



használatával fűzzem be. Hogyan érhetném el, hogy az `ide-scsi` modul a 2.4 rendszermag alatt is csak a `/dev/hdc`-t használja?

Michael Soulier, michael.soulier@home.com

Ha jól értem, azt szeretnéd, hogy a `hdc` SCSI-utánzást használjon, míg a `hdd` továbbra is IDE-eszköz maradjon. Általában az IDE CD-támogatása tiltott, a SCSI-utánzás pedig engedélyezett, ezért látszik mindkettő SCSI-eszközként. Olvasd el a

☞ http://www.wizball.co.uk/linux/cd_rewriter.php és a
☞ <http://www.teknoospy.com/pages/howtos/cdburn.php> oldalakon található leírásokat.

Paul Christensen, pchristensen@penguincomputing.com

Merevlemez beépítése

Beépítettem egy második merevlemezt a gépembe, fel is osztottam lemezzszekre, de nem tudok rajta fájlrendszert létrehozni.

Kevin Williams, williams_kevin@btconnect.com

Először is győződj meg róla, hogy melyik eszközlől van szó. Az alábbi egyszerű lista az IDE-felületű egységeket foglalja össze, talán segít az eligazodásban:

```
/dev/hda1 az elsődleges csatolón lévő mester
(primary master) első lemezzsze
/dev/hda2 az elsődleges mester második
lemezzsze
/dev/hdb1 az elsődleges szolga (slave) első
lemezzsze
/dev/hdb2 az elsődleges szolga második lemezzsze
/dev/hdc1 a másodlagos csatolón lévő mester
(secondary master) első lemezzsze
/dev/hdc2 a másodlagos mester második
lemezzsze
/dev/hdd1 a másodlagos szolga első lemezzsze
/dev/hdd2 a másodlagos szolga második lemezzsze
```

Most válaszd ki a használni kívánt fájlrendszert. A SuSE-terjesztésben a ReiserFS található – én ezt javaslom, mivel használata esetén egy hibás rendszerleállítás utáni újraindításkor nem kell kivárnunk a hosszadalmas ellenőrzést. Ezt követően formázd meg a lemezt. A ReiserFS lemezzsék létrehozásához szükséges parancs az `mkreiserfs` `<eszk znØv>`, ahol az `<eszk znØv>` a lemezzsék neve. Ha ragaszkodsz az `ext2`-höz, ugyanezt a parancsot használd, de az `mkreiserfs`-t cseréld le `mke2fs`-re. Létrejött tehát egy használható lemezzsék, melyet már csak be kell fűznöd. Válassz vagy hozz létre egy befűzési pontot. Ebben a példában én a `/mnt/storage` könyvtárat használok. Hozd létre ezt az `mkdir /mnt/storage` parancssal. Amint láthatod, a befűzési pont igazából egy könyvtár. Most fűzd be a meghajtót:

```
mount <eszk znØv> /mnt/storage -t
↳ <fÆjlrndszer_t pusa>
```

Itt az `<eszk znØv>` a lemezzsék által használt eszköznév, a `<fÆjlrndszer_t pusa>` pedig `reiserfs` vagy `ext2`.

Most már egy második használható linuxos merevlemez is van. Csak egy lépés maradt hátra. Feltételezem, hogy az új lemezt minden rendszerindítás után használni akarod, tehát az új lemezzsék a `/etc/fstab` táblázatba is be kell vezetni. Írj ehhez a fájlhoz egy sort: `<eszk znØv> <befßzØsi pont>`
↳ `<fÆjlrndszer_t pusa> defaults 0 0`
Az `<eszk znØv>` itt is a lemezzsék eszköznév, a többi szintén egyértelmű. Ha a ReiserFS-t használod, a sor végén `0 0`, ha az `ext2`-t, akkor pedig `1 2` szerepeljen.
Ben Ford, ben@kalifornia.com

128 bites pontosság GCC-vel

Az `x1C` parancsot használom C++ programok lefordítására Unix-felületen. Amennyiben 64 bitről 128 bitre kell növelnem a matematikai számítások pontosságát, a következő parancsot alkalmazom:
`x1C128 -qldb1128 <fÆjlrndszer_t pusa> [-lm]`
A `-lm` kapcsolóval fűzöm be a matematikai programkönyvtárakat (amennyiben hiányoznak). Ezeket a számításokat ugyanolyan pontossággal Linux alatt is el szeretném végezni. Mivel próbálkozzam? Ha a matematikai programkönyvtárakat is fel kell használnom, kérem, adjátok meg, honnan szerezhetem meg őket.
Pramod, l_pramod@hotmail.com

Szerezd be a GMP-t (Gnu Math Precision)! Ez a szabad forrású programkönyvtár tetszőleges pontosságú aritmetikát valósít meg előjeles egészekkel, racionális és lebegőpontos számokkal. A ☞ <http://www.swox.com/gpm> címről tölthető le.

Mac-lemezzsék befűzése

Van egy 2001-es iMacDV számítógépem (400 MHz, 128 MB RAM), melyen Mac OS 9.1 és a Yellow Dog 2.0 fut. A Mac OS 9.1 lemezzséken lévő fájlokat sajnos nem tudom elérni. Azt hiszem, hogy a Mac OS lemezzsék a `hda1`, de lehet, hogy tévedek.

Bill MacKay, w.mackay1@ntlworld.com

Valóban tévedsz, nem a `hda1`-ről van szó. A Mac-gépek a lemezzségeket valamivel bonyolultabban tartják nyilván. A lemezzségeket a `cat /proc/partitions` parancssal nézheted meg. Remélem, sikerült könnyebbé tenni a helyzetet.

Hogyan indítsam el önműködően a webkiszolgálót?

A HTTPD a gép indításakor nem indul el magától, így nekem kell megtennem a `/etc/rc.d/init.d/` könyvtárból. Melyik beállításfájl változtassam meg, hogy ez önműködően is megtörténjen?

W. Huang, whuang53@excite.com

Add ki a `chkconfig --level 5 httpd on` parancsot. Ez azt feltételezi, hogy az 5. futáásszintet használod (GUI a RedHatben). Ha más futáásszintet akarsz, csak írd át a számot.

Ben Ford, ben@kalifornia.com

A Linux Journal honlapján számtalan gond megoldásához találhattok további segítséget. A Sunsite tükörodalait, a gyakran feltett kérdéseket és az egyéb útmutatásokat a ☞ www.linuxjournal.com honlapon olvashatjátok el. A rovatban közzétett válaszokat Linux-szakértők kis csapata készítette el. További kérdéseiteket szívesen fogadják (angol nyelven) a ☞ www.linuxjournal.com/lj-issues/techsup.html címen, ahol csak egy kérdőívet kell kitöltenetek, de a bts@ssc.com címre levelet is írhattok. A levél tárgyában szerepeljen a „BTS” kulcsszó.

Vírusellenőrzés a Squid proxykiszolgálón

A rendszergazdák természetes igénye, hogy minél több vírusellenőrzési pontot iktassanak be a számítógépes hálózatokba. Az egyik ilyen pont a HTTP-proxykiszolgáló, ugyanis ezen haladnak keresztül azok az Internetről letöltött fájlok, amelyeket a felhasználók böngészői szednek le.

A mennyiben e ponton sikerül kiszűrni a vírusos állományokat, el sem jutnak a felhasználók gépeihez. A Squid általánosan használt proxykiszolgáló, vele működik együtt a *viralator* program, amelynek segítségével elvégezhetjük a víruskeresést.

A *viralator* Perlben írt CGI-héjprogram, amely képes a bemeneti értéként megadott fájlokat letölteni a kiszolgálóra, és a letöltött állományokon egy külső vírusirtó programot futtat. Eközben a felhasználóval az ügyfélgépen futó internet-böngészőn keresztül tartja a kapcsolatot, azaz tájékoztat a letöltés menetéről, és arról, hogy vírusos-e a fájl. Nézzük meg működés közben!

A böngésző ablakában a kívánt hivatkozásra kattintva kezdjük meg a letöltést. A böngésző a letöltési kérést elküldi a távoli kiszolgálóhoz – ezt a kérést kapja el a *viralator* program (1. kép). A böngészőnek visszaküld egy „downloading...” tartalmú oldalt, ezután nyit egy kék háttérű ablakot, ahol a letöltés menetét láthatjuk – az ablak alján a vírusellenőrzés eredményével. A kék ablakban egy *Stop* gomb segítségével a folyamatot megállíthatjuk. Ha a teljes állomány a kiszolgálóra került és nem volt vírusos, előugrik a böngésző letöltési ablaka és menthetjük a fájlt. Miután az állomány az ügyfélgépre is megérkezett, térjünk vissza a *viralator* kék ablakához, és nyomjuk meg a *Close window* gombot. Ezután egy ablak tájékoztat arról, hogy a program a kiszolgálóról letölti az állományt, majd el is tűnik.

A *viralator* működéséhez szükséges programok

Vírusirtó program: a *viralator*ba nincs vírusirtó beépítve, külső programot indít el. A múlt havi számban ismertettem a Sophos sweep telepítését, a *viralator* képes vele együttműködni.

wget: a böngésző által kért állományt a *wget* programmal tölti le. Mindenképpen telepítsük, nagyon hasznos program.

HTTP-proxykiszolgáló: esetünkben a Squid, azonban most sem a telepítésére, sem a beállítására nem térek ki (lásd még *Linuxvilág* 2001. február–márciusi számának 74. oldalát). Az ügyfelek böngészőinek az Internetet a Squiden keresztül kell elérniük.

Átirányító program: ez a Squidhez intézett letöltési kéréseket egy külső programnak küldi át. A *viralator*hoz ajánlott átirányító a *Squirm*.

Webkiszolgáló: a *wget*-tel letöltött fájl a webkiszolgálón keresztül jut el az ügyfélgépre. Apache-kiszolgáló működését feltételezem, a telepítést pedig Debian Potato rendszerre írom le.

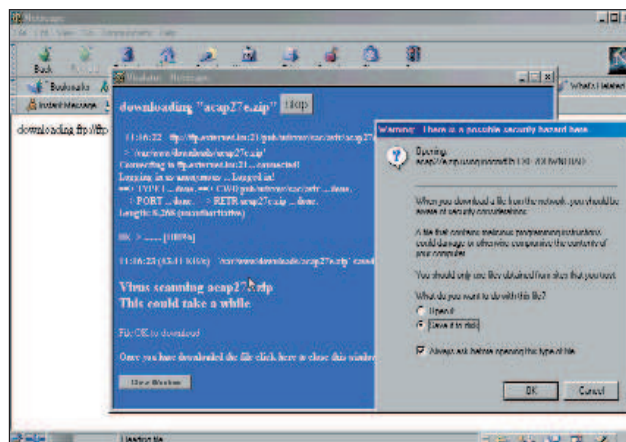
A *Squirm* telepítése

A *Squirm* honlapján a <http://squirm.foote.com.au> címen részletes telepítési leírást találunk, a forrást pedig a

```
➔ http://squirm.foote.com.au/squirm-1.0betaB.tar.gz címről
tölthetjük le. Amennyiben ezt megtettük, csomagoljuk ki a
/usr/src-be, majd lépünk be a squirm-1.0betaB könyvtárba.
Adjuk ki a
```

```
cd regex
./configure
make clean
make
parancsot. A regex könyvtárban létrejövő két fájlt az eggyel
feljebb lévő könyvtárba kell másolnunk:
```

```
cp -p regex.o regex.h ..
```



1. kép Letöltés Netscape-pel

Meg kell tudnunk, hogy a Squid milyen felhasználóként fut, amit a

```
grep cache_effective_user /etc/squid.conf
```

utasítással tehetünk meg. Debianon ez a proxy felhasználó, míg a *Squirm* a *squid* felhasználóra van beállítva. Ennek szellemében kell a *Squirm* Makefile-ját az *install* résznél módosítani:

```
install -m 755 -o root -g root
➔ -d /usr/local/squirm \
/usr/local/squirm/bin
install -m 770 -o root -g proxy
➔ -d /etc/squirm
install -m 750 -o proxy -g proxy
➔ -d /var/log/squirm
install -m 660 -o root -g proxy
➔ squirm.local.dist squirm.patterns.dist \
```

```
/etc/squirm
install -m 755 -o root -g root --strip squirm
↳ /usr/local/squirm/bin
```

A program a beállítóállományokat eredetileg a `/usr/local/squirm/etc`, a naplófájlokat pedig a `/usr/local/squirm/log` könyvtárba tette. Mivel erre nem találtam elégséges indokot



2. kép A Viralator weboldala

és zavaró is lehet, ezeket is átírtam, így a fenti mintában már `/etc/squirm` és `/var/log/squirm` szerepel. Ezután a `paths.h` fájlt is módosítani kell ott, ahol az eredeti elérési útvonalak voltak:

```
#define LOG_MATCH "/var/log/squirm/squirm.match"
#define LOG_FAIL "/var/log/squirm/squirm.fail"
#define LOG_ERROR "/var/log/squirm/squirm.error"
#define LOG_WHERE "/var/log/squirm/squirm.where"
#define LOG_DEBUG "/var/log/squirm/squirm.debug"
#define LOG_INFO "/var/log/squirm/squirm.info"

/***** Configuration file locations *****/
#define LOCAL_ADDRESSES "/etc/squirm/squirm.local"
#define REDIRECT_PATTERNS "/etc/squirm/squirm.patterns"
```

Adjuk ki a

```
make
make install
```

parancsokat és próbáljuk ki, hogy az átirányító fut-e rendszer-gazdaként a rendszerünkön:

```
/usr/local/squirm/bin/squirm
```

```
Squirm running as UID 0: writing logs to stderr
Wed Nov 21 10:55:01 2001:unable to open local
↳addresses file [/etc/squirm/squirm.local]
Wed Nov 21 10:55:01 2001:unable to open
↳redirect patterns file
Wed Nov 21 10:55:01 2001:Invalid condition
↳- continuing in DODO mode
Wed Nov 21 10:55:01 2001:Squirm (PID 24924)
↳started
```

Mivel a Squirmnek a `/etc/squirm` könyvtárban még nem készí-

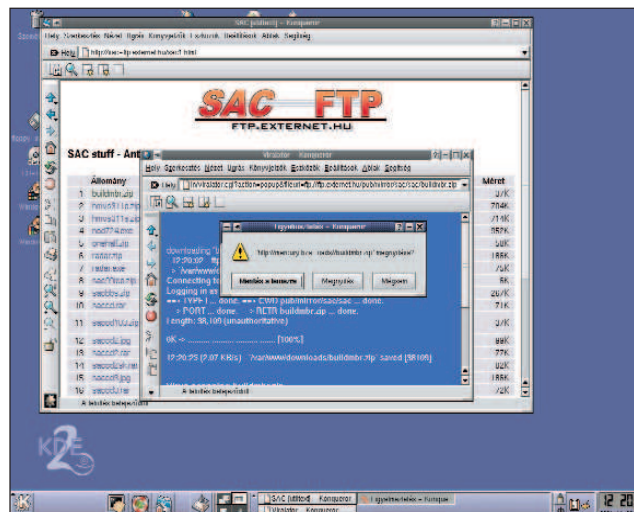
tettünk beállítóállományokat és csak dodo-módban indult el, CTRL+C-vel lépünk ki. A `/etc/squid.conf` fájlban keressük meg a `redirect_program` részt és módosítjuk:

```
#redirect_program none

redirect_program /usr/local/squirm/bin/squirm
redirect_children 10
```



3. kép Hasznos programok tárháza



4. kép Letöltés Konquerorral

A Squidet indítsuk újra. A `/var/log/squid/cache.log` állományban megjelenő

```
helperOpenServers: Starting 10 'squirm'
↳processes
```

bejegyzés tájékoztat arról, hogy a Squirm elindult.

A Squirm beállítása

A Squirm telepítésekor a `/etc/squirm` könyvtárba két mintafájl hoz létre: a `squirm.local.dist`-t és a `squirm.patterns.dist`-t. Másoljuk át őket `squirm.local` és `squirm.patterns` néven ugyanebbe a könyvtárba.