

## Slash

Az egyik legrégebbi és legtöbb szolgáltatást nyújtó webközösség-rendszer egyben az egyik legnehezebben telepíthető is. Reuven ezúttal `mod_perl`-lel és egyéb előfeltételekkel kapcsolatos tanácsokkal segít Slash webmesterré válni.

**A** webnapló, vagy más néven blog népszerűsége néhány éve megállíthatatlanul nő, és semmi jele annak, hogy kicsit is alábbhagyna. Noha sok webnapló-író továbbra is független szolgáltatások – például a LiveJournal vagy a Blogger – segítségével jegyzi le gondolatait, egyre könnyebbé válik saját kiszolgálón webnaplót működtetni. Az elmúlt néhány hónapban számos programcsomagot megnéztünk, amelyek biztosítják ezt a lehetőséget. Ezek közé tartozik a COREBlog, amely egy Zope termék, és a Blossom, egy Perl-ben íródott CGI programcsoport. A múlt hónapban, amikor a XOOBS-ot vizsgáltuk, egy kicsit másmilyen weblog-rendszert láttunk. A XOOBS, akárcsak az unokatestvérei, a PHPNuke és a PostNuke, a csoportok és tagok felügyelete mellett lehetővé teszi a felhasználóknak tartalom-kezelés és online közösség létrehozását is. A XOOBS segítségével egyszerűen lehet a rendszer minden felhasználójának saját, egyéni webnaplót biztosítani. Bármennyire is népszerű a XOOBS, a közösségi programok ösátyja vitathatatlanul a Slash, amely a közkedvelt

➔ <http://www.slashdot.org> és a ➔ <http://www.use.perl.org> webhelyek motorja. A Slash elsősorban hírek és hozzászólások közzétételére szolgál, de nagyon hatékony webnapló szolgáltatása is van, amely a rendszer minden felhasználója számára elérhető. Néhányan erre azt mondanák, hogy a Slashdot maga egy közösség által szerkesztett webnapló, és ez a nézet számomra is egész meggyőző. A legjobb az, hogy ez a webnapló szolgáltatás együttműködik az oldal többi elemével, így ha egy másik felhasználótól különösen éles elméjű vagy ízléstelen hozzászólást olvassunk, rögtön megnézhetjük a naplóját, hogy többet tudhassunk meg róla. Ebben a hónapban egy egyszerű Slash webhely telepítését és beállítását nézzük meg, amely beépített támogatást biztosít a felhasználóknak és a webnaplóknak (vagy ahogy a Slash-zsargon mondja: „journal”-nak). A következő hónapban közelebről tanulmányozzuk a Slash webnapló lehetőségét, valamint azt, hogyan lehet beállítani és változtatni, hogy megfeleljen egyedi igényeinknek.

### Háttér

A Slash terjesztésének és megvitatásának a fő helye a Slashcode. A cikk keletkezésékor, 2004. áprilisának elején, a legutolsó hozzászólás címe: „Barátságos Slash telepítés”,

amelyben a szerző azt kérdezi, létezik-e a Slash telepítésének világos, egyszerű módja. Sajnos, úgy tűnik, a válasz nem – és ennek számos oka van:

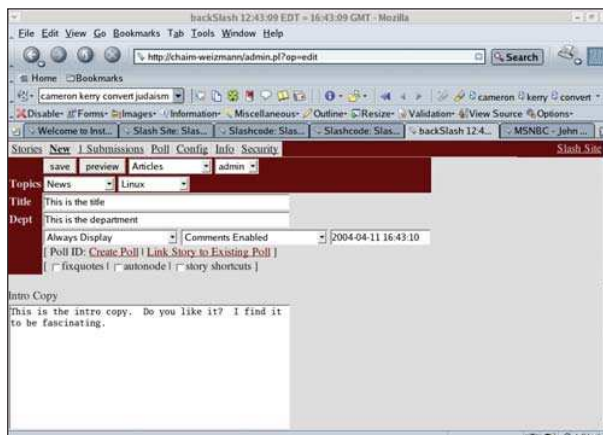
- A Slash leírása és használati utasítása az emberek többségét a hivatalosan kiadott `.tar.gz` csomagokhoz vezeti, amelyek több, mint két éve elavultak.
- A legújabb CVS (*Concurrent Versions System*) változat szabadon hozzáférhető és naprakész, de nem kizárt, hogy megbízhatatlan, hacsak nem tudjuk, melyik címkézett változatot kell letölteni.
- A telepítés nagyrészt manuális művelet, számos kritikus ponton hibalehetőségekkel.
- A Slash leírása nem túl jó minőségű.
- Ha sikerül is a kód egy friss, működő példányát telepíteni, a teljes Slash telepítéshez szükség van a `mod_perl`, a Template Toolkit, a MySQL telepítésére, egy csomó más Perl modulra és független eszközre, a saját, kissé eltérő és nem szabványos beállításaiikkal.

Ha járatosak vagyunk a `mod_perl`-ben, a MySQL-ben, és az Apache-ban, akkor a Slash telepítése egy kissé kellemetlen, de elvégezhető folyamat. Ha kevésbé vagyunk gyakorlottak, az eredmény valószínűleg megéri a fáradságot, számítsunk rá, hogy menet közben nagyon sokat megtanulunk ezekről az alkalmazásokról. Lehet azonban, hogy egyszer csak azon kapjuk magunkat, hogy az IRC csatornán, vagy a webhelyen keresünk tanácsokat és ötleteket.

A Slash telepítésének első lépése az Apache, a `mod_perl` és a MySQL telepítése. A MySQL bármelyik újabb változata megfelelő, az igazi gond az Apache-csal és a MySQL-lel lehet, amelyeket elsöre talán bonyolult telepíteni. Szerencsére, az évek során a telepítés folyamatának ez a része alapvetően nem változott, ami azt jelenti, hogy követhetjük az InstallSlash oldalon található MySQL, Apache és `mod_perl` telepítési utasításokat. (Lásd az online források részt). Ha sem a Perl, sem az `expat` nincs telepítve a rendszeren, akkor ehhez is segítséget nyújtanak az InstallSlash utasításai. Ne feledjük, hogy RedHat és Fedora rendszereken nem csak az `expat` RPM-et, hanem az `expat-devel` RPM-et is telepíteni kell. Meg kell adnunk egy MySQL adatbázist is, amelyben a hely adatai tárolódnak, alaphelyzetben ez a `slashdb` nevet kapja. Az

Apache és a *mod\_perl* fordításakor az `EVERYTHING=1` megadásának szükségessége jelentős problémaforrás. Ez működésbe hozza a *mod\_perl* összes horgát (*hook*), ezzel lehetővé teszi a *mod\_perl* számára, hogy felülbírálja az Apache minden alapértelmezésbeli működését, beleértve a hitelesítést, az engedélyezést, az URL átirást és a naplózást. Az `EVERYTHING=1` megadása nélkül a *mod\_perl* csak tartalmat hozhat létre. Ha a rendszerünket már telepített *mod\_perl*-lel szereztük, valószínűleg az `EVERYTHING=1` megadása nélkül fordították, ami azt jelenti, hogy magunknak kell újrafordítanunk.

A Slash használati utasítás a `PERL_MARK_WHERE=1` beállítását is javasolja a rendszergazdának fordításakor, habár a *mod\_perl* kódja és leírása szerint ez az utasítás törli a meghatározatlan értékekre vonatkozó figyelmeztetések nagy részét a hibnaplóból. Én figyelmen kívül hagytam ezt a tanácsot, és a Slash-t a létező *mod\_perl* telepítésemen használtam, de nem tapasztaltam semmilyen káros hatást.



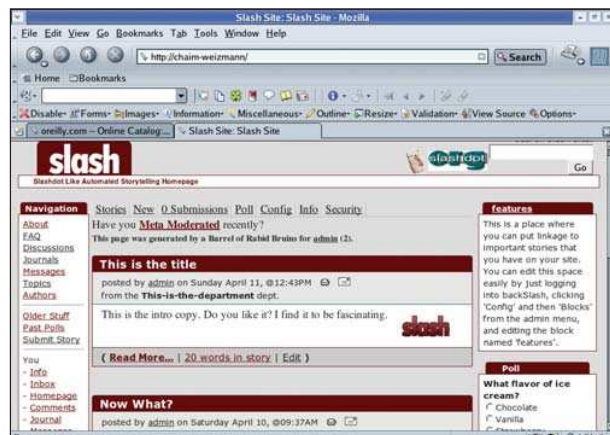
1. ábra Ide írjuk a szöveget...

Ha úgy döntünk, hogy biztonsági okokból, vagy egyéni ízlésünk miatt megváltoztatjuk az értékeket, semmiképp ne felejtjük el, milyen neveket használtunk.

A *Bundle::Slash* letöltésének legnehezebb része a Template Toolkit telepítése. A Template Toolkit egy népszerű és hatékony sablon-rendszer a *mod\_perl* alatt, ami arra szolgál, hogy egy Slash webhely különböző oldalait következetes és praktikus módon jelenítse meg.

### Címkézett változatok letöltése a CVS-ről

Ennél a pontnál magát a Slash kódot kell letöltenünk és telepítenünk. Az *InstallSlash* használati utasítás végigkísér minket a SourceForge-on található, legfrissebb csomagolt változat (2.2.6) letöltésének és telepítésének folyamatán. Viszont, mint fentebb említettem, 2.2.6 kibocsátása óta eltelt két évben rengeteg fejlesztés történt, és ezek a javítások csak a CVS változatban elérhetőek.



2. ábra ...és így jelenik meg.

Végül, telepítsük a rendszerre a CPAN-ről (olyan kiadók világméretű hálózata, amelyek szabadon felhasználható Perl modulokat és leírásokat tartalmaznak) a *Bundle::Slash* csomagot, az önműködő CPAN eszköz segítségével. A *Bundle::Slash* tulajdonképpen semmilyen kódot nem tartalmaz, csak felsorolja a Slash futtatásához szükséges modulokat, így nekünk nem kell megjegyezni (és begépelni) az összes telepítendő Slash-hez kapcsolódó modult. Rendszergazdaként bejelentkezve a következő sor begépelésével telepíthetjük a modulokat:

```
# perl -MCPAN -e 'install Bundle::Slash'
```

Ha még ezelőtt sosem használtuk a CPAN-t, meg kell adnunk néhány CPAN-nel kapcsolatos paramétert, köztük a legközelebbi CPAN archívumot, amelyről modulokat tölthetünk le. Ez lehet, hogy elsőre egy kicsit bonyolult lesz, bár elfogadhatjuk az alapértékeket is, valószínűleg minden következmény nélkül.

A CPAN modulok telepítésének egyik nehézsége a *DBIx::Password*, amely a telepítésekor a webhelyre vonatkozó információkat kér. Az *InstallSlash* utasításai megmondják, mit kell válaszként beírni a parancssorba.

A következő paranccsal letölthetjük a legfrissebb változatot a CVS-ről, és elmenthetjük egy slash nevű könyvtárban.

```
cvsv -z3 -d:pserver:anonymous@cvs.sourceforge.net:
  /cvsroot/slashcode co -r T_2_3_0_148 slash
```

Ugyanakkor a legfrissebb CVS változat SourceForge-ről való letöltésének megvannak a maga buktatói, mert ezzel vállaljuk a kockázatot, hogy olyan kódot telepítünk, amit mostanában töltöttek fel, de nem ellenőriztek. A legjobb megoldás a címkézett változat használata. Ahogy a tapasztalt CVS fejlesztők tudják, minden CVS állománynak van egy változatszám (revision number), mint például 1.5, vagy 2.8 vagy 3.1.1.2. Ezek a változatszámok csak az egyes állományok változataira vonatkoznak, és semmi közük a teljes projekthez. Ezért annak ellenére, hogy program projekt 2.0-s változatként kerül a nyilvánosság elé, az egyes állományok majdnem biztos, hogy nem 2.0-s változatszámúak. Hogy egy közös nevet (vagy számot) lehessen rendelni a projekt összes állományának a pillanatnyi állapotához, címkéket, vagy más néven jelképes változatokat kell használni. Amennyiben van írási jogunk, megcímkézhetjük a jelenlegi *release-2-0* könyvtárat, a következőképpen:

cvs tag release-2-0

A címkék nem tartalmazhatnak bizonyos karaktereket, például vesszőt vagy pontot sem. Ezért kell kötőjelet használni pont helyett annak jelzésére, hogy a programunk 2.0-s kiadású. Ez felveti azt a kérdést, hogy melyik címkét tudjuk vagy szükséges letölteni. A Slash esetében a kód lassan halad a 2.3.0-s változat felé, és a fejlesztők egységesítették a címkéket, a következőképpen: T\_2\_3\_0\_XXX. A T azt jelöli, hogy tesztelt, és az XXX minden egyes címkével növekszik. A cikk írásáig a legfrissebb címke a T\_2\_3\_0\_148, melyet a következőképpen tölthetünk le:

```
cvs -z3 -d:pserver:anonymous@cvs.sourceforge.net:
➔ /cvsroot/slashcode co -r T_2_3_0_148 slash
```

Ez a parancs létrehoz egy *slash* nevű könyvtárat, és minden Slash-hez kapcsolódó kódot, könyvtárat és leírást ebbe tesz. Az *INSTALL* állomány, amely azoknak az utasításoknak a naprakész változata, amelyeket arra terveztek, hogy az éppen letöltött CVS változattal működjenek, különösen hasznos. Követtem a fájlban leírt utasításokat, begépeltem a `make install` parancsot az összes Slash alkotóelem telepítéséhez. A telepítési folyamat során kapott üzenet megmutatja, milyen parancsot tegyünk az Apache beállító fájl végére. Ez biztosítja, hogy az Apache tartalmazza a megfelelő Perl modulokat az induláskor, ezzel lényegesen csökkentve a *mod\_perl* memóriaigényét és futási idejét.

Az egyik jó dolog a Slash-ben, hogy könnyedén tud több webhelyet kezelni ugyanazzal a kóddal. Azaz, ha úgy döntünk, hogy külön online közösséget szeretnénk a Perl, a Python, a Tcl és a Ruby számára, akkor a Slash ezt képes kezelni. Minden közösségnek saját gépnévre van szüksége, de ezek lehetnek egymástól teljesen különbözőek. Más szóval, egy Slash weboldal telepítése a Slash program telepítésétől különálló folyamat. Amikor a programot már telepítettük, alaphelyzetben a `/usr/local/slash` alá, a következő parancs futtatásával hozhatjuk létre az új webhelyet:

```
/usr/local/slash/bin/install-slashesite -u USER
```

Itt a USER ugyanaz a virtuális felhasználó, amelyet még a *DBIx::Password* telepítésekor hoztunk létre, alaphelyzetben a *virtslash* nevet kapja. Az `install-slashesite` még néhány kérdést feltesz, többek közt a rendszergazda felhasználónevet és jelszavát, ezután az Apache `apachectl` programmal való leállítására és elindítására utasít. Ez a program általában a `/usr/local/apache/sbin` alá van telepítve. Az `apachectl` program `restart` utasítását nem szabad használni, különösen, amikor *mod\_perl*-lel dolgozunk. Le kell állítani az Apache-ot, várni néhány másodpercet, hogy a folyamatok teljesen ki tudjanak lépni, és azután újraindítani.

Amikor megpróbáltam futtatni az `install-slashesite`-ot, észrevettem, hogy legalább egy CPAN modul hiányzik (*LWP::Parallel::UserAgent*). Nem tartott sokáig telepíteni, de csalódott voltam, amikor kiderült, hogy sem a *Bundle::Slash*, sem a `make install` nem észlelte vagy javította ezt.

## Egy egyszerű Slash weboldal

A Slash webhely most már használatra kész. Az én gépem, amelynek a saját otthoni hálózatomon *chaim-weizmann* a neve, meg tudtam nézni a webhelyemet úgy, hogy a `http://chaim-weizmann` címre ugrottam. Egy újonnan telepített Slash weboldal természetesen nem valami izgalmas. Ha azzal a rendszergazda felhasználónevet és jelszóval lépünk be, amit az `install-slashesite`-nek adtunk meg, számos lehetőséget és menüt kapunk, amely elég nagy irányítást biztosít a rendszer fölött. Sok menüt nehéz megtalálni, és igénybe vesz némi időt, mire a Slash rendszergazda rájön, hogy hol hajtódnak végre a különböző változtatások, a webhelyen, vagy a lemezen lévő sablonokban és programokban. A *slashguide.pod*, amely a CVS változattal kapott docs könyvtárban található, jó bevezető a Slash webhelyek karbantartásába.

Ha már valamennyire ismerjük az eredeti Slashdot webhelyet, rögtön elkezdhetünk történeteket feltenni, és gyűjteni a hozzászólásokat. A webhely rendszergazdájaként bejelentkezve kattintsunk a főoldal tetején a *New* (Új) hivatkozásra. Adjunk meg egy témát, egy címet, egy kategóriát, egy bemutatató változatot (a főoldalon látható majd) és egy hosszabb változatot (az önálló történet-oldalon látható majd). Még közvélemény-kutatást is csinálhatunk, vagy összekapcsolhatunk egy történetet egy meglévő közvélemény-kutatással. Mikor jóváhagytuk a történetet, az egész világ számára láthatóvá válik. Az oldal felhasználóinak tetszés szerint lehetővé tehetjük a hozzászólást, sőt, még moderátorok is lehetnek. A Slash történetek közösségi moderálása és meta-moderálása tényleg a legizgalmasabb ötlet, amit a Slash letett az asztalra.

## Következtetés

A Slash-t nehezebb telepíteni, mint bármely programcsomagot, amelyről eddig szó volt. Ez részben azon múlik, hogy a szerzők milyen megoldást választottak, mivel a *mod\_perl*-t lényegesen nehezebb telepíteni, mint a XOOPS és a Zope által használt PHP-t. Ugyanakkor, a Slash több közösségi és webnapló lehetőséget kínál, mint más programcsomagok, közismerten nagy forgalmat képes kezelni, és igen magas szinten lehet karban tartani.

Ha nem félünk, hogy összepiszkoljuk a kezünket, és szeretnénk elérni azt a hatékonyságot, amit a Slashdot webhely biztosít, a Slash jó választás lehet a közösségek vagy egyének által vezetett webnaplók számára. Következő hónapban olyan személyes naplókat nézünk meg, amelyeket a Slash segítségével lehet létrehozni egy olyan társrendszerrel együtt, amelynek segítségével csoportosítani lehet a felhasználókat a rendszerben, olyan virtuális közösséget létrehozva, amely a valódi világot tükrözi.

Linux Journal 2004. július, 123. szám



Reuven M. Lerner (☞ <http://www.lerner.co.il/atf>)

Nyílt forrású programokra, valamint web- és adatbázis-alkalmazásokra szakosodott tanácsadó.

Könyve, a *Core Perl*, 2002 januárjában jelent meg a Prentice Hall gondozásában. Reuven feleségével és lányaival nemrég költözött Chicagóba.