

A PPPD beállítása Linuxban (1. rész)

Az Internethez való kapcsolódás könnyebb lehet, mint gondolnánk. Tony kétrészes cikkében a modemek Linux alatti beállítását ismerteti.



Manapság asztali számítógépükre egyre többen telepítenek Linuxot. Sajnos sokan nehézségekbe ütköznek, amint olyan lehetőséget próbálnak ki, amelyről ma már senki sem hajlandó lemondani: kapcsolódni szeretnének az Internetre. Milyen akadályokkal találják is szemben magukat ezek a felhasználók? Több is jelentkezik, de a legfontosabb akadály az MS Windows távoli elérése hivatalos lehetőségének vagy a Mac által biztosított ConfigPPP-nek a hiánya. A végeredmény az, hogy sok felhasználó a gépére hasonló feladatokat ellátó rendszereket telepít, úgymint Gnome betárcsázót, illetve Linuxconf, KDE tárcsázó (dialer) stb. programot, anélkül téve azonban, hogy ezekről a programokról különösebb ismeretekről rendelkezzenek.

Túlságosan is nagy lehetősége van egy olyan helyzet kialakulásának, amikor semmilyen önműködő beállítóprogram sincs telepítve a gépre, például a felhasználó sem a Gnome-ot, sem a KDE-t nem jelölte ki telepítésre. Ezek után miért ne vállalnánk a kihívást, hogy megtanuljuk az internetkapcsolat kézi beállítását? Ez a cikk éppen a modemek Linux-rendszer alá történő telepítéséről szól, amelynek megoldása, úgy tűnik, számos felhasználónak fejtörést okoz. Elmondom, hogyan is kell beállítani a PPPD-t, amelynek előfeltétele természetesen a modem jó beállítása.

Előzetes ismeretek

Először a héjprogram használatával célszerű megismerkednünk: hogyan kell könyvtárat váltani, egy könyvtár tartalmát listázni, bármilyen szerkesztőprogram segítségével állományokat szerkeszteni, valamint az X-rendszerben virtuális konzolokat és terminálokat használni.

Cikkünkben feltételezzük, hogy olvasónk nem szoftveres modemmel rendelkezik, amely csak a Windowszal képes együttműködni. Bár az ilyen modem használata is lehetséges, beállítása sok fáradsággal jár, és túlmutat jelen írás keretein.

A modem megkeresése

Elsőként meg kell állapítanunk, tulajdonképpen hol is van a modem. Azt a soros kaput kell keresnünk, amelyhez a modem kapcsolódik. Ez még abban az esetben is szükséges, ha beépített modemet használunk, ugyanis a modemkártyán is található soros kapu. A számítógépbe feltehetőleg két soros kapu lett beépítve. Nagyon is valószínű, hogy az egyiket már az égér használja, hacsak nem PS/2 vagy USB-csatolófelületű egerünk van. A Unix-rendszerben minden fizikai berendezés a `/dev` könyvtár egy-egy bejegyzéseként jelenik meg. Ez a könyvtár minden a számítógépre telepített készülékre nézve tartalmaz egy bejegyzést. A soros kapukat a `ttyS` betűsorozattal szokták jelölni, amelyeket 0-tól 3-ig terjedő, egyjegyű szám követ.

```
cd dev
ls -l ttyS*
```

```
crw----- 1 merc  tty  4,
└─64 Aug  3 10:24 ttyS0
crwxr-xr-x 1 root  tty  4,
└─65 Aug  3 10:25 ttyS1
crw----- 1 root  tty  4,
└─66 May  6 1998 ttyS2
crw----- 1 root  tty  4,
└─67 May  6 1998 ttyS3
```

Vajon melyikhez csatlakozik a modem? A válasz: attól függ, hogy melyik aljzatba dugtuk be a modem kábelét, illetve a belső modemnél az dönti el, hogy hogyan állítottuk be. Ha már megszoktuk, hogy a soros kapukra COM1, COM2 stb. névvel hivatkoznak, azt is tudnunk kell, hogy ezek Linuxban használatos megfelelői a következők: COM1=ttyS0, COM2=ttyS1, COM3=ttyS2, COM4=ttyS3. Amennyiben nem tudjuk megállapítani, hogy a modem melyik csatlakozóba lett bedugva, rövid úton kideríthetjük.

Rendszerint létezik egy modemnévvel ellátott közvetett hivatkozás – ez a Windows szóhasználatban nagyjából a parancsikon megfelelője –, amely az egyik soros kapura mutat. Az én rendszeremben például:

```
ls -l modem
lrwxrwxrwx 1 root root 5 Feb 7 2000
└─modem ->ttyS1
```

Az én esetemben a modem a második soros kapura – a `ttyS1`-re – van kötve, azaz a fenti példában bemutatott közvetett hivatkozás azt jelenti, hogy bármilyen program, amely a `/dev/modem` állományt használja, valójában a `/dev/ttyS1` kapun dolgozik.

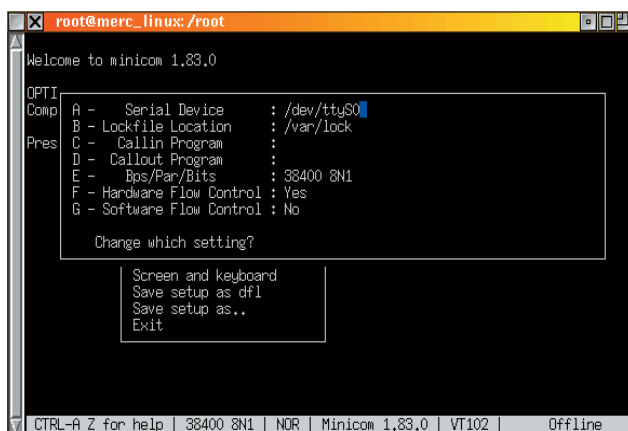
Sose felejtjük el, hogy rendszerünk eltérhet a példában bemutatottól, sőt még az is előfordulhat, hogy a `ttyS0` vagy `ttyS1` kapukra mutató közvetett hivatkozások sem léteznek. Most azt a célt tűztük ki, hogy egy olyan bejegyzést hozunk létre a `/dev` könyvtárban, amely a megfelelő soros kapura mutat, vagyis arra, amelyiken a modem kapcsolódik.

Nos, először is meg kell vizsgálnunk, melyik kapura van csatlakoztatva a modem. Gépeljük be a `minicom` parancsot. Ahhoz, hogy ellenőrizzük, a `minicom` képes adatcserét lebonyolítani a modemmel, gépeljük be az `at` parancsot, és üssük le az `ENTER`-t. Ha OK válasz érkezik, akkor a `minicom` a `/dev` könyvtárbeli megfelelő állományt használja a modem elérésére. Máskülönben a `minicom` valamilyen okból kifolyólag nem tud „párbeszédet” folytatni a modemmel. Amennyiben a modemtől nem érkezik megerősítő válasz, elsőként ellenőrizzük, hogy hová lett kötve, vagyis melyik soros kapuhoz csatlakozik. Ez a feladat a `minicom` programból könnyen elvégezhető: nyomjuk meg a `CTRL-A` billentyűkombinációt, majd az `O`-t. Fontos megemlíteni, hogy a `minicom` néha úgy

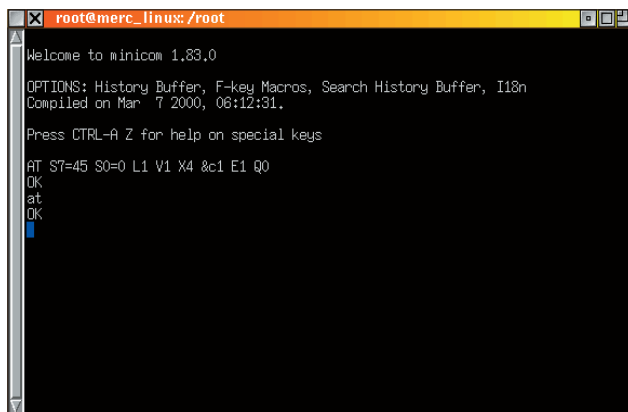
van beállítva, hogy CTRL helyett az ALT gombot használja. Amennyiben ez a helyzet, ne felejtjük el, hogy a CTRL billentyű helyett az ALT billentyűt kell lenyomni, vagyis a fenti példában a ALT-A után az O billentyűt.

A képernyőn a `minicom` beállítási képernyője jelenik meg. Most válasszuk ki a *Serial port setup* (a soros kapu beállítása) lehetőséget, majd üssünk ENTER-t.

Ezt követően bökjünk az A-ra, és válasszunk másik soros kaput. Az éppen beállított soros kaput (`/dev/ttyS0`) változtassuk meg például `/dev/ttyS1`-re (1. kép). Lépjünk ki a `minicom` programból és indítsuk újra. Nyomjuk meg az ENTER gombot, majd mentjük a beállítást a *Save default as dfl* (Beállítás mentése alapértelmezettként), végül válasszuk a *Exit* (Kilépés) lehetőséget. Ekkor a `minicom` programból a CTRL-A, majd az Q megnyomása után kiszálltunk.



1. kép Erről a képernyőről megtudhatjuk, a `minicom`nak melyik soros kaput kell használnia, hogy sikeres adatcserét folytathasson a modemmel



2. kép Így kell működnie a `minicom`nak, ha a modem OK választ küld

Futtassuk ismét a `minicom` programot, hogy ellenőrizni tudjuk, vajon végez-e adatcserét az általunk választott soros kapuval. Üssük le az ENTER billentyűt, és ha ezúttal megkaptuk az OK-t, máris megtaláltuk a modemet (2. kép). Egyébként az ebben a fejezetben leírt lépéseket mindaddig ismételnünk kell, amíg a keresett soros kapura rá nem bukkanunk, tehát a modemtől OK választ nem kapunk (az `at` parancs begépelésekor a `minicom` parancs futtatását követően).

Mindig arra gondoljunk, hogy a modem csatlakozási pontjának keresése közben a közönséges hibakeresési módszereket kell követnünk. Különösen fontos meggyőződnünk róla, hogy a modem be van-e kapcsolva, csatlakoztatva van-e a géphez,

megfelelő-e a tápellátása, és megbízható-e a számítógéppel való kapcsolata. Ezek a vizsgálatok talán jelentéktelennek tűnnek, de éppen magától értetődő jellegük miatt gyakran figyelmen kívül szokás hagyni őket. Ennek nyomán órákat tölthetünk el egy modem beállításával, amely nincs a géphez csatlakoztatva, sőt talán be sincs kapcsolva.

Ha a modemtől megérkezik az OK válasz, akkor minden működésre kész, és már csak néhány teendőt kell elvégeznünk. A `minicom` programon belül a CTRL-A, majd O billentyű használatával a beállító képernyő jelenik meg; itt válasszuk a *Serial port setup* (Soros kapu beállítása) lehetőséget, majd üssük le az ENTER billentyűt.

Mindjárt az első sorban szembetűnik, hogy a `minicom` melyik állományt használja a modemmel folytatott adatcsere lebonyolítására. Ha ez a `/dev/modem`, akkor minden helyesen van beállítva a rendszerben, és máris cikkünk *Párbeszéd a modemmel* részére ugorhatunk.

Amennyiben a modemtől OK választ kapunk, de a `minicom` egy másik soros kapura van beállítva, a `/dev` könyvtárban létre kell hoznunk a modem névvel ellátott közvetett hivatkozást.

Ez a `/dev/modem` bejegyzés fogja biztosítani a modem Linux-rendszerbeli működését – tekintsünk vissza: pontosan ezt tűztük rövid távú célul. Vessük papírra, hogy a `minicom` melyik kaput használja, most tételezzük fel, hogy ez a `/dev/ttyS0`. Lépjünk ki a beállítási képernyőből, vagyis üssük le az ENTER billentyűt, és válasszuk az *Exit*-et (Kilépést), és szálljunk ki a `minicom` programból is a CTRL-A, majd a Q billentyűk leütésével. Ezzel vissza is tértünk a héjprogramhoz. Most lépjünk be a `/dev` könyvtárba, és hozzunk létre közvetett hivatkozást a modem számára az alábbi módon:

```
cd /dev
ln -sf ttyS0 modem
ls -l modem
lrwxrwxrwx 1 root root 5 Aug 3
12:32 modem ->ttyS0
```

Természetesen a `ttyS0` kaput a `minicom` beállítási képernyőjén talált kapubeállítással kell helyettesítenünk. Most ismét futtassuk a `minicom` programot. Az AT parancs begépelésével és az ENTER leütésével győződjünk meg róla, hogy minden továbbra is működik. A megerősítő OK válasznak ismét meg kell érkeznie (2. kép).

Újra lépjünk a `minicom` beállító képernyőjébe – CTRL-A és O –, majd válasszuk a *Serial port setup* (Soros kapu beállítása) lehetőséget. Most az A billentyű lenyomásával a `/dev/modem` bejegyzéssel végezzük el a `/dev/ttyS0` állomány helyettesítését (1. kép). Bökjünk rá az ENTER-re, válasszuk a *Save default as dfl*-t (Beállítás mentése alapértelmezésként), és lépjünk ki a beállítóképernyőből (Exit). Hogy a `minicom` az `œj beáll tÆsokkal` megfelelően működjön, ki kell lépni a `minicom` programból, majd újra kell indítanunk: CTRL-A és O.

A `minicom` most már a `/dev/modem` készüléket használja. Gépeljük be az AT parancsot, erre meg kell kapnunk az OK választ (2. kép). Ha ez a helyzet, gratulálunk a sikeres beállításhoz.

A modem beállítása a legkritikusabb lépés az Internethez történő kapcsolódás során. Sok felhasználó nem tudja, hogy modemje melyik soros kapuhoz van kapcsolva. Amint az előző fejezetben láttuk, a modem már korábban jól be lett állítva, most csak meg kellett állapítanunk, hol van, és a `/dev` könyvtárban létre kellett hoznunk a helyes bejegyzést, amely a megfelelő készülékállományra mutat.

© Kiskapu Kft. Minden jog fenntartva

Párbeszéd a modemmel

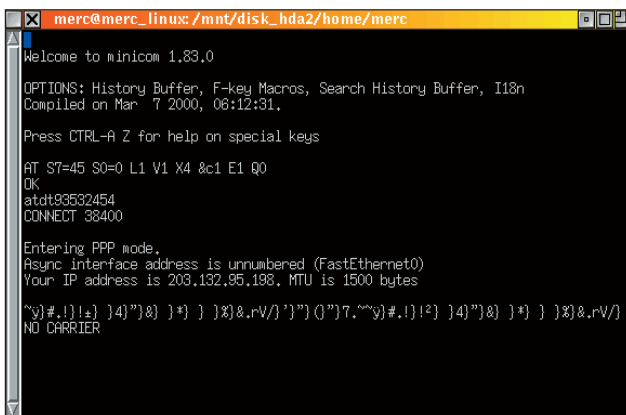
Érdeemes megjegyezni, hogy maga a minicom semmit sem tud az AT parancsról vagy az OK válaszról. Fő feladatai valójában rendkívül egyszerűek: megjeleníti a soros kapu felől érkező karaktereket, illetve a felhasználó által a billentyűzeten begépet karaktereket elküldi a soros kapu felé.

Bizonyos értelemben a modem olyan, akár egy robot, amely a géppel a soros kapun keresztül társalog. Ez az oka annak, hogy a minicom bizonyos értelemben megengedi, hogy „magánbeszélgetést” folytassunk a modemmel.

A fenti példában a felhasználó küldte az AT jelet, majd a modem válaszolt az OK< Enter > jelsorozattal.

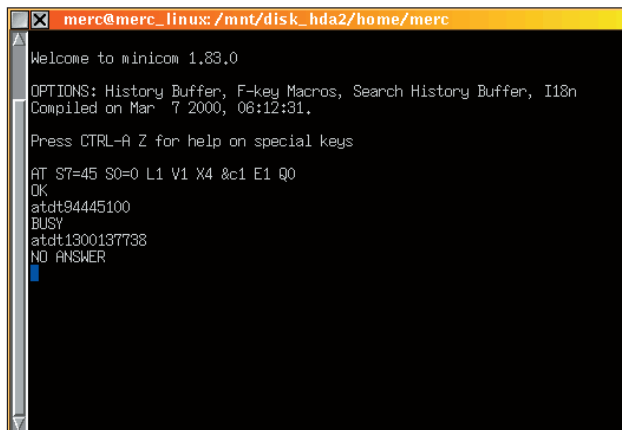
Miféle párbeszédet folytathatunk a modemmel? Tény, hogy minden modem képes az at-parancskészleten alapuló Hayes-parancskészlet fogadására. De némely alkatrészgyártó a szabványon túl érdekes bővítésekkel is rendelkezik. Ha kíváncsiak vagyunk rá, melyek ezek, nem kell mást tennünk, csak tanulmányozzuk át őket a felhasználói útmutatóban. Például az ALT-1, ALT-2, ALT-3 minden modem esetén ugyanazt jelenti: a hangszóró hangerőszabályozóját alacsony, közepes vagy nagy teljesítmény fokozatra állítja. Annak ellenére, hogy e parancsok közül nem soknak az ismeretére van szükségünk, talán mégis szeretnénk kipróbálni néhány parancsot, hogy egy kicsivel többet tudjunk modemünk működéséről. Mindig gondoljunk arra, hogy az AT&F és aT&W parancsokkal bármikor visszaállíthatjuk a gyári értékeket, amennyiben bizonytalan működésű parancsokat használtunk, amelyek elrontották a korábban jól működő beállítást, sőt talán még a kapcsolatlétesítést is lehetetlenné tették.

Az ATDT93355100 parancsot tárcsázásra használhatjuk, jelen esetben a 93355100 telefonszám hívására. A DT jel (tone) üzemmódot jelent, míg a DP impulzus (pulse) üzemmódot jelent. Az utóbbit akkor kell használni, ha vidéki környezetben analóg telefonközponthoz kapcsolódunk. Kíséreljünk meg ATDT pa-



3. kép Ilyen volt az én képernyőm, miután internetszolgáltatómmal sikerült kapcsolatot teremtenem

rancsot küldeni a modemnek, ezután adjuk meg az internetszolgáltató telefonszámát, majd kísérjük figyelemmel, mi történik. Az én esetemet a 3. kép szemlélteti. Mint az világosan látszik, az alkatrészek összehangolása után – ez mindkét oldalon sok összespólolással jár – a modem szépen küldi az CONNECT 52500 üzenetet, amely arra utal, hogy a kapcsolat hiba nélkül létrejött. Természetesen emellett léteznek más üzenetek is: NO DIALTONE: a modemhez nincs telefonvonal kapcsolva; BUSY: a telefonvonal foglalt; NO ANSWER: nincs válasz stb. (4. kép). Az én esetemben a kapcsolat sikeresen létrejött, és amint kap-



4. kép Két szám tárcsázásának eredménye: az első szám foglalt volt, a második pedig nem felelt

csolódtam, internetszolgáltatóm rejtélyes jelsorozatot küldött, amely valahogy így kezdődött: ~}#. !}!-} }4} }&} }*} }%}. Arról van szó, hogy internetszolgáltatóm a PPP-démómmal szeretne kapcsolatot teremteni, ugyanakkor én a minicomot futtatom, ami azt mutatja, amit internetszolgáltatóm küldött a PPP-démómmal.

Innentől kezdve mindig tartsuk észben, hogy a modem ezentúl nem hajlandó az a parancsaira válaszolni, minden soros kapura küldött adat modulálva lesz, és a vonal másik végére lesz küldve. Ugyanakkor a vonal másik végéről érkező adatok demodulálva lesznek, és a soros kapura lesznek átirányítva – innen származik a modem elnevezés. A számítógép, jobban mondva a soros kapu minderről semmit sem tud. Minden pontosan úgy működik, mintha a gépet a másik oldallal soros kábel kötné össze. Tény, hogy ha két géppel rendelkezünk, ezeket a soros kapukon egy soros kábel segítségével keresztül könnyedén összeköthetjük a PPPD-vel.

Csak arra kell figyelni, hogy fordítás kábel legyen, amit használunk, vagyis az első gép küldő tűcsatlakozója a másik gépnél fogadó csatlakozó legyen és fordítva.

A következő hónapban a számítógép egy olyan beállításáról fogok beszélni, amelynek segítségével elérhető, hogy az Internethez mindenféle bonyolult eszköz nélkül csatlakozni tudjunk. A lényeg az, hogy mardjunk vonalban!

Linux Journal 2002. február, 94. szám



Tony Mobily

(merc@mobily.com) a Login olasz számítógépes magazin műszaki szerkesztője, valamint linuxos vizsgára készít fel, továbbá rendszergazda és táncoktató.

A nyelvek közül többek között az angolt, az olaszt, a C-t és a Perl-t ismeri. Tony honlapja a

☞ <http://www.linuxcertification.com> címen kereshető fel.

Kapcsolódó címek

- ☞ <http://www.tldp.org/HOWTO/PPP-HOWTO/>
- ☞ <http://www.topology.org/linux/pppd.html>
- ☞ <http://oh3tr.ele.tut.fi/~oh3fg/ppp/ppps.html>