

Linux a Linksys Wi-Fi útválasztóin

Ennek a megbízható és költséghatékony platformnak a meghódítása lehet az első lépés egy vezeték nélküli projekt sikere felé. Ha például egy nagyobb területet akarunk lefedni, összeköthetjük a hozzáférési pontokat, a flash-memóriában nagyobb munkaterületet alakíthatunk ki, mesterségesen fölcsvarhatjuk a teljesítményszabályzót, és így tovább.

A vezeték nélküli hálózati megoldások tömegcikké váltak. Ma már egy *802.11*-es szabványnak megfelelő vagy más néven *Wi-Fi* eszköz teljesen hétköznapi terméknek számít. Több ezer rivális gyártó verseng látszólag megegyező termékeivel a *Wi-Fi* piacon a pénzünkért. Ilyen forrongó területen természetes, ha a gyártók a legolcsóbb megoldásokat keresik. És mit választanak ennek érdekében? Természetesen a *Linux*.

Napjainkra a Linux vált az olcsó, többfunkciós, vezeték nélküli hálózatépítés elsődleges operációs rendszerévé. A terület egyik legnagyobb szereplője, a *Linksys* a következő generációs *802.11g Wi-Fi* eszközeivel a *Linux* felé fordult. Amikor 2003 elején a *Cisco* megvásárolta a *Linksys*-t, hozományként megkapta a linuxos eszközöket, velük pedig egy a kiadatlan *GPL*-kódok felett folytatott elhúzóvívát. A nyílt forráskód híveinek néhány hónapos lobbizása után a *Cisco* megenyhült és nyilvánosságra hozta a forráskódot.

A *Linksys WRT54G* termékét (1. kép) az alacsony ár és a beépített hardver teszi különösen érdekessé.

A *WRT54G*-nek része egy négykapus *Ethernet* elosztó, egy *Ethernet* WAN kapu, támogatja az új nagy sebességű 54 MB/s *802.11g* vezeték nélküli protokollt, és megfelel a korábbi *802.11b* eszközökkel szemben támasztott követelményeknek is.

Mégis az teszi igazán érdekessé a *WRT54G*-t, amit hiányolhatunk belőle. A motorháztető alatt egy 125 MHz-es *MIPS* processzor duruzsol 16 MB RAM-mal. Ennek teljesítménye bőven elég lenne néhány komolyabb alkalmazás futtatására is, miért ne adjunk tehát hozzá néhányat?

A fejlesztőkörnyezet üzembe helyezése

A *Linksys* honlapján található legfrissebb forráskód 145 MB-os és teljes eszközláncot biztosít a *MIPS*-re történő keresztfejlesztéshez.

Kövessük a *WRT54G/src* alkönyvtár *README* nevű fájljának utasításait a közvetett hivatkozás és a *PATH*-kiegészítések létrehozásához, majd lépünk be a *router* alkönyvtárba és futtassuk a *make menuconfig* parancsot.



1. kép A 100 dollár alatti áron elérhető Linksys WRT54G a 16 MB memóriájával és 125 MHz-es processzorával egy alkalmas Linux-platform, amely támogatja a 802.11b és g protokollokat.

Az első fordításunkhoz hagyjuk meg az alapbeállításokat, és kattintgassunk végig a lehetőségeken a saját beállítófájljaink létrehozásához. Lépünk feljebb egy könyvtárszinttel a *WRT54G/src* alkönyvtárba, majd adjuk ki a *make* parancsot. Ennyi az egész. A *WRT54G/image* könyvtárban létrejött egy *code.bin* fájl, amely egy tömörített *cramfs* fájlrendszert és egy 2.4.20-as *Linux* rendszermagot tartalmaz.

Most következnek a dolog rémisztő része: hogyan tudjuk ezt az új *firmware*-t a *Linksys*-re juttatni? Két eljárás is létezik erre, az egyik a *ftpt*, a másik pedig a web-alapú *firmware*-frissítő csatlakozófelület. Azt javaslom, hogy első alkalommal a webes frissítéssel próbálkozzunk. Adjuk meg a *Linksys* dobozunk címét – ennek alapértelmezett értéke *198.168.1.1* –, és jelentkezünk be. Válasszuk ki az *Administration (Felügyelet)* menü *Firmware Upgrade (Firmware Frissítés)* menüpontját, majd töltsük fel a *code.bin* fájlunkat. Az útválasztó újraindul. Gratulálok, éppen most „moddoltuk” a *Linksys*-dobozunkat.

A Ping-trükk

A WRT54G-n lévő *Linux* létezésének felfedezéséhez egy a *Diagnostics (Hibakeresés)* menü *ping* eszközében lévő hiba vezetett el. Az 1.42.2-nél korábbi változatok megengettek tetszőleges kód futtatását a *ping* ablakából, amennyiben *back-tick* operátorok közé tettük. Ha egy korábbi *firmware*-rel rendelkező készülékünk van, próbáljuk meg begépelni az

```
1s -1 /
```

parancsot a *ping* ablak IP-cím mezőjébe. És íme, csodálatos módon megjelenik a gyökérkönyvtár tartalma.

A *ping*-trükk a kíváncsiaknak lehetővé teszi, hogy a forráskód módosítása nélkül fedezzék fel a készüléküket.

A felfedezésnek ez a módja azonban lassú és unalmas. Igazából egy a készüléken futó parancshéjra lenne szükségünk. A *ping*-trükk forráskódban történő kiterjesztésével egy olyan egyedi *firmware*-képfájl hozható létre, amely hálózatos felületen keresztül is rendelkezik a *Linux* héj teljes hatékonyságával. A parancshéj elkészítésének leírását a hálózaton keresztül elérhető forráscímek között találjuk meg.

De miért állnánk meg itt? Az eredeti *firmware cramfs* fájlrendszere 200K szabad helyet hagy a *flash*-memóriában. Helyet találhat itt magának egy sereg hasznos alkalmazás, akár egy *telnet* program, egy *Secure Shell (Biztonságos parancshéj)*, esetleg egy *VPN* ügyfél vagy kiszolgáló.

A wl parancs

Az egyik hasznos parancs, amit a *Linksys* csak bináris formában szolgáltat a *wl*. A *wl* parancs több tucat olyan beépített parancsot tartalmaz, amellyel a vezeték nélküli beállításokat vezérelhetjük, beleértve a népszerű teljesítményt szabályozó beállítást is. A *wl* parancs paraméter nélküli beírásával egy teljes listát kapunk a képességeiről.

A WRT54G alapértelmezett teljesítménybeállítása 28 milliwatt, és ez a beállítás kívülről nem is állítható. A *ping*-trükk vagy egy parancshéj segítségével azonban használhatjuk a *wl* parancsot, paraméterként megadva utána a *txpwr* alparancsot és az 1 és 84 milliwatt közti teljesítményértéket. Ez az érték a következő újraindításig megemeli vagy csökkenti az alapértelmezett teljesítménybeállítást.

A teljesítmény növelése, vagy a beépített antenna cseréje növelheti a kisugárzott teljesítményt, amivel megsérthetjük az erre vonatkozó helyi szabályokat. Ha kicseréljük az antennákat és csökkentjük a teljesítményt, akkor úgy tudjuk jelentősen megnövelni az egység hatótávolságát, hogy ezzel nem sértjük az engedélyezett sugárzási teljesítményre vonatkozó helyi előírásokat.

A WRT54G két külső antennát támogat, és önműködően választ a kettő közül attól függően, hogy melyikkel fogta be a legutóbbi aktív adatsomagot. Amikor egy új, nagyobb hatékonyságú antennát szerelünk fel, nyilván nem ezt a beállítást szeretnénk használni, hanem minden alkalommal a nagyobb teljesítményű antenna használatára szeretnénk a készüléket utasítani. Ezt a bejövő adatok esetén a *wl txant* paranccsal, küldésre pedig a *wl antdiv* paranccsal tehetjük meg. A 0 paraméter a bal oldali antennát jelöli ki, az 1 pedig a jobb oldalt, ha előlről nézzük a készüléket.

Az SSH (biztonsági parancshéj) telepítése

Egy vállalkozó szellemű felhasználó a teljes *OpenSSH* eszközkészletet átültette a *Linksys* készülékébe. Sajnos az *OpenSSH* bináris állományának mérete miatt sok alapvető *Linksys* funkciót is el kellett távolítani a megfelelő méretű szabad hely létrehozásához, ráadásul a kapott RAM-igény is a rendelkezésre álló memória határán volt. Igazából egy kis memóriaigényű *SSH*-kiszolgálóra lenne szükség, s ezt az igényt a *Dropbear SSH*-démont kifejezetten olyan, szűkös memóriával rendelkező eszközökön való futtatásra tervezte, mint amilyen a *Linksys*. Az eredeti *Linksys Linux* megvalósításából hiányzik egy sereg olyan alapvető fájl, ami a többfelhasználós *Linux* rendszereken nélkülözhetetlen. Ezek közül kettő – a */etc* könyvtárban lévő *passwd* és *groups* – a *Linux* alkalmazások többségének futtatásához mindenképpen szükséges. Ahhoz, hogy futtatni tudjuk a *Dropbear* kiszolgálót, ezeket a fájlokat hozzá kell adnunk a *flash*-csomaghoz. Egy jelszóval nem rendelkező root bejegyzést tartalmazó *passwd* fájl és egy megfelelő *groups* fájl létrehozásával már majdnem alkalmas is a környezet a *Dropbear* futtatására. Ezek a fájlok a *flash* képfájl */etc* könyvtárába kerülnek és a *Linksys*-en csak olvashatóak.

Futtatás közben a *Dropbear* ezeken kívül igényli a nyilvános kulcshoz való hozzáférést, amely az *SSH* kapcsolatfelvételhez és hitelesítéshez szükséges, valamint a *known_hosts* fájlt, amely az elfogadandó ügyfélgépek nyilvános kulcsait tartalmazza. A *dropbearkey* programmal egy pillanat alatt létrehozhatjuk a saját kulcsunkat, de a kulcs tárolása a *Linksys*-en már egy kicsit trükkösebb.

A WRT54G egy hash-térképet tartalmaz a kulcsok név-érték páraival, amelyek egy *nvr*-nak nevezett nem felejtő memóriában tárolódnak. A kapott *nvr* segédprogram és *API* lehetővé teszi, hogy írjuk és olvassuk ezt a memóriaterületet. A *Dropbear* saját kulcsát és a saját *.ssh* könyvtárunkban tárolt *id_rsa.pub* fájlban lévő nyilvános kulcsunkat az *nvr*-ban tároljuk és rendszerindításkor innen másoljuk a memóriában lévő virtuális lemezre.

Lefordítjuk a *Dropbear* a kulcsfájl-hitelesítés beállításával, és máris biztonságos módszerünk van a *Linksys*-re történő bejelentkezéshez. Ha jelszavas belépésre van szükségünk, egy folttal kiegészítve a *Dropbear* kódját rávehetjük, hogy a rendszerjelszót olvassa ki az *nvr*-ból, és így lehetővé válik a jelszavas bejelentkezés is.

A Flash-memória tömörítésének növelése

Az *SSH* és a *telnet* programokhoz hasonló eszközök telepítése után hamarosan tapasztalni fogjuk, hogy a *Linksys* *firmware* képfájla feszegetni kezdi az eszközben lévő *flash*-memória tárolókapacitásának határait. Szükségünk lenne egy olyan fájlrendszerre, ami jobb tömörítést nyújt, mint a *cramfs* és egyúttal megfelel a *Linksys Linux* rendszermag elvárásainak is.

Az alapértelmezett *cramfs* fájlrendszer 4K méretű blokkokba tömöríti az adatokat, ez a 4K mérethatár azonban korlátozza az elérhető tömörítési arányt. Ha találunk egy olyan fájlrendszert, ami nagyobb adatblokkokat használ, de a hozzárendelése megfelel az operációs rendszer oldalméretének, sokkal több adatot és programot tudnánk tárolni a *firmware*-ben.

Philip Lougher squashfs fájlrendszere 32K méretű adatblokkokat használ a tömörítéshez és megfelel a 2.4 és 2.6-os rendszermagoknak. Ha szerét ejthetnénk a *Linksys firmware cramfs*-ről *squashfs*-re történő átültetésének, talán elég helyünk lenne arra, hogy egy *VPN* ügyfelet és kiszolgálót is telepítsünk a rendszerre.

A *Linksys* rendszermagja egy 2.4.20 verziójú rendszermag, amelyet a *Broadcom* írt át. A *Broadcom* cég vezető szerepet tölt be a 802.11g áramkörtől a rádióáramkörig és a *WRT54G*-ben lévő központi egység és rádióáramkör is a munkáját dicséri. A *squashfs tar*-fájlja tartalmaz foltokat a 2.4.20-tól a 2.4.22-ig terjedő rendszermagokhoz, de sajnos ezek közül egyik sem alkalmazható teljesen a *Broadcom* rendszermag-fájához, ezért egy kis manuális beavatkozásra is szükség lesz. A legkevesebb hibát tartalmazó folt a 2.4.22-es változat, ez az alkalmazásakor csak egy kódreszletet hiányol. A folt fájljának elolvasásával és a hiányzó darab megkeresésével kézzel módosíthatjuk a kódot. A *Sveasoft* honlapján található egy *WRT54G*-hez készült *squashfs*-foltot is.

A következő lépés a *Broadcom* rendszermag indító kódjának a szerkesztése és egy *squashfs* ellenőrző rész hozzáadása. A *do_mount.c* fájl szinte ugyanazt a kódot tartalmazza és jól használható, amikor az *arch/mips/brcm-boards/bcm947xx* könyvtárban lévő *startup.c* fájl frissítjük.

A rendszermag megfoltozása után az útvalasztó *Makefile*-jét kell megfoltozni oly módon, hogy hozzon létre egy *squashfs*-képfájlt, és a Linux rendszermag beállításába a *squashfs* támogatását is bele kell foglalni. Az elért ered-

mény bőven megéri a befektetett munkát: az újrafordítás után mintegy 500K-val több szabad helyet kapunk az eredeti *cramfs* fájlrendszerhez képest.

A vezeték nélküli elosztórendszer

A normál *WRT54G* egy vezeték nélküli hozzáférési pont (AP), ami azt jelenti, hogy képes más vezeték nélküli ügyfelekkel való kapcsolattartásra, azonban más vezeték nélküli hozzáférési pontokkal már nem. A *wl* paranccsal elérhető, hogy a *Wireless Distribution System (WDS, vezeték nélküli elosztórendszer)* segítségével más hozzáférési pontokhoz kapcsoljuk, vagy hogy vezeték nélküli ügyfélként működtessük.

A *Wireless Distribution System* egy olyan *IEEE* műszaki leírás, amely lehetővé teszi, hogy egy nagy távolságokat áthidaló hálózatban összekapcsoljuk a vezeték nélküli hozzáférési pontokat. Habár ezért némi teljesítményvesztéssel kell fizetnünk, a kapott kiterjesztett vezeték nélküli hálózat sokkal nagyobb hatótávolságú, mint amilyen a külön álló hozzáférési pontok alkalmazásával elérhető lenne.

Két hozzáférési pont *WDS*-sel történő összekapcsolásához ismernünk kell a készülékek egyedi *MAC*-azonosítóit. Jelentkezzünk be az egyes készülékekre, és futtassuk a *wl wds [MAC-cím]* parancsot megadva a másik készülék adatát. Ennek hatására egy új, *wds0.2* eszköz jelenik meg a készülékeken, amelyhez most már IP-cím rendelhető. Ha elvégeztük az IP-címek hozzárendelését és a két készülék közti útvonalválasztást is beállítottuk, akkor a *ping* paranccsal ellenőrizhetjük is az összeköttetést a két készülék között.



A Linksys WRT54G forrásfa

Amikor kibontjuk a *Linksys*ől származó *GPL* forrást, a *WRT54G* főkönyvtár alatt egy könyvtárszerkezet jön létre. A következőkben ismertetem ennek fontosabb részeit.

A *tar*-csomag főkönyvtára a *WRT54G*. A fő *Makefile* a */release/src* könyvtárban található. A forrás kicsomagolása után olvassuk el az itt található *README* fájlt a fordításhoz szükséges tudnivalókról.

A *Linksys* egységgel kapott minden program a */release/src/router* könyvtárból került lefordításra. Ha további programokat szeretnénk hozzáadni, azt itt tehetjük meg és módosítanunk kell az itt található *Makefile*-t. A *Linux* rendszermag forrásfáját a *Broadcom* cég módosította, amely a *WRT54G*-ben található áramkörtől a processzor gyártója. A rendszermag módosításainkat és foltjainkat itt, a */release/src/linux/linux* könyvtárban kell létrehozni.

Egy közvetett hivatkozást kell létrehozni a */opt* könyvtárból az itt lévő */tools/brcm* könyvtárra. A *brcm* alatti két könyvtárat hozzá kell adni a *PATH* változóhoz. Erről részletesen a *README* fájlban olvashatunk.

A foltok és a frissített forráskód letölthető a *Sveasoft* oldaláról.

Minden egyes *WDS*-kapcsolat a hálózaton belüli adatforgalom megkettőződésével jár. Mivel a *802.11g* protokoll félduplex jellegű, ez a hálózati átvitel feleződését jelenti. Amennyiben a hozzáférési pontok 54MB/s sebességen működnek, ez nem nagy áldozat, amíg a kapcsolatok száma három vagy annál kevesebb.

Ügyfél-módú összeköttetés

Az összekapcsolás egyszerűbb formája az egyik készülék ügyfélként való beállítása és egy hozzáférési ponthoz való csatlakoztatása. Ezt *Ethernet* hídnek hívják és számos eszköz között kifejezetten ezzel a céllal működtek.

Az ügyfélmódot a *Linux* rendszermagban kell kiválasztani és ezzel a beállítással kell a rendszermagot lefordítani. Ha ezzel készen vagyunk, a rendszermagot a *Broadcom* egy bináris moduljával kell összeépíteni, amely tartalmazza mind az AP-, mint az ügyfélmódot. A *wl ap 0* ügyfélmódba kapcsolja a készüléket, míg a *wl join [SSID]* egy másik hozzáférési ponthoz köti. Ha az ügyfélen az útvonalválasztás alapértelmezett átjárójának a hozzáférési pont IP-címét adjuk meg, az ügyfél azonnal a hozzáférési pontra irányítja az adatokat és a híd aktívvá válik. Több AP-ügyfél pár is beállítható, ami egy másik lehetőséget kínál a fent vázolt *WDS*-módszer mellett.

A nyílt forráskód hatékonysága

A *Linux* megtalálta a helyét a szuperszámítógépektől a beágyazott rendszerekig minden területen, így

a *Linksys* eszközökben is. A *Linuxra* történő átállás a kiélezett versenyt folytató piaci szereplőkkel szemben támasztott nagy teljesítmény és alacsony költségigény eredménye. Sok hasonló vezeték nélküli útvonalválasztó, így a *Belkin F5D7230-4*, a *Buffalotech WBR-G54* és az *ASUS WL-300g* és *WL-500g* használ *Linuxot* a firmware-jében, és a lista napról napra bővül. Sajnos egyik cég sem tesz eleget a *GPL* követelményeinek, és nem szabadítja fel a forráskódot. A jogi kifogások mellett ezek a termékek lehetőségeikben és szolgáltatásaikban is rövid időn belül messze le fognak maradni a *Linksys* nyílt forrású termékei mögött. A *Linksys* firmware-csomagjai naponta jelennek meg meglepő új lehetőségekkel. Az írásom idején léteznek a *Linksys WRT54G*-hez olyan firmware-fordítások, amelyek támogatják a *VPN*-eket, teljesítményszabályozást, antenna-kiválasztást, ügyfél- és *WDS*-módot, sávszélesség-felügyeletet és még rengeteg mindent, amely különféle forrásokból érhető el. A világháló és a nyílt forráskód összekapcsolásával, otthon vagy kis irodákban alkalmazható útvonalválasztóból hatékony sokfunkciós eszköz hozható létre.

Figyelmeztetés: kísérleti firmware használatával tönkretelhetjük a készülékünket és a *Linksys* garanciáját is elveszíthetjük. Ha alkalmi felhasználók vagyunk és csak otthoni vagy irodai vezeték nélküli hozzáférésre van szükségünk, jobb, ha nem kísérletezünk a leírtakkal, hanem ehelyett a *Linksys* gyári firmware-jét használjuk. Ha viszont hajlandóak vagyunk még a készülékünket is kockára tenni egy kis kísérletezésért, sokkal többet kihozhatunk a dobozból, mint amit a csomagolásán lévő leírásban olvashatunk – köszönet érte a *Linuxnak* és a nyílt forrású fejlesztésnek.

Linux Journal 2004. augusztus, 124. szám



James Ewing (james.ewing@sveasoft.com) több mint 20 éve vállalkozó és szoftverfejlesztő. Kaliforniából egy évtizede költözött át Svédországba, ahol jelenleg próbálja az idejét igazságosan elosztani a felesége, két gyermeke és a *Muppet Show* svéd szakács szerepének gyakorlása közt.

KAPCSOLÓDÓ CÍMEK

- <http://openwrt.kstilebo.net/>
- <http://sourceforge.net/projects/openwrt/>
- <http://www.linksys.com/products/product.asp?prid=508&scid=35>
- <http://www.batbox.org/wrt54g-linux.html>
- <http://dg834.grandou.net/>
- <http://openbrick.org/>
- <http://www.soekris.com/>
- <http://meshcube.org/english/specs.html>