

Csomagkezelés Debian/GNU alatt

Mihez kezdünk, ha már áll a rendszer, de valamit telepíteni, frissíteni szeretnénk?

Sokan mondják, hogy „igen, igen, a rendszert még sikerült feléleszteni, de amikor új programot kellett volna telepíteni, megállt a tudomány.” E cikkben azt szeretném röviden körbejárni, miként is lehet csomagokkal bánni. Mivel a téma önmagában is igen nagy falat, nem tudok minden kérdésre kitérni, de igyekszem az általánosan előforduló feladatokat – saját ízlésem szerint – sorra járni.

Mi az a .deb?

A programcsomagok általában három változatban érhetők el: egyszerű tömörített állományként (.tar.gz vagy .tar.bz2 formátum), .rpm csomagként (eredetileg a Red Hat által fejlesztett csomagkezelő rövidítéséből), illetve .deb csomagként (a Debian-vonal által használt formátumban). A két csomagformátum külön csomagkezelő rendszerek meglétét feltételezi, amelyek a csomagokban található kiegészítő adatok alapján szükség szerint maguk végzik el a programok kicsomagolását, fordítását, beállítását és telepítését. Egyre több olyan programmal találkozhatunk, amely saját telepítőrendszerrel rendelkezik, ilyen például a StarOffice is.

Most a .deb csomagokkal foglalkozunk, melyeket több programmal is használhatunk. Kezdjük talán onnan, hogy hol találjuk meg ezeket a .deb fájlokat? Több központi ftp-terület is létezik, melyek (a cikkben csak a Debian-vonallal foglalkozom) az alaprendszerhez tartozó csomagokat tárolják. Három fő terjesztés létezik: a Potato, a Woody és a Sid, melyek karbantartása külön zajlik. A három terjesztés közül egyet mindig „megbízhatónak” neveznek ki (Stable, jelenleg ez a Potato), egyet „bizonytalanak” (Unstable), egyet pedig a próbaváltozatnak (Testing, most éppen a Woody); a Sid pedig sohasem lesz se testing, se stable, mivel Still In Development, azaz „még fejlesztés alatt” a rövidítése.

Amikor rendszert telepítünk, tudnunk kell, hogy melyiket válasszuk. Lapunk lemez mellékletére legutóbb a Woody került fel. Ha a saját terjesztésünkhöz tartozó anyagokat akarjuk megtalálni, nem mindegy, hogy melyikben keressgélünk. Igaz, az esetek túlnyomó részében a más terjesztésből származó csomagok is működnek, illetve számos csomagot külön helyekről kell beszerezni, erről azonban egy kicsit később szövegek.

APT, a kényelmes megoldás

Most egy kicsit rendbontó leszek, és előrevetsem az apt-get programot, ezzel ugyanis nagyon sok mindent meg lehet oldani. Röviden írok róla, hiszen már foglalkoztunk vele. Az apt-get felkeresi a /etc/apt/sources.list fájlban megadott helyeket, és az ott található csomagokkal dolgozik. Itt megadhatunk helyi területet (file: vagy cdrom:), webes elérést (ftp: vagy http:), illetve titkosított kapcsolatot (ssh:). Nagyon jó leírásra bukkanhatunk a man sources.list parancs kiadásával. Amennyiben az apt által elérhető csomagok listáját frissíteni szeretnénk, adjuk ki az apt-get update parancsot; ha mondjuk a Gimpet kívánjuk telepíteni, akkor az apt-get install gimp parancsot akkor, ha pedig azt szeretnénk, hogy a kezelő a telepítés óta frissült összes csomagot önműködően

1. lista Hibaüzenet

```
#apt-get install joe
Reading Package Lists... Done
Building Dependency Tree... Done
You might want to run 'apt-get -f install'
to correct these:
Sorry, but the following packages have unmet
dependencies:
  joe: Depends: libncurses5 (>= 5.2.20010310-1)
      but 5.0-6 is to be installed
  libc6-dev: Depends: libc6 (= 2.1.3-10)
      but 2.2.4-1 is to be installed
      Conflicts: libc-dev
      Conflicts: libdl1-dev but it
      is not installable
      Conflicts: libdb1-dev
      Conflicts: libgdbm1-dev but it
      is not installable
      Conflicts: libpthread0-dev but
      it is not installable
locales:E: Unmet dependencies. Try
'apt-get -f install' with no packages
(or specify a solution).
```

frissítse, az apt-get upgrade parancsot használjuk.

No, itt kezdődnek a turpisságok. Mert mi történik, ha az 1. listán látható üzenetet kapunk?

Ilyenkor a szerencsétlen áldozat megpróbálkozik a hibaüzenetben javasolt -f kapcsolóval. Ez többnyire segít. Na, nem mindig. Előfordulhat például, hogy egy-egy csomag nem található meg a forrásterületen, pedig ott kellene lennie. Velem esett meg, hogy amikor a ktimer* csomagot kerestem, amelyből a lista szerint a 2.2.2-8-asnak kellett volna a tükörön lennie, már csak a 2.2.2-8.1-esre bukkantam. Ez az a pont, amikor kézzel kell megoldanunk a helyzetet.

Egy szinttel lejjebb: dpkg

Az apt-get a háttérben valójában a dpkg programot hívogatja. A fenti példánál maradva, ha a kezelőrendszer nem talál meg egy csomagot, magunknak kell a letöltési területre eltéblábolnunk, majd letöltés után a dpkg segítségével telepíteni:

```
# lynx --source ftp://...akarloh-is-van/
# ktimer_2.2.2-8.1_i386.deb > /opt/source/
# ktimer_2.2.2-8.1_i386.deb
# dpkg -i /opt/source/ktimer_2.2.2-8.1_i386.deb
```

Természetesen egy idő után a tükör is helyesen tartalmazza a csomagot, semmi baj, az upgrade képes ezt a helyzetet kezelni. Egyébként érdemes egyszer végignézni, mi mindenre képes ez a csuda ügyes program, majd komolyan megfontolni

2. lista Az atp használata

```

ferike:~# apt-get install ace-of-pen*
Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
  ace-of-penguins
The following NEW packages will be installed:
  ace-of-penguins
0 packages upgraded, 1 newly installed,
  0 to remove and 0 not upgraded.
Need to get 333kB of archives. After
  unpacking 595kB will be used.
Do you want to continue? [Y/n]
Get:1 http://dexter.sid/main ace-of-penguins
  1.1-2.1 [333kB]
Fetched 333kB in 0s (435kB/s)
Selecting previously deselected package
  ace-of-penguins.
(Reading database ... 38202 files and
  directories currently installed.)
Unpacking ace-of-penguins (from
  .../ace-of-penguins_1.1-2.1_i386.deb) ...
Setting up ace-of-penguins (1.1-2.1) ...

```

3. lista Részlet a /etc/apt-proxy/apt-proxy.conf fájlból

```

add_backend /debian/ \
  /mnt/store/apt-proxy/debian/ \
  ftp.linuxvilag.hu:mirrors/debian/ \
  ftp.fsn.hu:linux/debian/ \
  ftp.de.debian.org::debian/ \
  ftp2.de.debian.org::debian/

```

a dpkg -h által kiírt utolsó sort: „Felhasználóbarát csomagkezeléshez a dselect programot használjuk”.

Csili-vili csomagkezelés: dselect és aptitude

Bár először mindenkinek ezt a programot ajánlják, nem használom túl sokat. Sőt, hogy teljesen őszinte legyek, egyetlen izgalmas szolgáltatását alkalmazom: ha olyan csomagot keresek, amelynek nem tudom pontosan a nevét, elindítom a programot, belépek a *Select* területre (S, ENTER, majd SZÓKÖZ), és itt szövegesen keresek rá (/ (perjel), majd a keresett szöveg-rész). Ha például arra vagyok kíváncsi, hogy melyik csomagban is vannak a más rendszerekből már ismert pasziánszjátékok, rákeresek a *Solitaire* szóra, mire a rendszer az *ace-of-penguins* sorra ugrik. Ha telepíteni szeretném, ütök egy + (pluszjelet), majd az ENTER segítségével elhagyom a terepet. Ekkor szinte biztos, hogy a *dselect* hosszas magyarázkodásba kezd, hogy mit kell még telepítenie. Miután itt megadtuk a függőségeket (kibogoztuk a szálakat), a rendszer a kért csomagokat telepíti.

A másik megoldás, hogy a csomag megtalálása után kilépünk a *dselect*-ből (CTRL-C), majd *apt-get install csomagnév*. Megjegyzem, a csomagnév megadásakor helyettesítő karaktert is használhatunk.

Ha valaki kedveli a menüvezérelt csomagkezelőket, érdemes egy pillanatra elkalandoznia az *aptitude* világába. Ez a

4. lista Példa a /etc/rsyncd.conf fájlra

```

log file = /var/log/rsync.log
pid file = /var/run/rsync.pid

[debian]
comment = A Debian tukor
path = /var/ftp/debian
max connections = 20
read only = yes

```

kezelő is csupán egy „felület”, a háttérben szorgalmasan hívogatja a többieket. Az *aptitude* megjelenésekor elég szegényesnek látszik, ugyanis csupán faszerkezetet mutat. Aki viszont hozzászokik a használatához, döbbenetes sebességgel lesz képes a szükséges feladatokat megoldani. Nagy segítség, hogy a fában külön találjuk a telepített és a nem telepített csomagokat, a változatszámukkal együtt.

Használata hasonlít a többi hasonszorú programéra. A / gombbal tudunk keresni, a + jelöli ki a csomagokat telepítésre. A válogatás után a program a g hatására új listára vált át, melyben a telepítendő csomagokat láthatjuk, a g újbóli megnyomására pedig elindítja a letöltést. A programot most részletesen nem ismertetem, szerencsére eléri az egyik ismerősöm megadta „felhasználóbarát alapkövetelményt”: kiírja, hogy a ? segítségével kapjuk meg a használathoz szükséges alapvető parancsok leírását.

Ha már itt tartunk, gyorsan szembe kell néznünk a csomagkezelők nagy rákfenéjével: „Jó, jó, de mi történik akkor, ha csak egy rusnya modemes kapcsolatunk van? Hogyan frissítük vagy gyűjtjük be a szükséges csomagokat, és miként végezzük a karbantartásukat?”

Nézzük gyorsan végig, hol és miként érdemes beszerezni és tárolni a csomagokat. A legeslegjobb, ha folyamatos és terhelhető internetkapcsolattal rendelkezünk. Ekkor elég, ha a */etc/apt/sources.list*-ben megadjuk egy megbízható tükör címét, és a gép minden munkát a címről végez. De mi történik, ha nincs folyamatos kapcsolatunk? Ekkor több lehetőségünk kínálkozik: gyűjtögetjük a lemezeket, egyesével beolvastatjuk őket (*apt-cdrom add*); egyesével letöltögetjük a telepítendő csomagokat; elkészítjük a letöltendő csomagok listáját, majd a szomszédban található jó kapcsolaton keresztül töltjük le; helyi tükört hozunk létre; vagy letöltőgyűfelet üzemeltünk be. Az első két lehetőség magáért beszél, nézzük a többi.

Letöltési listák: apt-ftp

Azok számára, akik nem a célgépen kívánják letölteni a csomagokat, ez a két programocska nagyon hasznos lehet. Az első, az *apt-ftp-list* elkészít egy héjprogramot, amelyet egy távoli gépre magunkkal vihetünk (diáktothonos ismerősök előnyben), a távoli gépen a héjprogram a kívánt csomagokat elölti egy kellően nagy hordozható tárolóra (ziplemez, hordozható merevlemez stb.), és a lemezt a célgépre visszahozva az *apt-ftp-inst* egyszerűen futtathatjuk programot, mely elvégzi a telepítéseket.

Nézzük meg, hogy hogyan működik mindez a gyakorlatban! Ha valaki például hordozható merevlemez használ, és maga szeretné a befűzéseket kezelni, használhatja a *--medium k nyvt.ÉrnØv --skip-mount* kapcsolókat, így a két elkészítendő fájl a megadott könyvtárba kerül, a program pedig nem kíséreltezik a kérdéses könyvtár befűzésével (alapértelmezésben

5. lista Forrás megadása

```

$ rsync ftp.fsn.hu::
Welcome to the ftp.fsn.hu anonymous rsync
↳service.

Please email suggestions and questions to
↳bra@fsn.hu

You can access the archive using 'rsync'
↳program like
rsync -av
ftp.fsn.hu:: [Module] /path/to/the/files

Module          Content
-----
ftp              FTP area
linux           Linux distributions
cdimages        CD-ROM Images
$

```

a/ZIP-et befűzi, oda dolgozik, majd a művelet végén ki is fűzi). Az apt-zip-list két fájlt hoz létre. Az első a fetch-script-wget-g@pn0v, mely a távoli gépen futtatandó héj-program. Ha belenézünk, a fájl végén láthatjuk a ténylegesen letöltendő csomagok listáját. Ezt a listát a program magától készíti el, az apt-vel a telepítésre kijelölt, de még le nem töltött fájlok listájából. Fontos még megjegyezni, hogy a program jelenleg a wget-et használja – figyeljünk rá, hogy a letöltést végző környezetben rendelkezésre álljon.

A munka vége még egyszerűbb. Miután a lemezt visszahoztuk, igény szerint befűztük, kiadjuk az apt-zip-inst parancsot (az eszköz kezelésére vonatkozóan lásd a fenti megjegyzéseket), ami az újonnan hozott csomagokkal a kért műveleteket elvégzi.

Tükrök használata

A tükrök felállítása külön feladat, melyre most nem térek ki, egy-egy ilyen területet számos módon ki lehet alakítani, teljes tükröt viszont csak akkor érdemes felállítani, ha a helyileg elérhető tükrök megéri a hálózati terhelést. Szintén érdemes áthozni egy „alapot”, amennyiben nem rendelkezünk nagyon erős kapcsolattal, egy tükrök ugyanis akár húszigás területet is elfoglalhat (amiben a teljes fa minden ága nincs is benne). Mindenképpen megéri viszont tükröt üzemeltetni, ha a hálózatban legalább öt-hat debianos gép van, ugyanis mind a frsítés, mind a teljes telepítés nagyságrendekkel felgyorsulhat. Azonkívül ötvözni is lehet a módszereket, például a gépeken forrásként a helyi tükröt és a letöltőügyfelet is megadhatjuk. Amit szem előtt kell tartani, az, hogy a meglévő sávszélességet ne terheliük feleslegesen.

Letöltőügyfél kis hálózatoknak: apt-proxy

A végére hagytam a nekem legjobban tetsző megoldást. Ez a kis programocská kifejezetten olyan felhasználócsoport számára készült, akik ugyanazokat a fájlokat valószínűleg többször is felhasználják, de nem szeretnék az egész Debian-fát folyamatosan tükrözni. A program helyi gyorstárat üzemeltet és a karbantartását is elvégzi. Ha valaki gyorstárazott fájlt kér tőle, HTTP-protokollon keresztül szolgálja ki. Ha a keresett fájl nincs meg a gyorstárban, a program a háttérben az rsync

segítségével letölti, majd továbbítja. Ez a felhasználó számára elméletileg „áttetsző”, azaz nem okoz fennakadást. Természetesen a gyorstárterület egyúttal tükrök is lehet, kis sávszélesség esetén például hasznos megoldás a nagyobb frissítéseket lemezekről bemásolni (hétvégenként tükrözni stb.), amire pedig napközben van szükség, azt az apt-proxy lehúzza. Nagy előnye még, hogy egyszerre több forrásból is képes dolgozni (csak rsync-protokollon keresztül), ezért több forrást is meg tudunk adni neki, amelyeket a keresett fájlért sorrendben végigjár.

Az apt-proxy beállítása

A program telepítése után nagyon jó sűgóoldal (man apt-proxy) segít a telepítésben. A fejlesztő ígérete szerint a telepítési lépéseket szép lassan beépíti a csomagba, így nem kell sok további lépést megtennünk. Addig is röviden: a /etc/inetd.conf-ban engedélyezni kell a 9999-et, létre kell hozni egy felhasználót, ami alatt a program dolgozik (alapértelmezés: apt-user), majd a /etc/apt-proxy/apt-proxy.conf fájlban be kell állítani a gyorstár helyét és a tükröket.

Itt az add_backend sorok az érdekesek. Három részből állnak: az első a kérelem előtagja szerinti illesztés (ahogy a felhasználó által kért fájl kezdődik), a második a gyorstár a helyi lemezen, a többi pedig a használható források címe (rsync-formátumban).

Röviden az rsyncról

Az rsync hálózati fájlátviteli program, mely egy gyors ellenőrzés és hasonlítás segítségével csak a megváltozott részeket továbbítja. Más szóval: ha például egy negyven megás fájl végéhez hozzáfűzünk egy új sort, az rsync nem tölti le újból az egész fájlt, csupán azt a pár kilobájtot. Na jó, ennél egy kicsit bonyolultabban működik, de a lényege ennyi.

Egy terület megosztása rsync protokollon keresztül nem túl nehéz, csupán az rsync telepítése után olvassuk el a sűgóoldalt, készítsük el a /etc/rsyncd.conf fájlt, engedélyezzük az rsync-et a /etc/inetd.conf-ban, majd indítsuk újra az inetd-mont. A beállítófájl igen hasonló a samba.conf szerkezetéhez. Szerencsére például beállíthatjuk, hogy egyszerre legfeljebb hányan csatlakozhatnak a szolgáltatáshoz.

A felhasználó oldaláról az rsync nagyon hasonlóan működik, mint például az rsh vagy az ssh. Hogy az újabb protokoll ne okozzon további kavargást, az rsync-területek címzésénél két kettőspontot használjunk (vagy az rsync:// formátumot). Ha kettőspontot használunk, a program egyszerű távoli héjmásolást végez. Ha pedig az elérhető megosztások és fájlok listáját akarjuk megkapni, akkor csak a forrást kell megadni (lásd az 5. listát). Ha valakit a program felhasználhatósága részletesebben érdekel, melegen ajánlom az idevágó sűgóoldalakat (man rsync és man rsyncd.conf). A leírásokban teljes példákat is találhatunk a biztonsági másolatok és tükrök készítéséhez, illetve fenntartásához.

Röviden igyekeztem összefoglalni, miként érdemes elindulni a Debian-csomagok kezelésének útvesztőjében, és mint ahogy az elején írtam, a paletta sokkal szélesebb a most bemutatott programoknál. Érdemes minél többet megismerni, majd a nekünk legjobban tetszőket kiválasztani.



Szy György

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