jelenti, hogy rossz a zene. Ez nagyon jól megkomponált muzsika, de minden zene unalmassá válhat félnapos SC3U játék alatt.

### Linuxos megjegyzések

G400-as kártyával szerelt 500 MHz-es, Pentium III-as processzorral és 128 MB memóriával szerelt VA munkaállomáson, valamint egy GeForce kártyával szerelt 550 MHz-es Pentium III-as processzorral és 256 MB memóriával rendelkező gépen próbáltam ki a játékot, és mindkét rendszer kisebb nehézségekkel futatta. Bár a játék nem követel semmilyen 3D-s támogatást, egy száz négyzetkilométeres területen fekvő város szimulálásának számításai meglehetősen nagy igénybevételnek teszik ki a gépet, és ha a fejlesztés magasabb fokára lépsz, a képernyő mozgatásának és a nagyításának lassulása kezd egyre észrevehetőbbé válni. A Loki legkisebb gépkövetelményként 233 MHz-es processzort és 32 MB memóriát jelöl meg, miközben ezeknek a számoknak csak egy aprócska városkánál van értelme. Azt hiszem, hogy a profi felhasználók (vagy az igazán óriási városok építői) majdnem lehetetlennek fogják találni egy nagyváros megépítését a számítási idő korlátai miatt. A Loki azért javasolja a játék 16 b/p színmélységben való futtatását, mert így elkerülhető a futásidejű átalakítás, amit a 24 b/p vagy 32 b/p színmélységben való futtatás okozna. Ha használni akarsz a Building Architect Plus programot, szükséged lesz egy Open-GL-megfelelő 3D meghajtóprogramra (sem az én nVidia OpenGL meghajtómmal, sem a Mesával nem volt gondja a BAB-nak). Szükséged lesz egy OSS-megfelelő hangkártyára és legkevesebb 450 MB területre a merevlemezben (a teljes telepítés több mint 650 MB). Általánosságban a játék gépigénye kisebb, mint amit nyújt futás közben. Gyakran futtattam a kis, 500 MHz-es laptopomon és ugyanolyan játszhatóan találtam, mint az asztali gépeimen.

### Összegzés

Ezidáig ez a legjobb SimCity-változat, nagyfokú testreszabhatósággal, sok kiegészítővel, számos szolgáltatással, szebb grafikával és elődeinél jobb játszhatósággal. A játék jóval összetettebb lett. Ha egyszerű játékot keresel, akkor valószínűleg ez nem lenne jó választás számodra. Ha azonban tetszik a tervezés és a haditervek kovácso-lása, ráadásul élvezed a szórakoztató szellemi kihívást, akkor ez a te játékod! Véleményem szerint, ennek a játéknak a kipróbálása kötelező minden Linux-felhasználó számára.



J. Neil Doane  
(caine@valinux.com) a VA Linux Systems mérnöke. Ha gép- és videojáték-hóbortja engedi, szívesen vezet repülőket, emellett gitározik és újabban hódesházik is (ez utóbbiban még igencsak gyengécske).

### Előnyök

- Rendkívül rabul tud ejteni.
- Testreszabhatósági foka magas.
- Kítűnő a grafika.
- Bővíthető Interneten fellelhető forrásokon keresztül is.



### Hátrányok

- Ennek a SimCity-változatnak nincs kedvezőtlen tulajdonsága!
- Komolyan, ez a játék üt!
- Oké, talán egy kicsit bonyolult egyeseknek.



## Előkészületben

Június elején jelenik meg Brian Ward könyvének magyar fordítása A Linux hibaelhárító címmel.

Mivel is kezdjem e nagyszerű és valóban hiánypótló könyvecske ismertetését? Talán azzal, hogy a Linux egyre többek számára (és most a Windows-felhasználókra is gondolok) az operációs rendszerek Volkswagen Bogarává válik: a vállalati webkiszolgálóktól kezdve az otthoni tűzfalon át az asztali munkaállomásig rengeteg mindenre használják. Összetettebb rendszer, mint a Windows? Bizonyos szempontból igen, másrésztől nem. Nagyobb türelemre van-e szüksége a kezdő felhasználónak ahhoz, hogy hatékonyan használhassa? Nos, azt kell mondanom, hogy igen. És nemcsak türelemre, hanem ehhez hasonló könyvekre is. „Megoldások rendszergazdák számára” – olvasható az ajánlószövegben. Mielőtt a Linuxsal most ismerkedők egy „na, megint egy könyv, amiből egy kukkot sem fogok érteni” sóhajtással visszahelelyeznék a könyvet a polcra, végezzünk gyorsan egy kis jelentéstani elemzést. Kik a linuxos rendszergazdák? Semmiképpen sem csak a vállalat kis kuckóiban a termináblak fölé görnyedve kólát szopogató programozók, informatikusok. Nem is csak az egyetemi hálózat rettegeve tisztelt félistenei, akik néha (csokival vagy más, szénhidrátartalmú eledellel történő bundázás hatására) kegyeskednek a felhasználói könyvtárunk méretét öt megával megoldani. Korántsem ilyen egyszerű a helyzet. Ha Linuxot telepíték az otthoni gépemre, akkor én is rendszergazda vagyok. Ez világos. És az az ismerősöm is rendszergazda, aki a lakásában lévő kábelnetes előfizetésének bekötése után azonnal felállított egy Linux-alapú tűzfalat a külvilág és mezei Windows 98-as gépe között. Egy rendszergazdának pedig illene a lehető legtöbbet tudnia az operációs rendszeréről. Hogyan is kell a Sambát beállítani? Mert hát emlékszem én rá, hogyne emlékeznék, de a biztonság kedvéért jó lenne valahol leellenőrizni, hogy jól tudom-e. A startx meg az xinit melyik beállításfájlt is olvassa be? Az .xinitrc-t? Vagy azt mégis az xdm használja? Hm, újra kéne fordítani a rendszermagot, mert ennek a 2.2-nek már egy kicsit porszaga van. Hogyan is kezdjek

hozzá? Természetesen, csináltam már ilyet, de miképp van *pontosan*? Ilyen és ehhez hasonló „életbevágóan fontos” témák, megoldandó feladatok gondosan szerkesztett gyűjteménye a könyv, mely nyugodt szívvel ajánlható azoknak is, akik még életükben nem láttak Linuxot működés közben. A kilenc fejezet valóban a Linux használatának legkényesebb területeit érinti.



Az első, bevezető jellegű fejezet a rendszer indulásával és futásával kapcsolatos alapvető tudnivalókat, valamint a rendszermag végzett munka néhány segédeszközének, például a rendszernaplónak a használatát ismerteti. A második fejezet a hálózat tökéletes beállításában lehet segítségünkre. A harmadik fejezet az adatok hálózaton történő továbbításával foglalkozik. A tárgyalt témák között találjuk az NFS-t (ez a Network File System, azaz a hálózati fájlrendszer rövidítése), a NIS-t, amely viszonylag egyszerű hálózati adatbázisrendszer, valamint az rdist-et. Ez utóbbi program fájlcsoportokat többszörös két gép között. A negyedik fejezet a Linux és más operációs rendszerek, például a Windows vagy a MacOS közti adatcseréről szól. A fejezet utolsó részében a szerző egy proxykiszolgálót ismertet, melynek segítségével tovább fokozhatjuk hálózataunk teljesítményét. Az ötödik fejezet a rendszer – és a rendszergazda – szemszö-

géből elemzi a nyomtatási feladatokat, a Unix és a Linux nyomtatási folyamatának működését, a nyomtató által fogadott formátumtól kezdve a továbbítandó adatokat feldolgozó szűrőkig. A hatodik fejezetből a Unix programok hagyományos, forráskódból történő telepítését sajátíthatjuk el. A hetedik fejezet teljes egészében az új Linux rendszermag és az ahhoz tartozó modulok telepítéséről szól. A nyolcadik fejezet mindenki kedvenc témájával, a katasztrófa-elhárítással foglalkozik. Ha fontos adatokat is tárolunk rendszerünkön, biztonsági másolatokat kell készítenünk, és a fejezet ebben próbál segíteni. A könyvhöz mellékelt CD-ROM-ról történő rendszerindításhoz is itt találunk tanácsokat. Végül a kilencedik fejezet az új felhasználói fiókok létrehozását és kezelését ismerteti. Azt bizvást állíthatom, hogy magyar nyelven kevés ilyen jó szakmunka jelent meg e témában, és a Hibaelhárító még ezek között is igen előkelő helyet foglal el nálam. Közhelynek számít, ha egy könyvről azt mondják, hogy a kezdők és a profik egyaránt haszonnal forgathatják, de remélem, hogy a Hibaelhárító lapozgatva mindenki hasonló véleményre jut. Én személy szerint megszerettem. Kell nekem ez a könyv!



Borai János  
(borai.janos@linuxvilag.hu)  
az ELTE amerikanisztika  
szakos hallgatója,  
1997-ben ismerkedett meg  
a Linuxsal. Szabadidejében zenél, jelenleg egy otthoni stúdió  
kiépítésén fáradozik.

Szerző: Brian Ward  
Terjedelm: 300 oldal  
Ár: 3220 forint  
Kiadó: No Starch Press, 2000.  
Magyar kiadás: Kiskapu Kft., 2001.  
E-mail: kiskapu@kiskapu.hu  
Honlap: www.kiskapu.hu  
ISBN: 963 9301 24 8