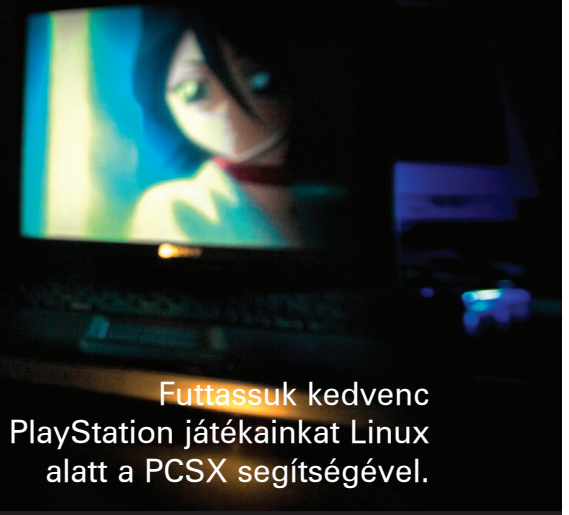


PlayStation játékok Linux alatt



Futtassuk kedvenc PlayStation játéinkat Linux alatt a PCSX segítségével.

■ Ez a cikk a *Sony PlayStation* játékokkal és a *PCSX PlayStation*-emulátorral foglalkozik. Azért választottam éppen ezt a rendszert, mert a *PlayStation* játékok a neten és a játékküzetekben egyaránt megtalálhatók, utóbbi esetben elsősorban használtcikk-részlegen keresséjük...

A PCSX beszerzése és telepítése

A *PCSX* beszerzéséhez látogassunk el az alkalmazás hivatalos honlapjára (lásd on-line irodalomjegyzéket). Számunkra most természetesen csak a linuxos rész lesz az érdekes. Töltsük le a legfrissebb verziót. Ha ez megvan, lépünk be abba a könyvtárba, ahova a fájlt helyeztük, és bontjuk ki a fájlt. Ha az anyagot az archívummal azonos könyvtárban szeretnénk elhelyezni, akkor ehhez elegendő a következő parancsot kiadni:

```
tar xzvf Lpcsx-1.5.tgz
```

Ez a művelet létrehoz egy *Pcsx* nevű könyvtárat az adott helyen (például `~/Downloads/Pcsx`). Ezzel kezünkben az alapeszköz, de a beszerző körútnak még nincs vége. Töltsük le a hozzá tartozó bővítményeket (*plugin*) is.

A bővítmények és a BIOS beszerzése

A *PCSX* tulajdonképpen csak egy programfuttatási környezet. Ahhoz,

hogy valóban tudjunk vele játékokat játszani, mindenféle bővítményekre lesz szükségünk, ezek biztosítják ugyanis azokat a funkciókat, amelyekre a játékoknak szükségük van. A *Next-Gen Emulation* webhelyén (lásd az on-line forrásokat) szerencsére egy igen kiváló gyűjteményt találhatunk. Kattintsunk itt a *PlayStation* szakaszra, majd a bal oldalon a *Plugins* linkre. Itt a *Linux Plugins* pont az, ami számunkra igazán fontos.

Jómagam *Pete XGL2 Linux* videóbővítményét, *P.E.O.P. Linux OSS SPU* hangkezelőjét, a *CDR Mooby Linux* bővítményt (ezzel lehet a valódi *CD*-k helyett *ISO* lemezeképből futtatni a programokat), valamint a *padJoy* rendszert választottam.

Ha ezen kívül bármelyik bővítményről többet szeretnénk megtudni, csak kattintsunk a neve melletti ház ikonra. Ha pedig valamit le akarunk tölteni, akkor természetesen a lemez ikon a megfelelő választás. A letöltésnél vagy adjuk meg célként a *PCSX* rendszer *Plugin* könyvtárát (esetünkben `~/Downloads/Pcsx/Plugin`) vagy, másoljuk át ide a fájlokat.

A bővítményeken kívül szükségünk lesz egy *PlayStation BIOS*-ra is. A „szükségünk lesz” talán túl erős kifejezés ezen a helyen, hiszen magához a *PCSX*-hez is tartozik valamiféle kez-

detleges *BIOS*, de ha azt szeretnénk, hogy rendszerünk a lehető legtöbb játékkal kompatibilis legyen, akkor a legtöbben azt javasolják, hogy töltsünk le egy valódi *PlayStation BIOS*-változatot. Miután egy *BIOS* tartalmának közzététele jogi szempontból kissé kérdéses ezért itt nem adok meg címet, de ha valaki elolvassa *TheGing* útmutatóját a *PlayStation BIOS* leképezéséről (lásd az irodalomjegyzéket), akkor nemcsak többet tudhat meg a *PlayStation BIOS*-okról, de talál egy listát is a különféle kipróbálható változatokról, természetesen letöltési címmel együtt. Az is jó megoldás, ha beírjuk a kipróbálandó változat nevét egy keresőbe és ígérem találni hamar meg fogjuk a fájlokat. Mentsük a kiválasztott fájlt a *Bios* alkönyvtárba (esetünkben `~/Downloads/Pcsx/Bios`), vagy mozgassuk oda, ha már szert tettünk rá.

Az egyszerűbb részek telepítése

Egyes összetevőket egyszerű telepíteni, másokat viszont bonyolultabb. Kezdjük a könnyebbekkel, legelőször a *BIOS*-szal. A letöltött fájlnak valószínűleg *.zip* kiterjesztése lesz, úgyhogy vagy csomagoljuk ki a grafikus fájlkezelővel, vagy a kézi kibontáshoz adjuk ki a következő parancsot:

```
unzip fájlnév
```

(A fájlnev lehet például *scph1001.zip*). És ennyi volt, a BIOS telepítve van. Ezután *Pete XGL2 Linux GPU* bővítményét telepítjük. A letöltött csomag neve ebben az esetben ilyesformán néz majd ki: *gpupetexgl208.tar.gz*. Csomagoljuk ki a fájlt az tetszőleges módon. Miután alapértelmezésben nincs konfigurációs könyvtár, hozunk létre egyet *Pcsx/cfg* néven. (például *~/Downloads/Pcsx/cfg*). Ezután másoljuk a *gpuPeteXGL2.cfg* és *cfgPeteXGL2* fájlokat a *cfg* könyvtárba. A következő lépés a *P.E.O.P.s Linux OSS SPU* plugin beszerzése. Itt az előzőekhez teljesen hasonlóan járhatunk el. A fájl neve *spupeopsoss108.tar.gz* vagy valami ilyesmi lesz. Bontsuk ki ezt a bővítményt is, és másoljuk be a *spuPeopsOSS.cfg* és a *cfgPeopsOSS* nevű fájlokat a beállításokat tároló könyvtárba. Aki többet tud a Linux hangkezeléséről, az a név alapján már valószínűleg sejtí, hogy ez a bővítmény az *Open Sound Systemet* használja. Ha rendszerünk nem használja az *OSS*-t, akkor először telepíteni kell azt és be is kell állítani ahhoz, hogy legyen hangunk. Nagy a valószínűsége persze, hogy – ha valamilyen népszerűbb terjesztést használunk – akkor a rendszer alapértelmezésként feltelepített ezt a hangrendszert. Az ezzel kapcsolatos részleteket értelemszerűen a dokumentációból tudhatjuk meg, vagy utánanézhethetünk a dolognak magával a csomagkezelővel is.

A CDR Mooby Linux telepítése

Ez a bővítmény egy kicsit keményebb dió lehet. A telepítése lehet gyerekjáték, de az is megeshet, hogy egyáltalán nem fog működni. Aggodalomra ugyanakkor nincs ok, létezik ugyanis gyors javítás. A *CDR Mooby* a *cdrmooby2.8.tgz* vagy hasonló nevű fájlban érkezik. Bontsuk ki ezt a tar csomagot a plugin könyvtárba. Valószínűleg mindössze ennyire lesz szükség. Ha később a *PCSX* elindításakor mégis hibát tapasztalunk (maga a program mindenképpen elindul) vizsgáljuk meg a kimenetet. Ha a következő üzenetet látjuk:

```
libbz2.so.1.0: cannot open
shared object file: No such
file or directory
```

akkor a *PCSX* rossz helyen keresi ezt a könyvtárat. Írjuk be az alábbi két

parancs valamelyikét (kezdjük a *locate*-tel és ha nem működik, próbáljuk a *find*-ot):

```
locate libbz2.so.1.0
```

vagy:

```
find / -name libbz2.so.1.0* 2>
/dev/null
```

Az eredmény a következőképpen nézhet ki:

```
/usr/lib/libbz2.so.1.0.2
```

Ha megtaláltuk a keresett megosztott könyvtárat, akkor jegyezzük meg a helyét, majd hozzunk létre egy rá mutató szimbolikus láncot. Ehhez – *root*-ként bejelentkezve – a következő parancsot kell kiadnunk:

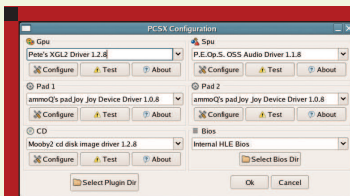
```
ln -s /usr/lib/libbz2.so.1.0.2
/usr/lib/libbz2.so.1.0
```

Természetesen előfordulhat, hogy a kérdéses könyvtár a mi rendszerünkön máshol található. Járjunk el értelemszerűen.

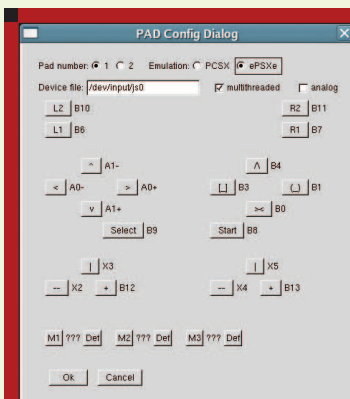
A padJoy telepítése

A „legrosszabbat” a végére hagytam. A *PCSX* használatához nem feltétlenül kell játékvezérlő (megteszi a billentyűzet is), de az igazi *PlayStation* élményhez a vezérlő használata is hozzátartozik. Szóval vágjunk bele. Azért mondtam, hogy ez a legrosszabb, mert a *padJoy* esetében többről van szó, mint egy egyszerű bővítmény telepítéséről. Magát a játékvezérlőt is működésre kell majd bírunk, de egyszerre csak egy dologgal foglalkozunk. Először ellenőrizzük, hogy telepítettük-e a *C* nyelvű programok lefordításához szükséges eszközöket (például a *GCC*-t). Szintén szükségünk lesz a *GNOME* fejlesztőeszközökre. Ezen kívül a csomagkezelő segítségével nézzük meg, hogy fent van-e a *gtk-devel* nevű csomag – ezt hívhatják *gtk+-devel*-nek vagy valami ehhez hasonlóknak is. Ha minden szükséges kellék együtt van, fordítsuk le a *padJoy* bővítményt. A letöltött *padJoy* fájl körülbelüli neve: *padJoy082.tgz*. Csomagoljuk ezt ki a *Plugin* mappába, ahol ezzel létrejön a saját – nem túl meglepő módon *padJoy* nevű – alkönyvtára is.

(Esetünkben tehát a *~/Downloads/Pcsx/Plugin/padJoy* helyre kerül az anyag.) Lépjünk be a *padJoy/src* könyvtárba (példánkban *~/Downloads/Pcsx/Plugin/padJoy/src*) és adjuk ki a *make* parancsot. Ezzel elindítjuk a bővítmény lefordítását. Ha a művelet hibával megáll, valószínűleg hiányzik egy olyan csomag vagy könyvtár, amire támaszkodik a kód. Hogy melyik, az szerencsére általában kiderül a hibaüzenetből. Ha minden rendben ment, akkor az *src* könyvtárban találunk egy *cfgPadJoy* és egy *libpadJoy-0.8* nevű fájlt. Másoljuk át a *cfgPadJoy*-t a *Pcsx/cfg* (példánkban *~/Downloads/Pcsx/cfg*) könyvtárba, a *libpadJoy-0.8*-at pedig a *Pcsx/Plugin* helyre (vagyis a *~/Downloads/Pcsx/Plugin*-ba). Mielőtt továbbmegyünk, vizsgáljuk meg a játékvezérlőt, amit a *padJoy*-jal használni akarunk, már ha rendelkezünk ilyen eszközzel. Digitális vagy analóg? Van-e rajta olyan csatlakozó, amivel a számítógéphez tudjuk kapcsolni (az *USB* például megfelelő)? Kell-e a csatlakoztatásához úgynevezett *Game Port*, és van-e ilyen a gépünkön? (Aki nem biztos benne, hol kell ezt a dolgot keresni, az kezdje a hangkártyával. Nagy valószínűséggel azon lesz egy közepes méretű csatlakozó. Ó az.) Előfordulhat, hogy a játékvezérlőnek egyedi, speciális csatlakozója van. Akinek van egy *Xbox* vezérlője (nem az *Xbox 360*-é, ami *USB*-vel megy, hanem az eredeti *Xbox*-é), az nézze meg *Dan Gray* honlapját (lásd on-line irodalomjegyzéket) ahonnan megtudhatja, hogy egy kis forrasztással hogyan alakítható át a vezérlő csatlakozója *USB*-vé. A leírtakat persze mindenki csak a saját felelősségére alkalmazza. Ha másféle vezérlőnk van egyedi csatlakozóval, rendszerint vehetünk hozzá harmadik féltől származó átalakítót, akár az interneten is. Én két különböző vezérlőt próbáltam ki a *PCSX*-szel. Először keresgéltem egy kicsit és találtam egy joysticket game portos csatlakozóval. Ezután felfedeztem, hogy a *SoundBlaster Live!* kártyámon van game port. Az első, amit észrevettem, hogy nincs alapértelmezett joystick eszköz a rendszeremen. (A */dev/js0*-át vagy a */dev/input/js0*-át kell keresni. Ezek gyakran ugyanazt az eszközfájlt takarják, csak egy szimbolikus láncal össze vannak kapcsolva).



1. ábra A PCSX Configuration fő párbeszédablaka



2. ábra A Pad szakasz beállításainak megadására szolgáló párbeszédablak

Aztán rájöttem, hogy ez azért van, mert az általam használt terjesztés *devfs*-t használ és csak azokat az eszközöket hozza létre, amikre abban a pillanatban szükségem van. Mindössze annyit kellett tennem, hogy root felhasználóként begépeltem az alábbi két parancsot:

```
modprobe analog
modprobe joydev
```

Ezután, mikor beírtam, hogy `ls /dev/j* /dev/input/j*`, már láttam is, hogy a rendszer valóban létrehozta a `/dev/input/js0`-át, jelezve ezzel, hogy megtalálta a botkormányt. Ha látszólag mindent jól állítottunk be és még mindig hiányzik az eszközfájl, akkor gépeljük be a következő parancsot:

```
mknod /dev/input/js0 c 13 0
```

Erre a fájl garantáltan létrejön. A botkormány (vagy gamepad, vagy amit éppen használunk) teszteléséhez telepítenünk kell a botkormányt (joystick) kezelő eszközöket, ha eddig ezt még nem tettük volna meg. Ezután adjuk ki a következő parancsot:

```
jstest /dev/input/js0
```

(Az eszközfájl elérési útvonalát természetesen az adott rendszernek megfelelően kell megadni.). Az eredménynek körülbelül a következőképpen kell kinéznie:

```
Joystick (Analog 3-axis 4-
  button joystick) has 3 axes
  and 4 buttons.
Driver version is 2.1.0.
Testing ... (interrupt to exit)
Axes: 0: 0 1: 0 2:
  -22892 Buttons: 0:off 1:off
  2:off 3:off
```

Ha ezt látjuk, az már jó jel. Mozgassuk a botkormány karját körbe és nyomogassuk a gombokat. A számoknak és a gombhelyzeteknek (*buttons on-off*) egyaránt változniuk kell. Ha így van, mehetünk tovább. Az eszközből való kilépéshez nyomjunk meg a *Ctrl-C*-t. Ha a fenti szöveg helyett egy hibaüzenetet kapunk, vagy egyáltalán nem látunk semmit, akkor a rendszer nem ismerte fel a botkormányt. Ebben az esetben a *SourceForge*-on (lásd az online irodalomjegyzéket) megtaláljuk a támogatott bemeneti eszköz listáját. Gyakran az is segít, ha beszerzünk valamilyen harmadik fél által gyártott átalakítót. Ezek lehetővé teszik olyan játékvezérlők csatlakoztatását is, mint például a *PlayStation 2 gamepad*.

Általánosságban elmondható, hogy ha egy gamepad csatlakoztatható *USB*-n keresztül, akkor valamilyen módon használható is Linux alatt.

Ha rendelkezünk egy eredeti *Xbox* vezérlővel, azt átalakíthatjuk úgy, hogy csatlakoztatható legyen szabványos *USB* kapura. (Mint korábban említette, ezzel kapcsolatban látogassunk el *Dan Gray* honlapjára.) Arra persze ügyeljünk, hogy az átalakítás után az eszköz igazi *Xbox*-hoz már nem lesz használható. Ezzel szemben az *Xbox 360* vezérlőknek eleve van *USB* csatlakozója. A *Gentoo Linux* terjesztést használóknak javasolom, látogassanak el a *Gentoo Wiki* oldalra (lásd irodalom), itt ugyanis számos hasznos információt szerezhetnek az *Xbox 360* vezérlők használatához. Természetesen más terjesztések felhasználói is megtehetik ugyanezt, de az utasításokat értelemszerűen saját *Linux* változatukhoz hozzá kell igazítaniuk. Meg kell például tanulniuk, hogyan fordíthatják le a rendszermagot forrásból arra az esetre, ha az álta-

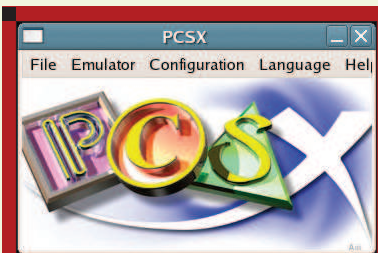
luk futtatott kernelben található *xpad* meghajtó régebbi, mint az, amire a *Gentoo* honlap hivatkozik. (A *Fedora Core 4 2.6.14-1.1656_FC4-i686* sorozatszámú rendszermagjában például a 0.5-ös meghajtó van, ami jóval régebbi annál az 1.6-osnál, ami már támogatja a 360-as játékvezérlőt. Itt tehát garantáltan frissítésre lesz szükség.) Akik *Xbox* vezérlőt használnak, azoknak szükségük van az *xpad* meghajtóra. Miután ezek *USB*-re csatlakoztatható vezérlők, a rendszer automatikusan be fogja tölteni a meghajtót, amikor bedugjuk a gamepad csatlakozóját. Persze ez csak akkor igaz, ha a rendszer megfelelően felismeri a gamepad-et. Az említett *jstest* program itt is jól működik. Ha – legalább nagyjából – biztosak vagyunk abban, hogy a hardverünk jól működik és minden bővítményt megfelelően telepítettünk, akkor végre továbbléphetünk, és beállíthatjuk magát az emulátort.

A PCSX és a bővítmények beállítása

Mint azt korábban is említettem, maga a *PSCX* csak egy alapprogram. Különböző bővítményekre van szüksége ahhoz, hogy bármit elvégezhessen. Éppen ezeket telepítettük az imént. A bővítmények beállításához lépünk be abba a könyvtárba, amit a fájlok kicsomagolásakor hoztunk létre (például `~/Downloads/Pcsx`). Indítsuk el ebből a könyvtárból a *PCSX* rendszert a `./pcsx` parancs kiadásával. Egy dialógusablak jelenik meg, amelyben a program felszólít bennünket, hogy adjuk meg a szükséges beállításokat. Kattintsunk az *OK* gombra a *PCSX Configuration* dialógusablak megnyitásához (1. ábra). A legtöbb ilyen dialógusablakban tulajdonképpen semmit nem kell csinálnunk, ha csak nincs valamilyen speciális igénye az általunk használt hang- vagy videórendszernek, vagy kifejezetten játszadoxni szeretnénk a lehetséges beállításokkal. Éppen ezért én most csak a legszükségesebb beállításokra fogok kitérni.

A legfontosabb dolog, amit be kell állítanunk a konfigurációs panel *Pad* szakasza, kattintsunk tehát ezen a fülön a „*Configure*” (Beállítás) pontra (2. ábra).

Az „*Emulation*” mellett kattintsunk rá a *PCSX* rádiógombra. Ha analóg



■ 3. ábra A PCSX fő ablaka

vezérlőnk van, jelöljük be azt is. Ettől kezdve a megfelelő pontokra kattintva szabadon megadhatjuk, hogy milyen funkciók legyenek hozzárendelve a különböző gombokhoz. Ha például botkormányt használunk, kattintsunk a FEL nyílra a bal oldalon látható négyes blokkban, majd nyomjuk előre a botkormányt. Néhány gombot a billentyűzetre is le lehet képezni, de a lehetőségek száma itt erősen véges, úgyhogy amit lehet, azt inkább magával a vezérlővel oldjuk meg. Ha makrókat akarunk készíteni, használjuk az alul látható M gombokat. Nyomjunk OK-t, ha készen vagyunk a vezérlőgombok hozzárendelésével. Ha két vezérlőnk van, kattintsunk a „Configure” gombra a Pad 2 fülön, és ismételjük meg az eljárást a második vezérlővel is. Győződjünk meg róla, hogy módosítottuk az eszközbejegyzést is. Hogy melyik gamepad tartozik a js0 és melyik a js1 eszközfájlhoz, arról a jstest segítségével győződhetünk meg. A vezérlők beállítása után meg kell mondani a PCSX-nek, hogy melyik BIOS-t használja. Ehhez a BIOS szakaszban válasszuk ki a legördülő menüből a megfelelő nevet. A beállítás ezzel kész is. Ha minden rendben, nyomjuk meg az OK-t, és a PCSX már el is indul (3. ábra).

Előkészületek a játékhöz

A PCSX nem tudja közvetlenül olvasni a PlayStation CD-ket, hacsak nem használunk az általam választottól eltérő bővítményt. Pánikra semmi ok, szándékosan választottam másféle kezelőt. Valójában ugyanis sokkal gyorsabban megy a játék, ha az eredeti játék CD-jéről lemezképet készítünk és annak a merevlemezen tárolt kópiáját használjuk. Ugyanakkor – számos kisebb-nagyobb eltérés miatt – a legtöbb, erre a célra általában használt eszköz most nem jó ehhez a művelet-



■ 3. ábra A PCSX fő ablaka

hez. (Akit ez bővebben is érdekel, nézze meg a *Mega Games* oldalt.) Ezek helyett tehát használjuk a cdrdao parancsot, amivel ISO fájlt készíthetünk a CD „nyers” tartalmából. Legtöbbünk számára a megfelelő parancs így fog kinézni:

```
cdrdao read-cd --read-raw
--datafile frogger.bin
->-device ATAPI:0,0,0 --driver
generic-mmc-raw frogger.toc
```

Itt frogger.bin az elkészítendő adatfájl neve (ezt a CD-t választottam magának a játéknak a bemutatására), a frogger.toc pedig az elkészítendő tartalomjegyzék-fájl. Mindkét fájlunk a játék alapján adtam nevet, hogy később ne kelljen keresgélnem, mi hol van. Az ATAPI:0,0,0 paraméter a legtöbb CD-meghajtóval működni fog.

Végre játszunk

Igen, hosszú menet volt, de végre kipróbálhatunk egy játékot is. Most mindjárt szólok is, hogy nem minden játék fog működni. A Frogger rögtön ment, a *Final Fantasy VII*-tel viszont még mindig küzdök. Persze ez egy jóval bonyolultabb játék... Indítsuk el a PCSX-et pont ugyanúgy, mint az előbb, vagyis lépünk be a Pcsx könyvtárba és írjuk be a ./pcsx parancsot. Most csak a PCSX párbeszédablak jelenik meg. Szükség esetén használjuk megint a „Configuration” menüt a beállítások módosításához. Ha készen állunk a játékra, válasszuk a File menü Run CD pontját, majd keressük meg az imént elkészített .bin és

.toc fájlokat. Válasszuk ki a .bin fájlt a játékhöz és kattintsunk az OK gombra (4. ábra). Kis gyakorlásra lehet szükség, hogy rájövünk a vezérlő beállításaira, de azután már könnyebb lesz.

Győződjünk meg róla, hogy a rendszeren nem futtatunk semmi olyasmit, ami foglalja a háttérben a processzort vagy memóriát. Ezt úgy ellenőrizhetjük, hogy nyitunk egy terminálablakot és a top parancsot beírva megjelenítjük a futó folyamatok listáját. Előfordulhat, hogy nem tudjuk a játék ablakát nagyobbra méretezni, sőt rossz esetben akár össze is omolhat a rendszer, ha erre kísérletet teszünk. Ilyenkor próbáljuk meg egy másik BIOS segítségével futtatni a kérdéses programot. Ehhez válasszuk a File menü Run CD Through BIOS pontját. Ezzel a módszerrel számos, eredetileg vonakodva működő játékot jobb belátásra téríthetünk.

Linux Journal 2006., 145. szám



Dee-Ann LeBlanc

(dee-ann.blog-city.com)
díjnyertes szakíró és újságíró. Szakterülete a Linux, és a kis termetű alaszakai kutyák.

A dee@renaissoft.com címen örömmel vesz mindennemű megjegyzést.

KAPCSOLÓDÓ CÍMEK

www.linuxjournal.com/article/8888