

Biztonságos FTP-szolgáltatás üzemeltetése vsftpd használatával

Ha meg szeretnéd őrizni FTP-helyed biztonságát, akkor csak névtelen hozzáféréseket engedélyezz, és használj minél egyszerűbb FTP-démont.

Gondolná bárki is, hogy az immár közel négy éve futó sorozatban még nem volt szó az FTP-szolgáltatás beállításáról? Most pótolom mulasztásomat, ehhez kedvenc FTP-kiszolgálómat, *Chris Evans* kiváló vsftpd-jét (Very Secure, nagyon biztonságos FTP démon) hívom segítségül. Mivel korlátozott nagyságú hellyel gazdálkodhatok, és az FTP legjobb használati módja a névtelen FTP, ez alkalommal erre fogunk összpontosítani. A vsftpd egyre népszerűbb, hogy újabban a Debian, SuSE, Fedora, Red Hat és egyéb Linux-terjesztésekben is megtalálható. Térnyeresésnek oka talán az, hogy egyedi módon egyesíti a biztonságot és a kényelmet. Üzembe helyezni mindössze néhány mozdulat, és nem kényszerülünk választani a biztonság és a kényelem között. *Chris Evans* fő szempontja a vsftpd fejlesztése közben a biztonság megőrzése volt, és eddigi eredményei alapján bizony jó munkát végzett. A vsftpd közel négy éve létezik, de komolyabb biztonsági hiányosságot soha nem találtak benne. Mennyire minimalista a vsftpd? Teljes forráskódfájának mérete tömörítés nélkül is alig haladja meg az 1 MB-ot. A futtatható vsftpd fájl mérete pedig mindössze 80 KB.

A vsftpd letöltése és telepítése

Már említettem, hogy a vsftpd sok Linux-terjesztésben alapelemként szerepel. Ha a terjesztésekből telepítjük, a bináris csomagok használatából fakadó szokásos előnyöket élvezhetjük: kényelem, könnyű foltozás, a rendszerprogramok működésének lehető legkisebb mértékű befolyásolása. Debian, SuSE, Fedora és Red Hat alatt logikusan a vsftpd nevű csomagot kell telepíteni. Különleges függőségei nincsenek. A legtöbb felhasználó tökéletesen elégedett lesz a saját terjesztésében szereplő vsftpd csomaggal. Ha saját terjesztésünkben a vsftpd nem szerepel bináris csomagként, vagy a terjesztésben megtalálhatónál újabb változatot akarunk használni, akkor a <http://www.vsftpd.beasts.org> címről kell letöltenünk a forrást, majd le kell fordítanunk a programot. A fordítás folyamata határozottan régimódi. Ha még nem tettük volna meg, váltsunk át root felhasználóra. Bontsuk ki a *.tar* állományt, majd lépünk át a gyökérfájárába, például így:

```
# tar -xzf vsftpd-1.2.1.tar.gz; cd vsftpd-1.2.1
```

Következő lépésként adjuk ki a *make* parancsot, mindenféle kapcsoló nélkül. Ha futása sikeres, akkor egy futtatható vsftpd fájlt kell találnunk az aktuális könyvtárban. Ellenőrizzük, hogy a nobody (senki) felhasználó létezik-e. Ha nem, hozzuk létre, a program ugyanis ennek jogosultságaival fut. Ha még nincs ilyen, hozzuk létre a */usr/share/empty* könyvtárat. Tulajdonosa a root legyen, de sem csoport által, sem átfogóan ne legyen írható. Ez lesz a vsftpd alapértelmezett chroot börtöne (*jail*). Hozzuk létre a névtelen FTP-felhasználó kezdőkönyvtárát. A SuSE hagyományosan a */srv/ftp* könyvtárat használja, más terjesztésekben a */var/ftp* a megszokott, de azt a könyvtárat választjuk, amelyiket csak akarjuk. Ezt a könyvtárat szintén a root tulajdonába kell adnunk, senki másnak nem szabad írási jogot kapnia rá. Hozzuk létre egy névtelen FTP-felhasználói fiókot, például *ftp* névvel, és ellenőrizzük, hogy kezdőkönyvtára az előző lépésben kiválasztottal egyezik-e. Lehet, hogy a rendszerben már létezik ilyen fiók. A névtelen FTP-felhasználónak ne adjuk írási jogot saját kezdőkönyvtárára, és semelyik fájlra vagy könyvtárra nem lehet a tulajdonosa. Most készen állunk arra, hogy a vsftpd-t, valamint a *vsftpd(8)* és a *vsftpd.conf(5)* man oldalakat a megfelelő helyre másoljuk, tehát adjuk ki a *make install* parancsot. A minta *vsftpd.conf* fájlt kézzel másoljuk a */etc* könyvtárba. Ha a programot önálló démonként szeretnénk futtatni, a */etc/init.d* alatt létre kell hoznunk számára egy indító parancsfájlt. Egyébként az *inetd* vagy az *xinetd* segítségével gondoskodhatunk szükség szerinti indításáról. (Lásd az Önálló démon vagy *inetd/xinetd* című részt.) Ha a vsftpd-t önálló démonként futtatjuk, akkor az indító parancsfájlt kétféleképpen tudjuk engedélyezni: RPM alapú Linux-terjesztés alatt a *chkconfig*, Debian GNU/Linux alatt pedig az *update-rc.d* paranccsal. Ha a telepítést RPM vagy deb csomagból végezzük, akkor mindezekre a műveletekre önműködően sor kerül, kivéve talán az utolsót. Mondtam már, hogy bináris csomagokat használni jóval kényelmesebb? Egyes terjesztéseknél az újonnan telepített csomagokat kézzel kell engedélyezni. Például a saját SuSE 9.0 rendszeremen a SuSE vsftpd RPM hiába telepítette

a `/etc/init.d/vsftpd` fájlt, engedélyezéséhez ki kellett adnom a `chkconfig --add vsftpd` és a `chkconfig --level 35 vsftpd` parancsot.

A vsftpd leírása

Most nézzük meg, honnan szerezhetünk információt a program működéséről. Először is, a vsftpd-hez tartozik egy **EXAMPLE/** könyvtár, amely mintabeállításokat tartalmaz különféle alkalmazási környezetekhez; ide értve az önállóan és xinetd alatt történő futtatást, valamint a névtelen és a helyi felhasználók kiszolgálását egyaránt. Ha a vsftpd-t forráskódból telepítettük, az **EXAMPLE** könyvtárat a forrást tartalmazó könyvtárban találjuk. Ha a telepítéshez bináris csomagot használtunk, akkor nagy valószínűséggel felkerült egy másolata a gépre, valahova a `/user/share/doc` könyvtár alá. SuSE rendszereken a `/usr/share/doc/packages/vsftpd/EXAMPLE` elérési úton találjuk. Az előző részben már említettem, hogy a vsftpd-hez tartoznak man oldalak, a `vsftpd(8)` és a `vsftpd.conf(5)`. Az utolsó megemlíthető forrás, az alapértelmezett (mintaként használható) `vsftpd.conf` maga is rengeteg megjegyzést tartalmaz. Ugyan nem tartalmazza az összes lehetséges beállítást, de a leggyakrabban használtakat igen. Jómagam számos vsftpd-példányt bírtam már működtetni úgy, hogy a minta `vsftpd.conf` fájlban csak elenyésző módosításokat kellett eszközölnöm.

Önálló démon vagy inetd/xinetd

Mielőtt magának a vsftpd-nek a beállításait megadnánk, el kell döntenünk, hogy önálló démonként vagy szuperkiszolgálóként, tehát inetd vagy xinetd segítségével szeretnénk futtatni. A vsftpd korábbi változataiban a fejlesztő annak naplózási és hozzáférés-vezérlési szolgáltatásai miatt az xinetd-vel történő használatot javasolta. A vsftpd 1.2-es és újabb változatai már maguk is képesek biztosítani ezeket a szolgáltatásokat. Éppen ezért **Evans** most már a program önálló démonként való futtatását javasolja. Az inetd vagy az xinetd használatából természetesen némi teljesítményvesztés is származik. Ez a veszteség semmivel sem ellentételezhető, ha dedikált FTP-kiszolgálót akarunk üzemeltetni, vagy úgy véljük, a rendszer terhelésének jelentős része fog az FTP-szolgáltatás futásából származni. Veszem magamnak a bátorságot, és a továbbiakban feltételezem, hogy önálló démon futtatására rendezkedünk be. A megfelelő leírások részletesen taglalják az inetd-vel és xinetd-vel való használatot, illetve a vsftpd **EXAMPLE** könyvtárban található példabeállítások alapján is sok mindenre rá tudunk jönni. Érdekes módon a SuSE 9 a vsftpd-t alapesetben xinetd, a Debian 3.0 pedig inetd alól futtatja. Utóbbi választás ésszerű, hiszen a Debian 3.0 a vsftpd egy régebbi, 1.0.0-s változatát tartalmazza, ám a SuSE 9.0-ban az 1.2-es változatot szerepel. A Fedora és a Red Hat terjesztésekhez készült RPM-ek önálló démonként telepítik a vsftpd-t. Tulajdonképpen mindegy is, a vsftpd inet/xinetd alóli indításról néhány lépéssel átváltható önálló indulásra. Először is, mint az **A vsftpd letöltése és telepítése** című részben már említettem, ellenőriznünk kell, hogy a `/etc/init.d` alatt van-e engedélyezett indító parancsfájl a vsftpd-hez. A Fedora Core 1 és a SuSE 9.0 csomagok tartalmazzák és telepítenek is ilyet, SuSE alatt a fájl megvan ugyan, ám az

xinetd alól történő futtatás miatt alapesetben le van tiltva. Ha a Debian 3.0 vsftpd csomagját használjuk, vagy forrásból végezzük a telepítést, akkor magunknak kell elkészítenünk az indító parancsfájlt. Ugyancsak létre kell hoznunk a megfelelő hivatkozásokat azoknak a futási szinteknek a könyvtárában – mint például `rc3.d` és `rc5.d` –, amelyekben futtatni szeretnénk a demont. Végül ki kell adnunk a `chkconfig` vagy az `update-rc.d` parancsot. Az átállás második lépése a vsftpd xinetd fájljának letiltása a `disable=yes` érték megadásával a `/etc/xinetd.d/vsftpd` fájlban, vagy a vsftpd sorának megjegyzésbe tételével a `/etc/inetd.conf` állományban. Azt is megtehetjük, hogy teljes egészében letiltjuk az inetd-t vagy az xinetd-t, persze csak akkor, ha a vsftpd volt az egyetlen fontos alóla indított dolog. Tudom, felelőtlenség rávenni valakit egy alkalmazás indító parancsfájljának engedélyezésére, míg annak biztonsági szolgáltatásai nincsenek pontosan beállítva. Úgy gondolom azonban, hogy az engedélyezésből önmagában semmi baj nem származhat, ha az illető pontosan követi a javaslataimat, és egyelőre letiltja a szolgáltatást. A harmadik lépés annak ellenőrzése, hogy a `/etc/vsftpd.conf` fájlban a `listen` kapcsoló **YES** értéket kapott-e. Ezután továbbléphetünk a tényleges beállításokra.

A vsftpd beállítása névtelen FTP-szolgáltatás biztosítására

Nagy valószínűséggel mást nem is kell tennünk ahhoz, hogy a vsftpd-vel biztonságos névtelen FTP-szolgáltatást tudjunk nyújtani. Alapértelmezett beállításai kizárólag ilyen hozzáférést engednek. Sőt mi több, alapesetben semmilyen írási parancs végrehajtását nem engedi, és a vsftpd újabb változatai lehetőség szerint chroot műveletet hajtanak végre a `/usr/share/empty` könyvtárban. Ez az egyik dolog, amiért a vsftpd-t szeretem. Igazából az általa nyújtott biztonságot elrontani több munkával jár, mint megőrizni vagy tovább erősíteni. Ha feltételezzük, hogy mindez saját terjesztésünkben sincs másképp, akkor csak annyit kell tennünk, hogy a névtelen FTP-felhasználó kezdőkönyvtárba átmásoljuk a másoknak letöltésre szánt anyagokat. Debian 3.0, SuSE 9.0 és Fedora Core 1 alatt a névtelen FTP-felhasználó alapesetben az ftp fiókot használja, kezdőkönyvtára Debian és SuSE alatt `/srv/ftp`, Fedora alatt pedig `/var/ftp`. Ha a telepítést forrásból végeztük, a névtelen FTP könyvtára az lesz, amit a névtelen FTP-felhasználó fiókjához kezdőkönyvtárként hozzárendeltünk. Az FTP könyvtárainak fájlokkal feltöltésekor ügyeljünk a tulajdonjogok és az engedélyek pontos beállítására. Előfordulhat, hogy az alapértékek nem megfelelőek, ám egy gyors `ls -al` mindent elárul. Ugyan a legtöbb felhasználó számára az alapértelmezett beállítások tökéletesen megfelelnek, tekintsük át a `vsftpd.conf`-ban szereplő, a névtelen FTP-vel leginkább kapcsolatos beállításokat. Alapesetben ez a fájl a `/etc` könyvtárban található, bár Red Hat és Fedora rendszereken `/etc/vsftpd/` könyvtárban találjuk. Az **1. kódrészlet** egy minta `vsftpd.conf` fájl szemléltet. A gyakorlatban a `vsftpd.conf` fájl senki nem használja az **1. kódrészletben** szereplő formában, ott ugyanis minden beállítás alapértelmezett értékkel szerepel. A kódrészletet inkább hivatkozási alapnak szántam. Lássuk tehát a beállításokat.

- `listen`: Utasítja a vsftpd-t, hogy démonként fusson, és ne az inetd vagy az xinetd által igény szerint, kapcsolatonként indított folyamatként. Alapértéke **NO**.

- `listen_address`: Megadja azt a helyi IP-címet, amelyen a vsftpd-nek figyelnie kell a bejövő kapcsolatokat. Alapértéke "" (null), ami az összes helyi IP-címet jelenti. Ha több képzetes FTP-kiszolgálót szeretnénk futtatni, akkor értékét mindegyik képzetes kiszolgáló beállító fájljában meg kell adnunk. (Lásd a Képzetes kiszolgálók című részt.)
- `anonymous_enable`: Alapértéke YES, meghatározza, hogy a vsftpd fogadja-e a névtelen bejelentkezéseket. Ha értéke YES vagy nincs megadva, akkor a vsftpd valódi jelszó kérése nélkül fogadja az anonymous (névtelen) és a ftp felhasználók kapcsolatait (a két felhasználó egyenértékű).
- `ftp_username`: A névtelen – vagyis az anonymous és az ftp névvel történő – FTP-bejelentkezésekhez használt felhasználói fiók neve. A fióknak léteznie kell a */etc/passwd* fájlban, és érvényes, nem az adott fiók által tulajdonolt kezdőkönyvtárral kell rendelkeznie. Alapértéke az ftp.
- `anon_root`: Az a könyvtár, amelybe a vsftpd chroot műveletet hajt végre a névtelen bejelentkezéseknél. Az alapérték a névtelen FTP-felhasználói fiók kezdőkönyvtára (lásd az `ftp_username` beállítást), de az `anon_root` segítségével ettől eltérő FTP kezdőkönyvtár is megadható. Bármelyik megoldást is választjuk, a könyvtár tulajdonosa ne legyen a névtelen FTP-felhasználó.
- `write_enable`: Hacsak értéke nem YES, egyik felhasználó sem tölthet fel semmilyen fájlt, függetlenül a *vsftpd.conf* egyéb beállításaitól. Alapértéke NO.
- `anon_upload_enable`: Ha ennek értéke, valamint a `write_enable` beállítás értéke egyaránt YES, akkor a névtelen felhasználók engedélyt kapnak fájlok feltöltésére azokba a könyvtárakba, amelyekre a névtelen felhasználó írási engedéllyel rendelkezik.
- `anon_mkdir_write_enable`: Ha ennek értéke, valamint a `write_enable` beállítás értéke egyaránt YES, akkor a névtelen felhasználók engedélyt kapnak könyvtárak létrehozására azokban a könyvtárakban, amelyekre a névtelen felhasználói fiók írási engedéllyel rendelkezik.
- `anon_other_write_enable`: Ha ennek értéke, valamint a `write_enable` beállítás értéke egyaránt YES, akkor a névtelen felhasználók engedélyt kapnak könyvtárak átnevezésére és törlésére azokban a könyvtárakban, amelyekre a névtelen felhasználó írási engedéllyel rendelkezik.
- `anon_world_readable_only`: Ha értéke YES, akkor a névtelen felhasználók nem tudják letölteni az általános jelleggel nem olvasható fájlokat. Akkor használható jól, ha a névtelen felhasználók engedélyt kapnak fájlok feltöltésére, de nem akarjuk, hogy más névtelen felhasználók ezeket a fájlokat letöltsék.
- `anon_max_rate`: Megadja, hogy a névtelen felhasználók legfeljebb hány bájt/másodperc sávszélességet használhatnak fel. Alapértelmezett értéke 0, ami azt jelenti, hogy semmilyen korlátozás nincs.
- `idle_session_timeout`: A felhasználók legfeljebb ennyi másodpercig adhatnak ki különféle FTP-parancsokat, mielőtt a kapcsolatot a kiszolgáló lezárná. Alapértéke 300, de ha aggódunk a szolgáltatásmegtagadási támadások miatt, csökkentjük.
- `ascii_download_enable`: Ha értéke YES, akkor a felhasználók ASCII módú letöltéseket is indíthatnak, nemcsak binárisokat. Alapértéke NO, mivel az ASCII mód

1. kódrészlet A vsftpd.conf fájlban szereplő beállítások névtelen FTP üzemeltetéséhez

```
listen=YES
# listen_address=
anonymous_enable=YES
ftp_username=ftp
# anon_root=[$az ftp_felhasználó kezdőkönyvtára]
write_enable=NO
anon_upload_enable=NO
anon_mkdir_write_enable=NO
anon_other_write_enable=NO
anon_world_readable_only=YES
anon_max_rate=0
idle_session_timeout=300
ascii_download_enable=NO
ascii_upload_enable=NO
connect_from_port_20=NO
port_enable=YES
hide_ids=NO
log_ftp_protocol=NO
syslog_enable=NO
max_per_ip=0
# cmds_allowed=
local_root=/usr/share/empty
nopriv_user=nobody
ftpd_banner=(vsFTPd 1.2.0)
```

használatára szinte soha nincs szükség, hatékonysága pedig annyira rossz, hogy kiváló eszköz szolgáltatásmegtagadási támadások indítására.

- `ascii_upload_enable`: Az ASCII módú feltöltés lehetősége bizonyos esetekben jól jöhet, például ha parancsfájlokat akarunk továbbítani. Alapértéke ettől függetlenül NO.
- `connect_from_port_20`: Az aktív módú FTP-kapcsolatoknál, ha egy felhasználó letölt valamit, akár csak egy könyvtár fájllistáját, a kiszolgáló új kapcsolatot nyit az ügyfél felé, és ez általában a 20-as számú TCP-kapuról indul ki. Alapesetben azonban a vsftpd az ilyen kapcsolatokat magasabb számú kapuról indítja, így nem muszáj rootként futnia. Ha az alapértéket meg akarjuk változtatni, mert például a felhasználók ilyen jellegű kapcsolatindításokra fel nem készített proxyk vagy tűzfalak mögül érkeznek, akkor adjunk neki YES értéket.
- `port_enable`: Ha NO értéket adunk neki, a PORT parancsokat letilthatjuk, amivel gyakorlatilag megtiltjuk az aktív módú FTP-t. Alapértéke YES.
- `hide_ids`: Ha YES értéket adunk neki, akkor a könyvtártartalmak listázásakor a felhasználók mindenhol ftp tulajdonost és ftp csoportot látnak. Szerintem nyilvános FTP-kiszolgálókon van értelme egyfajta rejtőzködésre használni, de alapértéke NO.
- `log_ftp_protocol`: Ha YES értéket adunk neki, engedélyezzük az FTP protokollparancsok részletes naplózását. Ezeket a felhasználói FTP-parancsok indítják ugyan, de azoktól eltérők. Hibakeresési szempontból értéktelen.

2. kódrészlet Képzetes FTP-kiszolgáló beállító fájlja (/etc/vsftpd.knusper)

```
listen=YES
listen_on=1.2.3.4
connect_from_port_20=YES
anonymous_enable=YES
anon_root=/srv/ftp/knusper
ftpd_banner=Üdvözlök a knusper.wiremonkeys.org
↳ FTP-helyén. Viselkedj jól!
```

3. kódrészlet Képzetes FTP-kiszolgáló beállító fájlja (/etc/vsftpd.rover)

```
listen=YES
listen_on=1.2.3.5
connect_from_port_20=YES
anonymous_enable=NO
ftpd_banner=Zártkörű FTP a rover.wiremonkeys.org
↳ címen. Idegeneket nem fogadunk.
# VIGYÁZAT: Ne használd, amíg nem tudod
↳ pontosan, hogy mit is csinálsz!
local_enable=YES
```

- `syslog_enable`: Normál esetben a `vsftpd` a naplőüzeneteket a `/var/log/vsftpd.log` fájlba írja. Ha a beállítás értéke `YES` (alapértéke `NO`), akkor az üzeneteket a rendszer `syslog` szolgáltatásának küldi.
- `max_per_ip`: Megadja, hogy egy-egy forrás IP-címről egyszerre legfeljebb hány kapcsolatot lehet indítani. Valamiféle korlátozást bevezetni nem rossz ötlet, bár az alapérték nulla, ami korlátlan kapcsolatszámot jelent. Ha viszont korlátozunk, a NAT/SPAT tűzfalak mögül érkező felhasználókkal kiszűrhatunk, hiszen ők úgy látszanak, mintha egy-egy forrás IP-címről több kapcsolat is eredne.
- `cmds_allowed`: A megengedett FTP-parancsok vesszővel ellátott listája. Alapértéke "" (null), vagyis nincs korlátozás. Csak FTP-protokoll szintű parancsokat lehet megadni, az FTP-ügyfélpogramok által gyakran kezelt parancsokat nem. Ha például az ügyfelek tevékenységét a fájlok listájának lekérdezésére, a munkakönyvtár megváltoztatására és a fájlok letöltésére szeretnénk korlátozni, akkor a következő beállítást kell használnunk: `cmds_allowed=USER,LIST,NLST,CWD,RETR,PORT,QUIT`. A <http://www.nsftools.com/tips/RawFTP.htm> oldalon kiváló ismertetőt találunk ezekről a parancsokról.
- `local_root`: Üres, a root által tulajdonolt könyvtárat ad meg, a `vsftpd` ide végez `chroot` műveletet, ha éppen a fájlrendszer egyik részéhez sem igényel hozzáférést. Alapértéke `/usr/share/empty`.
- `nopriv_user`: Kiemelt jogok nélküli felhasználó, a `vsftpd` ennek jogaival fut, ha lehetséges. Természetesen vannak olyan műveletek, amelyekhez root jogokkal kell futnia, ilyen például a 21-es TCP kapuhoz való kötődés. A `vsftpd` a lehető leghamarabb lefokozza önma-

gát, így próbál hozzájárulni a puffer-túlsordulásos és az egyéb, root-ként eredményesebben kivitelezhető támadások esélyének és sikerességének csökkentéséhez.

- `ftpd_banner`: A csatlakozni próbáló FTP-ügyfeleknek megjelenített üzenet. Az alapértelmezett üzenet bele van drótozva a `vsftpd`-be, az 1.2.0-s változatban ez egyszerűen csak (`vsftpd` 1.2.0). Ha saját üzenetet szeretnénk használni, akkor a `banner_file` beállításban adjuk meg az azt tartalmazó szöveges fájl nevét.

A `vsftpd.conf(5)` man oldal az itt szereplők mellett számos más beállítást is ismertet. Hihető vagy sem, itt csak a felszint érintettük.

Képzetes kiszolgálók

Egy több IP-címmel is rendelkező gépen több képzetes FTP-kiszolgálót is futtathatunk. Mindössze annyit kell tennünk, hogy több példányban futtatjuk a `vsftpd` demont, mindegyik példányt saját `vsftpd.conf` fájljal ellátva, amelyben megadjuk, hogy az adott példánynak melyik IP-címen kell fogadnia a kapcsolatokat, illetve melyik könyvtárat kell névtelen kezdőkönyvtárként használnia. Tegyük fel például, hogy a gépnek két IP-címe van, az 1.2.3.4 és az 1.2.3.5, ezek rendre a `knusper` és a `rover` DNS-névhez tartoznak. Ebben az esetben két `vsftpd` beállító fájlt is használhatok, például `/etc/vsftpd.knusper` és `/etc/vsftpd.rover` névvel. A 2. és a 3. kódrészlet ezeket a fájlokat tartalmazza.

Talán furcsának tűnhet a `local_enable` beállítás használata a 3. kódrészletben. Veszélyes dolog `YES` értéket adni neki, mivel ilyenkor az FTP-bejelentkezéshez használt azonosítókat nyílt szövegben kerülnek továbbításra. Elsősorban azért szerepeltetem itt, hogy bemutassam, minden képzetes kiszolgáló saját beállító fájlt használ, és akár merőben eltérő módon is működhet. Megtehetjük például, hogy indítunk egy névtelen felhasználók által is írható képzetes kiszolgálót a nyilvános feltöltések számára, egy másikat pedig egy szigorúan csak olvasható FTP-hely fenntartásához. Ha létrehoztuk a képzetes FTP-kiszolgálók saját beállító fájljait, az indító parancsfájlt is módosítanunk kell. Az én példakiszolgálómon – innen származik a 2. és a 3. kódrészlet is – az indító parancsfájlból az alábbi kettőhöz hasonló sorok kerültek:

```
vsftpd /etc/vsftpd.knusper
vsftpd /etc/vsftpd.rover
```

Ha Red Hat vagy Fedora rendszert futtatunk, akkor ezzel a feladattal nem kell foglalkoznunk. Az ezeknek a terjesztéseknek a `vsftpd` RPM-csomagjában szereplő `/etc/init.d/vsftpd` parancsfájl önműködően megkeresi a `/etc/vsftpd` könyvtárban szereplő beállító fájlokat, feltéve, hogy `.conf` kiterjesztéssel látjuk el őket.

Linux Journal 2004. július, 123. szám



Mick Bauer (mick@visi.com)

Biztonsági szakember, a Linux Journal biztonsági témákkal foglalkozó szerkesztője, biztonsági tanácsadó a Minnesota állambeli Minneapolisban található Upstream Solutions LLC Inc.-nél.