

Bemutatkozik a SHOUTcast

Saját internetes rádióállomást mindenkinek!

Sorozatunk előző részében igyekeztem felvázolni az internetes műsorszórás alapjait. Mít érne azonban önmagában a száraz elmélet, ha nem kóstolhatnánk bele a gyakorlat zamatos világába? Éppen ezért jelen cikkünkben a dolog közepébe vágunk, s a Nullsoft SHOUTcast nevű programjának segítségével felépítünk egy, az internetről vagy a helyi hálózatról elérhető rádióállomást – hiszen az előző számban megjelent témaleírásnak köszönhetően minden szükséges adatot ismerünk az internetes műsorszórással kapcsolatban.

Azért választottam egy MP3 alapú műsorszórás bemutatását, mert ennek a feltételei állnak legközelebb a széles körű megvalósíthatósághoz, ami a gyakorlatban annyit tesz, hogy bárki kipróbálhatja, akinek a gépe rendelkezik valamilyen egyszerű hálózati csatlakozással. Ilyenkor csak annyi a dolgunk, hogy telepítjük a gépünkre a rádióállomást, majd egy másik masinával, amely TCP/IP-protokollon keresztül képes látni a miénket, hallgatóságként csatlakozhatunk az állomáshoz. Megjegyzem, a saját gépünkről is csatlakozhatunk a gépünkön futó rádióállomás-szolgáltatáshoz, de ugye az élmény kedvéért mégiscsak szebb lenne, ha fizikailag is egy másik ügyfél hallgatna bennünket. Ha valamilyen széles sávú kapcsolattal bírunk (ADSL, kábeltéves kapcsolat), nyugodtan szóljunk a szomszédnak, hogy próbaképpen csatlakozzon a gépünkhöz. A feltöltésre engedélyezett 16 kb elég ahhoz, hogy egy, legfeljebb két ember bárhol az internetről csatlakozzon a gépünkhöz. A modemes kapcsolat sajnos nem elég, de igazából egyáltalán nincs szükségünk internet-elérésre; megteszi, ha van egy helyi (LAN) kapcsolatunk a szobában, a lakásban, egy kollégiumban, netán össze vagyunk kötve a mellettünk lévő lakással, s él a TCP/IP-kapcsolat.

Mi is a SHOUTcast?

A program egy ingyenes adatfolyam-előállító és kiszolgáló, amit a Winampot is fejlesztő Nullsoft készített – alapvetően a windowsos Winamphoz, ám a

program linuxos változata is elérhető. A program valójában két részből áll: az egyik a SHOUTcast kiszolgáló, amely a számára előállított adatfolyamot a távolról kapcsolódó gépek számára elérhetővé teszi, a másik része pedig egy olyan adatfolyam-készítő program (ez a mi esetünkben a TRANScast), amely kifejezetten a kiszolgálónak szolgáltatja az adatokat, ez pedig szétszórja azokat az interneten. Mindkét program elengedhetetlenül szükséges a működéshez, de nem muszáj egyazon gépről futniuk – csupán annyi kell, hogy a hálózaton keresztül lássák egymást. A SHOUTcast kiszolgáló egyébként nemcsak Windows vagy Linux, de FreeBSD, MacOS és Solaris felületeken is futtatható, a hozzá kapcsolódó adatfolyam-szolgáltató segédprogram tudása azonban különbözik az egyes rendszereken.

Telepítsük a programot!

A programok megtalálhatók a magazin 54. CD-mellékletének Magazin/SHOUTcast könyvtárában. A *shoutcast-1-9-2-linux-glibc6.tar.gz* nevű állomány tartalmazza a SHOUTcast kiszolgáló, az *sc_trans_posix_040.tgz* pedig a hozzá tartozó TRANScast nevű adatfolyam-szolgáltató programot. Egyébként mindkettő a

☞ <http://www.shoutcast.com> címről tölthető le, az újabb változatok keresése során is itt járhatunk sikerrel. Csomagoljuk ki az állományokat bármilyen tetszőleges könyvtárba (külön a két állomány tartalmát). A SHOUTcast kiszolgáló egy binárisból és egy beállításállományból áll, hasonlóképpen, mint a TRANScast, de a TRANScast rögtön három bináris állományt is magában foglal, amelyek közül mi az *sc_trans_linux* nevűt fogjuk majd használni.

A kicsomagolás után – ami egyben a telepítés – először testre kell szabnunk a kiszolgálót. Kezdjük is el! Nyissuk meg a kicsomagolt könyvtárban található *sc_serv.conf* fájlt, ez sok-sok megjegyzéssel tűzdelve tartalmazza a kiszolgáló beállításait. A seregnyi beállítási lehetőségre való tekintettel előbb csak azokat vesszük sorra, amelyekkel kapcsolatban feltétlenül vagy nagyon



A SHOUTcast hátátája

ajánlott, hogy a működtetéshez testreszabjuk őket. Az egyes kényelmi szolgáltatásokra és a különleges szolgáltatások beállításának mikéntjére sorozatunk következő részében fogunk kitérni.

A SHOUTcast kiszolgáló alapvető beállításai

- **MaxUser:** az elejéről haladva az első beállítás a MaxUser nevű lehetőség, amely azt mondja meg, hogy legfeljebb hány hallgató csatlakozhat az állomásunkhoz. Ezt célszerű a rendelkezésre álló sáv szélesség és a később sugározni kívánt adatfolyam méretének a függvényében beállítani. Számoljuk ki, hogy hányszor fér bele egy adott sávba egy adott bitráta, és aszerint módosítsuk. Minden futó kiszolgálópéldány pontosan egy bemeneti adatfolyamot fogad, és annyi kimenőt engedélyez, amennyit itt beállítunk.
- **Password:** itt tároljuk a kiszolgáló jelszavát. Csak olyan szolgáltatóprogram csatlakozhat a kiszolgálónkhoz, amelyik ismeri ezt a jelszót. Később látni fogjuk, hogy a TRANScast megfelelő helyén majd szintén be kell állítani a csatlakozási jelszót! Ezzel védekezhünk az ellen, hogy illetéktelen személy hozzákapcsolja az általa előállított adatfolyamot a mi kiszolgálónkhoz.

Miért késik a hang?

Annak érdekében, hogy a pillanatnyi hálózati kimaradások vagy ideiglenes sávszélesség-csökkenések miatt a hallgatott adásban ne legyenek kiesések (szaggatás), a lejátszó-programok a bejövő adatokat átmenetítárazzák, így ha kisebb fennakadás keletkezik az adattovábbításban (s mint tudjuk, ez a mindennapokban elég gyakori), addig az átmeneti tárolóban lévő hanganyagot játssza le, s amint újra helyreáll a kapcsolat, újra feltölti az átmeneti tárat. Ennek eredményeképpen mi, hallgatók mit sem veszünk észre belőle. Ehhez jön még az is, hogy a rádióállomás is alkalmaz némi átmeneti tárazást. Kivételes esetekben további kieséseket okozhat, hogy továbbító (relay) kiszolgálók vannak a láncba iktatva, esetleg újra is tömörítik az adatfolyamot, s ez azt eredményezi, hogy nem mindegyik hallgató hallja szinkronban az adást. Mindennek a késleltetésnek nálunk akkor jelentkezik a hatása, ha a saját kiszolgálónkra csatlakozva tanulmányozzuk annak viselkedését: valamilyen változtatást végzünk az adásban, s figyeljük, hogy ez mit eredményez a hangszórókon. Ebben az esetben ne lepődjünk meg, ha a változtatás (új szám betöltése) csak 30–60 másodperc múlva történik meg. Ennyi kell ugyanis ahhoz, hogy az előre betárazott adatfolyam az átmeneti tárból kicserélődjék, s a mi fülünkhöz is már az új adás jusson el.

Ez a mező nem is maradhat üresen, kötelező kitölteni.

- **PortBase:** ez határozza meg, hogy melyik kaput használja a kiszolgáló. Erre a kapura csatlakozik mind a bemenő adatfolyam (amelyet a TRANScast hoz létre), mind a hallgatók. Fontos, hogy a beállított kapu és a tőle eggyel nagyobb sorszámú is szabad legyen, tehát egyetlen más program se használja az adott gépen. Ha egy gépen esetleg több rádióállomást is szeretnénk üzemeltetni, különböző kapuszámmal kell ellátnunk őket, s a csatlakozó hallgatóság e kapuszám alapján választhatja meg, hogy melyik szolgáltatást veszi igénybe. Például ilyen több példányban futtatott kiszolgálókkal tehetjük meg azt, hogy egyszerre sugárzunk egy jobb és egy gyengébb minőségű adatfolyamot. Akinek lassabb a kapcsolata, az a gyengébb adatfolyamot választja, így azok sem esnek el a szolgáltatástól, akik jó sávszélességgel rendelkeznek, s fontos számukra a minőség. (Az alapértelmezett kapuhoz nem kell hozzányúlnunk, nekünk megfelel ez a 8000-es kapu.)

Ezennel a fő beállítások végére is értünk, ezek után már csak el kell indítanunk az ott található binárist, és hagyni, hadd fusson – amennyiben nincs bemeneti adatfolyamunk, úgysem foglal le erőforrásokat.

(Megjegyzés: nem rendszergazdaként indítva a program futása rendszeresen megszakad, ha egy adatfolyamforrás csatlakozni szeretne hozzá.) Most azon-

ban nézzük adatfolyam-szolgáltató programunkat, a TRANScastot!

A TRANScast

Ez a program arra szolgál, hogy a számára megadott forrásanyagot MP3 formátumúvá alakítsa, majd folyamatosan átküldje a már futó SHOUTcast kiszolgálónak. Ez az alkalmazás készíti el magát a rádióadást. A program alapvetően kétféleképpen működik: az egyik működési módban az általunk összeállított lejátszási lista tartalmát olvassa be, és adja tovább a kiszolgálónak, a másikban egy másik SHOUTcast-kiszolgálótól veszi át az adatot, kisebb méretűre (rosszabb minőségűre) tömöríti át, majd továbbadja az általunk futtatott SHOUTcast kiszolgálónak, így azok is élvezhetik az adást, akik lassúbb interneteléréssel csatlakoznak a gépünkhöz. Ha még mélyebbre ásunk, akkor láthatjuk, hogy a lejátszási listából történő adattovábbításnak is kétféle módja létezik: az egyik során MP3-fájlokat olvasunk be, s azt adjuk tovább, a másik mód csak linuxos rendszereken érhető el, ennek során az alkalmazás a hangkártya vonalbemenetére csatlakoztatott adatfolyamot alakítja át MP3-má. Ez azért jó, mert így például „élő” adások sugárzására is lehetőségünk nyílik, ugyanis a hangkártyára egy kisebb „stúdiót” köthetünk. A stúdió egy keverőpultból állhat, amelynek bemeneteire mikrofont, CD-lejátszót vagy egy másik számítógépet köthetünk – ezzel készíthetjük el az adást, s az így előállt műsorfolyamot kapja meg a kiszolgálógép.

Maga a program három részből áll:

a linuxos binárisból, a hozzá tartozó beállításfájlból, végül a lejátszási listafájlokból, amelyek vagy a hangkártya bemenetét, vagy az MP3-as fájlok elérési útvonalait tartalmazzák, egymás alatt felsorolva.

A lejátszási lista beállításai

Mindenekelőtt kezdjük ezzel! Javaslom, tegyünk pár MP3-at a háttértárolóra, majd hozzunk létre egy lejátszási listát. A lista neve tetszőleges lehet, ugyanis később a TRANScast beállításai között mondjuk majd meg, hogy melyik állományt használja lejátszási lista gyanánt. A fájl formátuma egyértelműen kiderül az *example.lst* állományban található megjegyzésekből is. Egymás alatt soroljuk fel őket a teljes elérési úttal együtt. Hogy egy album esetén ne mindent egyenként kelljen felvinni, könnyítésként egy könyvtárlistát is beleíranyíthatunk.

Abban az esetben, ha a hangkártya vonalbemenetét szeretnénk használni, csak egyetlen sort írjunk az MP3-fájlnemek helyére: *DSP:/dev/audiod*. (Ha a hangkártya másképp lett beállítva, természetesen a vonalbemenet-eszköz kezelőállományát kell megadni.) Figyelni kell arra, hogy a program a sorrendben legelső MP3-at valamiért kihagyja, tehát első körben egyszerűen nem játssza le.

A TRANScast beállítási lehetőségei

A program beállításai az *sc_trans.conf* fájlban találhatóak. A változtatások elvégzéséhez nyissuk meg ezt a fájlt, és nézzük, mit is állíthatunk be:

- **PlaylistFile:** itt adhatjuk meg, hogy a program melyik fájlt kezelje lejátszási listaként. Kötelező megadni, hacsak nem továbbító üzemmódban használjuk. Adjuk meg annak a fájlnak a nevét, amelyet létrehoztunk (ha az *example.lst*-t módosítottuk, ezt az értéket hagyjuk változatlanul).
- **ServerIP, ServerPort:** a SHOUTcast kiszolgálót futtató gép IP-címe és kapuszáma (lásd *A SHOUTcast kiszolgáló alapvető beállításai* című részben). A mi esetünkben a két program ugyanazon a gépen fut, így a ServerIP értékét állítsuk *localhost*-ra, a kapuszámot pedig arra az értékre, amelyet a kiszolgálónál beállítottunk. (Ha ott nem nyúlunk hozzá, akkor itt se tegyük!)
- **Password:** a jelszó a SHOUTcast kiszolgálóhoz, amely az adatfolya-

Jelek

A jelek (signals) valójában a folyamatok közötti kapcsolattartás szabványos eszközei, amelyek a legtöbb operációs rendszerben általánosan elfogadott üzenetküldési lehetőséget nyújtanak. Nincs ez másként a Linux, illetve Unix-rendszerekben sem; ezekben a `kill` parancs segítségével küldhetünk üzeneteket az egyes folyamatoknak, amelyek a rendszermagon keresztül jutnak el az adott folyamathoz. Eközben a rendszermag ellenőrzi a jel jogosultságát az adott folyamathoz, vagy önmaga avatkozik be. Erről a parancsról már biztosan sokan hallottak, a legtöbben arra használták, amire a neve alapján elsőként gondolunk: a folyamatok „megölésére”, vagyis futásuk megszakítására. Ekkor ugyanis egy olyan jelet küldünk a folyamatnak, amelynek a tartalma a következő: fejezd be a futásodat! (Meggjegyzem, hogy ez igazából egy különleges jel, amelyet a rendszermag maga értelmez és hajt végre.) Használatának módja a következő:

```
kill -s <üzenet> <folyamat azonosítója>
```

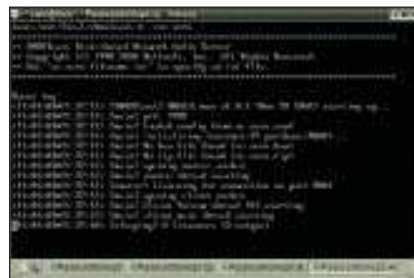
munkát elérhetővé teszi a kapcsolódó hallgatók számára. Azt a jelszót adjuk meg, amit a SHOUTcast kiszolgáló beállításánál már korábban szerepeltettünk. Figyeljünk oda a kis- és nagybetűkre! Ezzel a rendszerszintű beállítások végére is értünk, az ez után következő beállítások az adatfolyamra vonatkoznak majd.

- `StreamTitle`: ez az adásunk címe, az itt beállított érték látszik a hallgatónál a *cím* rovatban, miután csatlakozott hozzánk.
- `StreamURL`: ez egy adathivatkozás, amely a hallgatónál szintén megjelenik, és ezen a weblapon biztosíthatunk számunkra egyéb adatokat az adásról.
- `Shuffle=0/1`: ha ennek az értéke nem nulla, akkor a lejátszási lista tartalmát nem a felsorolt fájlnevek sorrendjében játssza le, hanem véletlenszerűen választ a repertoárból.
- `BitRate/SampleRate/Channels`: ezekkel az értékekkel befolyásolhatjuk az elkészülő MP3-folyam jellemzőit, a minőségét, s ezzel együtt méretét. A `BitRate` mutatja meg, hogy egységnyi hosszon mennyi adatot tárolunk a zenéből, a `SampleRate` a mintavételezési frekvenciát jelenti, míg a `Channels` értéke a csatornák számát. Ez magyarul annyit tesz, hogyha a `Channels` értéke 1, akkor a program mono-, s amennyiben az érték 2, akkor sztereoadást készít. Attól függően, hogy milyen kapcsolatunk van és hol szeretnénk sugározni, állítsunk be valamilyen értéket. Ha helyi hálózaton (LAN) sugárzunk, célszerű a `128000/44100/2` (`BitRate/SampleRate/Channels`) hármast használni, lassabb kapcsolat esetén felezzük, negyedeljük az

értékeket, s monoadatfolyamot sugározzunk.

Kapcsoljuk be a rádióállomást!

Ha ezekkel az alapbeállításokkal végeztünk, nincs más dolgunk, mint futtatni az adatfolyam-készítő programunkat, amely az adást továbbítani fogja a kiszolgálónak, amihez már csatlakozhatnak is a hallgatók. Jelen esetben nekünk az `sc_trans_linux` nevű binárist kell futtatnunk. Elindítás után láthatjuk, hogy csatlakozik a másik szálon futó kiszolgálóhoz. A másik szála átkapcsolva máris láthatjuk, hogy a kiszolgáló működik, s azt is megmutatja, hogy hány hallgató csatlakozott hozzá. Ha semmilyen hibáüzenetet nem kapunk, az azt jelenti, hogy a gépünk máris rádióállomás. Ellenőrzésként indítsunk el egy XMMS-t, és fájl helyett most a `http://localhost:8000` URL-t adjunk hozzá a lejátszási listához, amely a saját gépünkön futó SHOUTcast kiszolgáló címe. Ezek után hallanunk kellene a hangszórókból kiáramló muzsikát, a zene címében pedig az általunk beállított rádióadás címét kell látnunk. Ha valóban működik, a kezdeti sikereken felbuzdulva próbáljunk meg más gépekről is csatlakozni a kiszolgálóhoz, s élvezzük az éterben megszületett új lehetőségeket. Miután egy kicsit megnyugodtunk, vizsgáljuk meg a hiányosságokat is. Tegyük fel, hogy folyamatos adást szeretnénk, ugyanakkor változtatásokra van szükség a lejátszási listában: másik számot szeretnénk elindítani, esetleg jó lenne, ha a programunk véletlenszerűen, és nem sorban játszaná le a zenéket stb. Mivel azonban a TRANScast a beállításokat csak indításkor olvassa be, első megközelítésben a lejátszási lista vagy a beállításfájl átírása után újra kellene indítanunk a programot, amely



A kiszolgáló indítása



Az ügyfél indítása

mindenképpen egy kis szünetet okozna az adásban – arról nem is beszélve, hogy a kiszolgáló bizonyos idő eltelte után alapértelmezetten eldobja a hallgatók által létesített kapcsolatokat. Hogyan lehetne ezt másképp megoldani? A válasz egyszerű: a fejlesztők is gondoltak erre a problémára, ezért lehetővé tették számunkra, hogy éppen futó TRANScast-folyamatunknak jelek (signals) formájában üzenetet küldjünk.

A TRANScast kezelése jelekkel

A programot úgy írták meg, hogy futás közben lekezeljen bizonyos üzeneteket, amelyek segítségével a fent említett módosítások elvégezhetők. Ahhoz, hogy ilyen jelet küldjünk a programnak, szükségünk van az indítása során a rendszermagtól kapott folyamatazonosítóra (PID), amit indítás után közvetlenül ki is ír a konzolra. Vegyük például az alábbi:

```
<10/27/03@10:36:06>
↳ [MAIN] PID: 2203)
```

Ezek után a `kill` paranccsal a folyamatnak máris olyan üzeneteket küldhetünk, amelyeket kezelni tud, ehhez azonban előbb ismerjük meg ezeket a bizonyos jeleket:

- `HUP`: hatására a program frissíti a naplófájlokat, közben a konzolra történő naplózás leáll.
- `WINCH`: ha ilyen jelet küldünk a programnak, azonnal elkezdi játszani a lejátszási lista következő számát.



A SHOUTcast webes felülete

- **USR1:** újraolvassa a lejátszási listában szereplő elemeket, de eközben a jelenlegi fájl lejátszását nem szakítja meg, ám amikor az véget ér, már az új lejátszási lista elemeit fogja játszani.
- **USR2:** ki-, illetve bekapcsolja a véletlenszerű lejátszási sorrendet. (Ha ki van kapcsolva, a lejátszási listában megadott sorrendnek megfelelően játssza le az elemeket.)
- **TERM:** leállítja a TRANScast adatfolyam-kiszolgálót.

Ezek után, ha a lejátszási listához újonnan hozzáadott zenéket szeretnénk beépíteni az adásba, adjuk ki a 'kill -s USR1 9999' parancsot, ahol a 9999-es PID helyére mindenki a TRANScast indításakor kijelzett négyjegyű folyamatazonosítót írja. Fontos, hogy lehetőleg ugyanannak a felhasználónak a nevében adjuk ki a parancsot, aki másik szálon a programot is futtatja, különben azt a hibaüzenetet kapjuk, hogy nincs jogunk jelet küldeni a programnak. Ha semmilyen üzenetet nem kapunk a kill parancs végrehajtása után, az azt jelenti, hogy a jel eljutott a címzetthez, s az ezt le is kezelte. Ennek ellenőrzéséhez váltsunk át a futó TRANScasthoz, a s a konzolra azt kell kiírva látnunk, hogy az általunk kiadott parancsot az alábbi eredménnyel fogadta:

```
<10/29/03@11:39:46> [MAIN]
↳SIGUSR1; Reload Playlist
<10/29/03@11:39:46> [MAIN]
↳Reloading playlist
<10/29/03@11:39:46> [MAIN]
↳Loading playlist
(example.lst)
<10/29/03@11:39:46> [MAIN]
↳Found (3) entries in playlist
```

Javasolom, hogy az összes jelet próbálgassuk ki, figyeljünk a hatására olyan módon, hogy közben a saját rádióadásunkat hallgatjuk.

A SHOUTcast kiszolgáló mint webkiszolgáló

Az előző cikkben is említettem már, hogy a legtöbb adatfolyam-szolgáltatás a HTTP protokoll felett zajlik, ami azt jelenti, hogy a rádióállomásoknak valójában egy webkiszolgálót kell megvalósítaniuk. A SHOUTcast kiszolgáló esetében is ugyanez a helyzet, s a fejlesztők, ha már úgyis egy csökkentett tudású webkiszolgálót írtak, gondoskodtak arról, hogy egy böngésző segítségével is csatlakozhassunk a kiszolgálóhoz. Ám ekkor nem az adatfolyamot kapjuk, hanem mindenféle többletadatot a rádióállomással, az épp futó adással, a hallgatókkal stb. kapcsolatban. Ezenkívül rendszergazdai feladatok ellátására is lehetőségünk nyílik (bizonyos IP-k letiltása, bizonyos hallgatók kirúgása stb.). A szolgáltatás igénybevételéhez nem kell mást tennünk, mint a böngészőnkben beírni a rádióállomás címét, kapuszámát. Futtassuk még mindig a kiszolgálónkat, a TRANScast segítségével küldjünk neki adatfolyamot, s XMMS segítségével a gépünkről csatlakozzunk saját magunkhoz – a lényeg, hogy legyen működő kiszolgáló, s hozzá valamekkora (akár egy főnyi) hallgatóság. Ekkor indítsunk egy böngészőt, és látogassunk el a <http://localhost:8000> címre. Máris láthatjuk rádióállomásunk jellemzőit. A jobb oldalon található **Admin login** menüpontra kattintva az admin/<beállított kiszolgáló jelszó> párossal azonosíthatjuk magunkat, s elénk tárul a beavatkozó felület, amelynek segítségével felügyelhetjük a rádióállomást. A SHOUTcast kiszolgáló beállításai között megadhatjuk, hogy ez a felület csak kivételes esetben engedjen meg beavatkozást, általában inkább csak adatokat jelenítsen meg.

Áttekintés

Gondolom, senki sem vitatja a megoldás egyszerűségében rejlő nagyszerűségét. Természetesen az egyszerűség alatt nem feltétlenül a sok-sok beállítási lehetőséget értem, hanem azt, hogy pár óra alatt rádióadást vagyunk képesek csinálni! Hol lehet mindez hasznos? Nem szeretnék az élettől távol eső példákkal szolgálni, ezért álljon itt egy ellesett ötlet: képzeljünk el egy kollégiumot, ahol több száz diák lakik, s a közösségi életet egy mindenki által élvezhető rádióadás beindítása színesítené. A legtöbb kollégiumban mára már kiépítették a helyi hálózatokat, amelyek sávszélessége nagyságrendileg 100 Mbit körül mozog. Ez kényelmesen elég ahhoz, hogy a belső

hálózaton hozzuk létre a közösség rádióállomását, amelyhez minden – az adott tartományban lévő – gép csatlakozhat. (Gondoljunk most bele abba, hogy egy hagyományos, kis hatósugarú állomás üzemeltetéséhez mindenféle, általában pénzes engedélyek szükségesek, a „kábel” terjedő adás pedig a kábelzés iszonyatos költségét ruházná ránk.) Mire van mindehhez szükségünk? Egy Pentium II-es processzorral szerelt gépre, erre az ingyenesen használható adatfolyam-kiszolgálóra – és némi rátermettségre, hogy mindezt megvalósítsuk! Ahhoz, hogy ez a szolgáltatás kellő színvonalú legyen, természetesen nem elég ez az egyébként működő összeállítás: nem árt, ha kiszolgálónkat teljesen testreszabjuk. Ehhez szeretne segítséget nyújtani sorozatunk következő része, amelyben az emelt szintű beállításokról és a webes felület használatáról szeretnék szólni.



Komáromi Zoltán

(komi@kiskapu.hu)
23 éves, a BME hallgatója, mellette PHP-programozóként dolgozik. Kedvenc területe a multimédia.

