



# A NÉGY BŰVÖS HÁRMAS

**WWW FTP SQL PHP**

BRKK::Békéscsaba  
Linux rendszergazda képzés  
2008



# Beszerzés

- Bő a választék, mi most a
  - PURE-FTP
  - Apache
  - PHP5
  - MYSQL
  - Postgree SQL
- rendszereket telepítjük



# Telepítés

- Hmmm mondjuk aptitude segítségével telepítsük fel a következő csomagokat, és hagyjuk hadd telepítse a gép a függőségeket:
  - apache2
  - libapache2-mod-php5
  - php5
  - mysql-server-5.0
  - postgresql
  - pure-ftpd



# UTILS

- Néhány hasznos program az SQL rendszerhez
  - php5-mysql, php-MyAdmin
  - php5-pgsql, php-pgAdmin
- Fentiek az  
`apt-get install phpmyadmin phppgadmin`  
parancssal feltelepülnek, persze húzzák magukkal a függőségeik miatt azokat a csomagokat melyek a működésükhöz szükségesek.



# It works

- `ln -s /etc/phpadmin/apache.conf \ /etc/apache/conf.d/phpadmin.conf`
- Nos ha az It works üzenetnél többet szeretnénk:
  - Adj/szerezz nevet a weblapodnak a DNS-ben
  - Kiindulásnak másold le a `/etc/apache2/sites-available/default` fájlt, elvileg mindegy milyen néven, de a rend kedvéért célszerű a DNS nevét adni
  - Szerkesszük át kicsikét, hogy az új site-nak is megfeleljen



# Egy példa site

<VirtualHost \*:80>

ServerAdmin [webmaster@valaki.hol](mailto:webmaster@valaki.hol)

ServerName www.valaki.hol

DocumentRoot /home/valaki.hol/www

<Directory /home/valaki.hol/www/>

Options Indexes FollowSymLinks MultiViews

AllowOverride None

Order allow,deny

allow from all

</Directory>

ScriptAlias /cgi-bin/ /home/valaki.hol/cgi-bin/

<Directory "/usr/lib//home/valaki.hol/cgi-bin">

AllowOverride None

Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch

Order allow,deny

Allow from all

</Directory>

ErrorLog /var/log/apache2/error.valaki.hol.log

LogLevel warn

CustomLog /var/log/apache2/access.valaki.hol.log combined

ServerSignature On

***A [www.valaki.hol](http://www.valaki.hol) értékét természetesen egy igazi névvel helyettesítsd!!!***

</VirtualHost>





# Ha biztonságos kapcsolatot szeretnénk

- **SSL: SECURE SOCKET LAYER**

Az SSL (Secure Socket Layer) egy protokoll réteg, amely a szállítási rétegbeli protokoll (pl. TCP/IP) és valamely alkalmazási rétegbeli protokoll (pl. jelen esetben HTTP) között helyezkedik el, az OSI terminológia szerinti viszony- és megjelenítési réteg feladatait látva el. Webböngészésnél például az SSL biztosítja a biztonságos kommunikációt a kliens (böngésző) és a szerver (webszerver) között. Autentikációhoz digitálisan aláírt tanúsítványokat használ, a kommunikáció titkosítva zajlik (az SSL kézfogás során közösen megegyeznek egy kulcsban, ebből generálják azután az egy session erejéig használatos session keyt, és ezt használják valamely szimmetrikus titkosító algoritmussal, pl. DES, AES, stb).

Az SSL utódja a TLS (TRANSPORT LAYER SECURITY) protokoll, a kettő között gyakorlatilag csak a konstansokban van eltérés, a protokoll maga lényegében változatlan

BŐVEBBEN: <http://wiki.hup.hu/index.php/TLS>



# Ha biztonságos kapcsolatot szeretnénk

- Készítsük el a tanúsítványunkat:

```
mkdir -p /etc/apache2/ssl-certs/valaki.hol
```

```
make-ssl-cert /usr/share/ssl-cert/ssleay.cnf \  
/etc/apache2/ssl-certs/www.valaki.hol/apache.pem
```

- Azaz először készítünk egy könyvtárat a tanúsítványnak, majd hozzuk létre a tanúsítványt magát!



# Ha biztonságos kapcsolatot is szeretnénk



- Az Apache beállításához persze szerkesztenünk kell az előbb létrehozott site fájlunkat is, és persze meg kell mondani az apache-nak, hogy ne csak a 80-as, de a 443-as porton is „hallgatózzon”.
- A `/etc/apache2/ports.conf`-ban:

```
Listen 80
<IfModule mod_ssl.c>
    Listen 443
</IfModule>
```



# Ha biztonságos kapcsolatot is szeretnénk

- A `/etc/apache2/sites-available/www.valaki.hol` - ban hozzunk létre a 443-as portra is VirtualHost bejegyzést a 80-as alapján a következő eltérésekkel:

```
<VirtualHost *:443>
    . . .
    SSLEngine On
    SSLCertificateFile /etc/apache2/ssl-certs/www.valaki.hol/apache.pem
    SSLCertificateKeyFile /etc/apache2/ssl-certs/www.valaki.hol/apache.pem

    DocumentRoot /home/valaki.hol/www-ssl
    <Directory /home/valaki.hol/www-ssl/>
        . . .
    </Directory>
    . . .
</VirtualHost>
```



# APACHE RELOADED

- Még 3 feladatunk van:
  - Bekapcsolni az ssl modult
  - Bekapcsolni a weboldalunkat
  - Újraindítani az Apache-t

```
a2enmod ssl
```

```
a2ensite www.valami.hol
```

```
/etc/init.d/apache restart
```

- *Sajnos az Apache jelenleg csak egy tanúsítványt hajlandó kezelni IP címenként.*



# Pure FTP beállítás

- Miután föltelepítettük a pure-t és vetettünk egy pillantást a `/etc/pure-pftpd/conf` könyvtárra.
- Ebben a könyvtárban található minden egyes fájl, a pure-ftp-nek 1-1 opcióját állítja úgy, hogy az állítandó opció a fájl neve, a beállítandó értéket pedig a fájlba beírt adat határozza meg
- hozzunk létre egy Unix felhasználót, a puredb FTP fiókok ennek a felhasználónak a jogosultságával rendelkezzenek majd  
`addgroup --system ftpgroup`  
`adduser --system --ingroup ftpgroup --home /dev/null --no-create-home --shell=/bin/false ftpuser`



# A Pure FTP beállítása

A beállítandó opciók a következők:  
(A fájlneveknél ügyeljünk a kis- és nagybetűkre)

- `AltLog = clf:/var/log/pure-ftpd/transfer.log`
- `ChrootEveryone = yes`
- `CreateHomeDir = yes`
- `MinUID = 33`
- `NoAnonymous = yes`
- `PassivePortRange = 65500 65534`



# FTP SZERVER CONFIG

Az FTP rendszerünkbe könnyen felvehetünk, törölhetünk felhasználót. Erre a **pure-pw** parancs szolgál.

```
pure-pw useradd <login> [-f <passwd file>] -u <uid> [-g <gid>]
-D/-d <home directory> [-c <gecos>]
[-t <download bandwidth>] [-T <upload bandwidth>]
[-n <max number of files>] [-N <max Mbytes>]
[-q <upload ratio>] [-Q <download ratio>]
[-r <allow client host>[/<mask>][,<allow client host>[/<mask>]]...]
[-R <deny client host>[/<mask>][,<deny client host>[/<mask>]]...]
[-i <allow local host>[/<mask>][,<allow client host>[/<mask>]]...]
[-l <deny local host>[/<mask>][,<deny local host>[/<mask>]]...]
[-y <max number of concurrent sessions>]
[-z <hhmm>-<hhmm>] [-m]
```



# Példa



- Hozzunk létre egy gipsz nevű FTP fiókot, home könyvtára /home/ftpusers/gipsz, a 192.168.1.0/24 IP tartományból engedélyezett számára a kapcsolat, 9.00 - 18.00 időszakban

```
pure-pw useradd gipsz -u ftpuser -d /home/ftpusers/gipsz -r 192.168.1.0/24  
-z 0900-1800
```

```
pure-pw mkdb
```

(A pureftpd.passwd fájlban lévő adatokat speciális adatbázis fájlba konvertálja, ezt is ki kell adni, hogy érvényre lépjenek a változások. Ha a -m kapcsolót használjuk, akkor automatikusan megtörténik)

- Beállítjuk a quota értéket 10MB-ra és 1000 db állományra.

```
pure-pw usermod gipsz -n 1000 -N 10
```

-



# Példa

- Felhasználó adatok megjelenítése:

```
pure-pw show gipsz
```

```
Login                : gipsz
Password             : $1$tny5F4A0$WIC/XGaTGHsGmSJrbOwcJ.
UID                  : 103 (ftpuser)
GID                  : 103 (ftpgroup)
Directory            : /home/ftpusers/gipsz/.
Full name            :
Download bandwidth  : 0 Kb (unlimited)
Upload bandwidth    : 0 Kb (unlimited)
Max files            : 1000 (enabled)
Max size             : 10 Mb (enabled)
Ratio                : 0:0 (unlimited:unlimited)
Allowed local IPs    :
Denied local IPs     :
Allowed client IPs   : 192.168.1.0/24
Denied client IPs    :
Time restrictions    : 0900-1800 (enabled)
Max sim sessions     : 0 (unlimited)
```

- FTP fiók törlése `pure-pw userdel gipsz`



**Köszönöm a figyelmet!**