



Egy nyílt forráskódú PBX-rendszer, az Asterisk

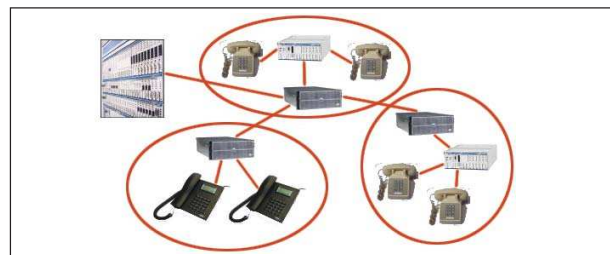
Használj egyetlen rendszert az IP alapú és a hagyományos telefonhang továbbítására, a hangposta vezérlésére, valamint írv CGI-szerű alkalmazásokat a telefonhasználók számára.

Házi telefonközpontot (Private Branch eXchange, PBX) kellene felszerelni irodánkban? Esetleg egy hangposta rendszert szeretnénk üzembe helyezni otthoni linuxos gépünkön? Mit szólnánk egy önműködő hangválaszrendszerhez (Interactive Voice Response, IVR) otthonunk automatizálásához? Esetleg az interneten keresztüli telefonálás lehetősége (VoIP) sem jönne rosszul? Hogyan valósíthatnánk meg mindezt? Az Asterisk egy rendkívül érdekes és hatékony, a GPL felhasználási szerződés alatt terjesztett linuxos PBX-rendszer, melynek segítségével áthidalható az otthon is megtalálható hagyományos telefon és a világhálón keresztül történő hangátvitel közötti szakadék. Az Asterisk ezen felül számos más szolgáltatással is bír, amelyek valóban vonzó választássá teszik. Írásunkban megismerhetünk néhányat e tulajdonságok közül, és elég adatot szerezhethünk be a programról ahhoz, hogy mindenféle különleges eszköz beszerzése nélkül belevághassunk a dologba.

A háttér

Az Asterisk, a Digium által támogatott nyílt forrású projekt, melynek elsődleges gazdája *Mark Spencer*, de számos foltot a közösség készített el hozzá. Írásom időpontjában még kizárólag Intel processzorokon futó Linuxon működik, bár nemrégiben már sikerült bizonyos eredményeket elérni a PowerPC-n futó Linux terén is, és erőfeszítéseket tettek az Asterisk BSD-re történő átültetésére is. A Digium emellett különböző eszközöket is kínál, amelyek együttműködnek az Asteriskkel. Ezek az eszközök olyan PCI-kártyák, amelyek a szabványos analóg telefonvonalat kötik össze a számítógépünkkel. A program más eszközöket is támogat, ilyenek például a Dialogic vagy a Quicknet termékei. Az Asterisk saját, IAX elnevezésű VoIP protokollal rendelkezik, de támogatja a SIP és H.323 protokollokat is. Ezzel el is jutottunk az Asterisk egyik legnagyobb előnyéhez: a különböző technológiákat egy környezetben össze tudja kapcsolni. Az Asteriskon keresztül egy rendszerré köthetjük össze az IAX, SIP, H323 és a hagyományos telefonunkat (lásd az 1. képet).

A fejlesztő ki is terjesztheti az Asterisket a C API használatával, vagy az AGI-k segítségével, ezek a CGI-parancsfájlok megfelelői. AGI-kat bármilyen nyelven írhatunk, ezek



1. kép Az Asterisk képes a hagyományos telefonvonal és a különféle VoIP szabványok szerinti vonalak összekapcsolására

futtatása pedig külső folyamatként zajlik. Ez az Asterisk képességbővítésének legegyszerűbb és legrugalmasabb módja (lásd az 1. listát).

Vágjunk bele!

Egy ideje már nem készült új hivatalos kiadás, de szó van róla, hogy nemsokára megjelenik. Pillanatnyilag az Asterisk megszerzésének legjobb forrása a CVS:

```
export CVSROOT=\
:pserver:anoncvs@cvcs.digium.com:/usr/cvsroot
cvs login (a jelszó: "anoncvs")
cvs co asterisk
```

Ha Digium gyártmányú PCI-kártya használatát tervezzük, a zaptel-re is szükségünk lesz. Ha az összekapcsolhatóság is szerepel célkitűzéseink között, akkor a libpri-t szintén le kell töltenünk.

Beállító parancsfájl nincs, egyszerűen a make parancsot kell használnunk. Az Asterisk sikeres lefordításához szükségünk van a readline, OpenSSL csomagokra és a 2.4-es rendszer-magra a magforráskódokkal együtt:

```
cd asterisk
make clean install samples
```

Ez lefordítja az Asterisket, telepíti és emellett a minta-beállításfájlokat is felpakolja. Az utolsó rész felülírja a már létező beállításfájlokat, ezért ezt vagy ugorjuk át, vagy készítsünk biztonsági másolatot azokról a beállításfájlokról, amiket

meg szeretnénk tartani. A zaptel vagy ISDN használata esetén ezeket még az Asterisk fordítása előtt fordítsuk le. Az Asterisk telepítése alapértelmezetten a `/usr/sbin/` könyvtárba történik, a beállításfájlok pedig a `/etc/asterisk/` könyvtárba kerülnek. A hangposta üzenetei a `/var/spool/asterisk/voicemail/` könyvtárban tárolódnak. A számlázás CDR-jeit (Call Detail Record, azaz hívásadat rekord) és a naplózás fájljait a `/var/log/asterisk/` elérési útvonalon találjuk meg. Az Asterisk indításához az `asterisk` parancsot kell kiadnunk parancssorból, ehhez a kipróbálás szakaszában nem árt hozzáadni a `-vvv` kapcsolókat is. A `-vvv` kapcsoló a működésről a legtöbb üzenetet adó üzemmódot kapcsolja be, a `-c` kapcsoló pedig egy konzolképernyőt ad, amelyen keresztül kapcsolatot tarthatunk az Asterisk munkafolyamatával, például vezérlésre vagy állapotra vonatkozó parancsokat küldhetünk neki.

Az Asterisk tevékenysége és működőképessége több különböző beállításfájlon alapul. Ezek közül ebben a cikkben hármat vizsgálunk meg, de még továbbiak is léteznek. Az Asterisket beállítjuk, hogy a felhasználók az IAX protokollon keresztül hívhatják egymást, továbbá üzembe helyezzük a hangpostát, melynek segítségével a felhasználók lehetőséget kapnak hangüzeneteik kezelésére.

A tárcsázási eljárások

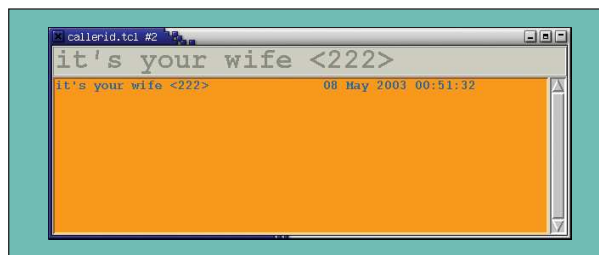
Mielőtt még nekilátnánk az Asterisk üzembe helyezésének, átfogó képpel kellene rendelkezniünk a tárcsázási eljárásokról (dialplan). Habár rugalmas és hatékony eszközről van szó, de könnyen bele is kavarodhatunk. A tárcsázási eljárás a számok értelmezésének meghatározására, valamint az útvonal kijelölésére használatos, s ebből kifolyólag az Asterisk lelkét képezi. A tárcsázási eljárás környezeteket (context) határoz meg, amik azoknak az úgynevezett kiterjesztéseknek (számjegyminták) a tárolói, amelyek meghatározott feladattal bírnak. Például előfordulhat, hogy az irodánk vagy a lakótársaink számára szeretnénk egy bizonyos környezetet meghatározni, amely számukra adott tárcsázási jogosultságokat teremt. Ugyanakkor meghatározhatunk egy külsősök vagy vendégek számára létrehozott környezetet is, ami csak bizonyos korlátozásokkal engedélyezi a tárcsázást, például tiltja a távolsági hívásokat. A környezetek neveit szögletes zárójelek ([]) között kell feltüntetni. Az adott környezetekhez rendelt kiterjesztések a név után írandók. Az egyes kiterjesztések elsőbbséggel (priority) rendelkezhetnek. A hívási folyamat végrehajtása sorrendben történik egészen addig, míg egy alkalmazás `-1` értéket nem ad vissza, ekkor a hívás félbeszakad, vagy az alkalmazás átirányítja a folyamatot. A kiterjesztés bejegyzésének formátuma a következő:

```
exten =>
<kiterjesztés>, <prioritás>, <alkalmazás(paraméterek)>
```

Íme néhány példa:

```
exten => 9911,1,wait(1)
exten => 9911,2,Dial(Zap/1/${EXTEN:1})
```

A kiterjesztést az `exten=>` kifejezés jelöli. Ebben az esetben a 9911 a kiterjesztés, az 1 és a 2 az elsőbbség vagy lépés száma (ezeknek sorrendben kell egymás után kö-



2. kép Grafikus program mutatja a beérkező hívások azonosító adatait

vetkezniük); a `Wait` és a `Dial` pedig az alkalmazások. Az Asterisk külön programokat használ a kiterjesztések egyes lépéseinek végrehajtásához. Ezekről a programokról leírást az Asterisk konzolján keresztül kaphatunk. A `show applications` parancs listát ad a támogatott programokról, a `show application <program neve>` parancs pedig a megadott program használatához nyújt segítséget.

A kiterjesztések használhatók mind a kimenő, mind a bejövő hívószámra, ezzel a hívások feldolgozásakor nagyfokú rugalmasság érhető el. Sablonok használatára is lehetőség nyílik, egy aláhúzás jel (`_`) vezeti be őket:

- N: egyetlen, 2 és 9 közötti számjegy.
- X: egyetlen, 0 és 9 közötti számjegy.
- [12-4]: bármilyen, a zárójelek közt lévő szám.
- .: helyettesítő karakter.

Például a `_NXX5551212` bármilyen adatszámra illeni fog, függetlenül a körzetszámtól.

A kiterjesztés bármilyen alfanumerikus karakterlánc lehet. Lássunk néhány beépített különleges karaktert:

- s: ha nem érkezik be tárcsázott szám, akkor itt kezdődik, mint egy analóg vonalról érkező bejövő hívás.
- t: időtűllépés esetén használatos.
- i: tiltott számok tárcsázása esetén használatos.
- o: operátori kiterjesztés.
- h: felfüggesztő kiterjesztés.

IAX-felhasználók létrehozása

Az első fájl, amit létrehozunk, az `iax.conf` (lásd a 2. listát). Ez a fájl vezérli az IAX protokoll működését, és a protokoll felhasználóit határozza meg. A protokollnak két változata létezik, a régi az IAX, az új pedig az IAX2 nevet viseli. A beállításfájl első szakasza az általános rész, amely az IAX protokoll jellemzőit határozza meg. A felsorolásban négy szerepel, de ennél több is megadható. A port érték annak a kapunak a száma, amelyen az IAX a párbeszédet folytatni fogja. Ez alapértelmezetten az 5036, így akár el is hagyható. A `binaddr` értéket arra használhatjuk, hogy az Asterisk egy bizonyos IP-címhez kapcsolódjon; például több hálózati kártyával felszerelt gép esetén. A `0.0.0.0` binaddr érték minden IP-címhez csatlakozni próbál. Az `amaflags` és `accountcode` értékeket a CDR-ek használják. Amennyiben az általános részben adjuk meg, akkor alapértelmezett értékként kerülnek felhasználásra, de lehetőség nyílik felhasználónkénti megadásra is. Az `amaflags` lehetséges értékei a `billing`, `documentation`, `omit` és a `default`. Az `accountcode` tetsző-

1. lista Példa a Caller ID AGI parancsfájltra

```
#!/bin/sh
# \
exec tclsh "$@" "${1+"$@"}"

set port 10000
set hosts [list 192.168.123.166 192.168.123.168]

##
## Adat küldése a gazdagépnek
##
proc sendInfo {ip_ port_ callerid_} {

    if {[catch {socket $ip_ $port_} sock]} {
        return
    }

    fconfigure $sock -buffering line
    puts $sock $callerid_
    close $sock

    return
}

##
## Az összes változó megszerzése az stdin-ről;
## ezek "agi_" kezdetűek; egy tömb feltöltése
## az értékekkel
##

while {[gets stdin l] > 0} {
    if {[regexp {^agi_(\w+):[\s]+(.*)} $l ->
        ↪ k v]} {
        set AGI($k) $v
    }
}

##
##      A callerid adatainak elküldése
##      az összes felsorolt gépre
##
foreach H $hosts {
    sendInfo $H $port $AGI($callerid)
}
```

leges értéket vehet fel. A mostani beállításnál a home kerül a hálózaton belüli felhasználókhöz, és external a hálózaton kívüli felhasználókhöz rendelt érték. Sok más értéket kihagyunk, de ezek nagy része a teljesítménnyel kapcsolatos. A hátralévő szakaszok a felhasználók meghatározásai. Nekem három felhasználóm van: brett, maria és nico. A típusmegadáskor három lehetséges értéket adhatunk meg: a peer minden hívást fogadhat, a user hívásokat kezdeményezhet, a friend-nek pedig mindkettőre lehetősége nyílik. Én mindegyiket friend típusúként adtam meg. Minden gépet dinamikusnak (dynamic) adtam meg, de ha valamelyik gépünk állandó IP-címmel bír, állandó (static) tulajdon-

ságúként határozhatjuk meg. A secret az a jelszó, amit a felhasználónak az Asterisk kiszolgálóhoz való csatlakozáskor kell megadnia. Ebben a fájlban két környezetet adok a felhasználók számára: a [cg1] és a [cg2] nevűeket. Ezeket részletesen az *extension.conf* fájl magyarázata során fogom ismertetni, de lényegében e környezetek adják a felhasználó számára a tárcsázási jogosultságokat.

A hangposta beállítása

A következő fájl a *voicemail.conf* (3. lista). Ennek is van egy általános része, amely a hangposta általános vagy átfogó jellemzőivel foglalkozik. Az első érték, a format, az üzenetek hangformátumait sorolja fel. A következő két értéket az elektronikus levélben történő értesítés használja: a serveremail a forráscím (mezőből), az attach pedig arra utasítja az Asterisket, hogy az üzenetet csatolja a levélhez. Példánkban nem szeretnénk üzenet csatolását, ezért ebben az esetben elhagyunk néhány értéket.

Az <mbbox> az a szám, amelyen a felhasználó számára az üzenetek menthetők és elérhetők. Ezt az *extension.conf* fájlban is használjuk a hívási folyamat megfelelő hangpostafiókba való irányítására. A <passwd> érték megadása az üzenetek lekérdezésekor szükséges, a <name> pedig a felhasználó neve. Az <email> és <pager> értékek olyan elektronikus címek, amelyeket a levél érkezéséről tájékoztató üzenetek küldésére használ a rendszer. A <pager> címre küldött üzenet rövid, mert kisebb eszközökön (személyhívó vagy mobiltelefon) is olvashatóknak kell maradnia. Sok mobiltelefon- és személyhívó-szolgáltató rendelkezik olyan elektronikus-levélátjárókkal, amelyek képesek az üzeneteket az eszközre továbbítani.

A kiterjesztések meghatározása

A jelenleg vizsgált fájlok közül az utolsó az *extension.conf* fájl (lásd a 4. listán, mely az 59. CD Magazin/Asterisk könyvtárban található). Ez az egyik legbonyolultabb fájl, mivel a tárcsázási eljárásokat is tartalmazza. A példánkban látható tárcsázási eljárás képességeihez viszonyítva meglehetősen egyszerű. A fájl egy *general* és egy *global* nevű szakasszal rendelkezik. Az általános szakasz hasonlít a korábbi fájlok általános szakaszára, vagyis általános értékeket határoz meg. Ebben a példában nem adtam meg általános értékeket. Az átfogó szakasz a teljes körű változók meghatározására használatos. Ezek a változók a tárcsázási eljárásokban a \${VÁLTOZÓ} utasításforma használatával érhetők el. Én egy változót határoztam meg TIMEOUT néven a válaszüzár-túllépésére. Az olyan beépített változók, mint a CONTEXT, EXTEN és CALLERID szintén használhatók a tárcsázási eljárásokon belül. A többi szakaszok mind környezet-meghatározások. A környezet egyszerűen számjegyminták egy csoportja. A példa esetében több környezetet is megadtam, ezek mindegyike valamilyen tárcsázási esetet – hangposta, iax és munkaidő utáni hívás – határoz meg. Ezeket úgy kell elképzelni, mint mini tárcsázási terveket. Ezek után két olyan környezetet adtam meg, amelyeket a felhasználókhöz rendeltem. E környezetek öröklik a már korábban meghatározott, az include kulcsszó használatával megadott környezetek adottságait.

Az első környezetet a voicemail, azoknak a számjegymintáknak a listája, amelyekkel a felhasználók érhetik el hangpos-

2. lista Az iax.conf fájl tartalma

```
[general]
port=5036
bindaddr=0.0.0.0
amaflags=default
accountcode=home

[brett]
type=friend
host=dynamic
secret=brettsecret
context=cg1
callerid="brett <111>"

[maria]
type=friend
host=dynamic
secret=mariasecret
context=cg1
callerid="maria <222>"

[niko]
type=friend
host=dynamic
secret=nikosecret
context=cg2
accountcode=external
callerid="Niko <333>"
```

tájukat. A felhasználók tárcsázzák a 6245 számot, mire a VoicemailMain2 program bekéri tőlük a postafiókjuk számát és a jelszót. A felhasználók itt a postaládájukban lévő üzenetekkel kapcsolatos teendőiket intézhetik el (meghallgathatják, törölhetik).

Az iax-környezet az IAX-felhasználók közti PBX-tárcsázásra ad lehetőséget. Minden felhasználó számára különböző kiterjesztéseket határoztam meg. A felhasználó nevét (maria) tartalmazó kezdés átírnyít a kiterjesztés számának bejegyzésére. Az 111 kiterjesztést szintén a hívóazonosítóval (callerid) egyeztettem. Amennyiben az azonosító megfelel, megváltoztatom a callerid nevét, így ennek viszonylagos jelentése van. Például, ha a tárcsázott kiterjesztés a 111, és a hívóazonosító 222, akkor a callerid nevét „it's your wife!” („a feleséged hív!”) szövegre cserélem. Ez az üzenet jelenik meg mindannyiszor, ha a feleségem hív.

Az utolsó számminta környezet a késői órákban érkező hívások esetén használatos. Mivel nem szeretném, hogy külső hívások zavarjanak éjszaka, bármilyen tárcsázott számra fennáll az egyezés (_). Vár egy másodpercig, fogadja a hívást, majd egy tárolt üzenetet játszik le, így a hívó kiválaszthatja, hogy kinek szeretne üzenetet hagyni („for brett, press 1 –, ha brettnek, akkor nyomja meg az 1-es gombot”). Ha a hívó megnyomja az 1-est, a hívás az 1,1,Voicemail12(111) bejegyzésen folytatódik, amely a felhasználót az 111 postafiókba irányítja. Ez egy egyszerű példa arra, miként hozhatunk létre egy önműködő hívásválaszrendszert.

3. lista A voicemail.conf fájl tartalma

```
[general]
format=gsm|wav49|wav
serveremail=asterisk
attach=no
maxmessage=180
maxgreet=60

;
; voicemail box definitions.
; mbox# => password,name,email,pager/mobile
;
[cg1]
111 => 1111,Brett,brett_schwarz@yahoo.com
222 =>
2222,Maria,maria@foo.com,4255551212@mob.net
```

A [cg1] és [cg2] környezetek olyan szolgáltatásokat foglalnak magukban, amiket más környezetekben is megadtam. Ez lehetővé teszi, hogy könnyedén létrehozassak különböző felhasználói csoportokat. Például a [cg1] minden általam korábban már meghatározott képességgel rendelkezik, viszont a [cg2] csak iax képességekkel, és a munkaidő utáni órákban hívás esetén a hangpostára kerül átírnításra. Az Asterisk tárcsázási eljárásai rugalmasságának kihasználásával hatékony tárcsázási lehetőségeket határozhatunk meg. A bemutatott példa csak felvillantja a benne rejlő lehetőségeket. A tárcsázási eljárásokat a makrók használatával szintén egyszerűbbé tehetjük, de ezt gyakorlásképpen meghagyom az olvasónak.

Az AGI használata

Az *extensions.conf* fájlban egy *callerid.agi* bejegyzés meghív egy AGI-parancsfájlt. Ez egy egyszerű példa az AGI-felület bemutatására. A parancsfájl a */var/lib/asterisk/agi-bin/* könyvtárban helyezkedik el, és az Asterisk hívja meg külső eljárásként. Az AGI és az Asterisk az *stdin*, *stdout* és *stderr* csatornákon tartják egymással a kapcsolatot. Az AGI-nak történő értékátadás az *stdin* csatornán keresztül történik, az AGI pedig az *stdout* csatornán keresztül ad vissza adatokat az Asterisknek. Az Asterisk konzoljának szánt üzenetek az *stderr* csatornára kerülnek. Két érték mindenképpen átadásra kerül az AGI-nak: az AGI teljes elérési útvonala és azok a tulajdonságok (argumentum), amelyeket az AGI az *exten* bejegyzésen keresztül kap meg. Az AGI beolvassa a *callerid*-t, majd átküldi a másik gépen futó grafikus felhasználói programnak. Ez a grafikus felhasználói program a weboldalamról szerezhető be (lásd a 2. képet). Az adatok kikereséséhez AGI-parancsfájlok is használhatók. Amennyiben egy adatbázisból a hívóval vagy a felhasználóval kapcsolatos adatot kell kikeresnünk, használhatjuk az AGI felhasználói felületét is.

Hívjunk fel valakit

Mihez kezdhetünk az elvégzett beállításokkal? Miután létrehoztuk a fenti beállításfájlokat és elindítottuk az Asterisket (*asterisk -vvvc*), megpróbálkozhatunk a híváskezde-

ményezéssel. Jelen pillanatban az IAX ügyfélprogramok elérhetősége korlátozott. A SIP-ügyfelek, amilyen a kphone és az xten és a Cisco, SNOM és egyéb gyártók eszközei elérhetőek és együttműködnek az Asteriskkel, de ebben a cikkben az IAX használatára összpontosítottam. A legrégebbi ügyfélprogram a Digium által kifejlesztett Gnophone. Jelenleg több felületen futó, windowsos ügyfélprogramon is dolgoznak. Hozzáférhető még a SourceForge tel projektjéhez tartozó ügyfélprogram is. Ennek az ügyfélprogramnak módosítottam a felhasználói felületét. A program még próbaállapotú, de már működik. Ezt az ügyfélprogramot használtam, amikor Németország (*Reinhard Max*), Ausztrália (*Steve Landers*) és az Egyesült Államok (jómagam) között hoztam létre összeköttetést. Bármelyik ügyfélprogram mellett döntünk is, egy felhasználói nevet, jelszót és egy környezetet meg kell határoznunk minden olyan Asterisk-kiszolgáló esetében, amelyhez csatlakozni szeretnénk. Ezután bárkit hívhatunk, aki az *iax.conf* fájlunkban megadásra került (feltéve, hogy a tárcsázási eljárás beállítása megfelelő). Amennyiben a feleségemet szeretném hívni, egyszerűen a 222 számot tárcsázom, vagy beírom a maria szót (mivel a tárcsázási eljárásban ezt adtam meg). Ha a hangpostámban lévő üzeneteket szeretném ellenőrizni, a 6245 számot kell tárcsáznom.

Összegzés

Csak néhányat mutattam be az Asterisk képességei közül, de úgy gondolom, a cikk alapján az olvasó képet alkothat a programban rejlő lehetőségekről. Az Asterisk egyaránt jól használható a kisebb és a nagy, összetettebb rendszerek

létrehozására. Például az egymástól távol lévő Asterisk kiszolgálókat az IAX protokollon keresztül összekapcsolva virtuális telefonközpont létrehozására is lehetőségünk nyílik. Mivel az Asterisk Linuxon fut, meglévő eszközeinket a csatlakozófelületek létrehozására és az Asterisk irányítására használhatjuk. Például lehetőségünk van a CDR-ek, a beállításfájlok és a hangposta weben keresztül történő elérésére. Tulajdonképpen az Asterisk mellé egy CGI parancsfájlt is kapunk, amely lehetővé teszi hangüzeneteink elérését egy böngészőprogram segítségével. Mindenkinek javaslom, hogy fedezze fel az Asteriskben rejlő további lehetőségeket, és minél jobban hasznosítsa ezt a hatékony eszközt.

Köszönetnyilvánítás

Köszönetet szeretnék mondani a Digiumnak, Reinhard Maxnak és Steve Landersnek a cikk megírásában való közreműködésért.

A cikkhez tartozó Kapcsolódó címek az 59. CD Magazin/ Asterisk könyvtárban található.

Linux Journal 2004. február 118. szám



Brett Schwartz (☞ <http://www.bschwartz.com>)

A Washington állambeli Seattle közelében él feleségével, fiával és kutyájával. Bár otthonosan mozog a különböző operációs rendszerek között, ha teheti, a Linuxot használja.

