

# PHP-s CMS-ek / blogok / wikik / stb.

A népszerűbb tartalom kezelő rendszerek telepítése, hasznos kiegészítők

- [Redmine telepítése](#)
- [Hogyan irányíthatom át a WordPress alapú oldalam HTTP protokollról HTTPS protokollra?](#)
- [Hogyan irányíthatom át a weboldalam HTTP protokollról HTTPS protokollra?](#)
- [Hogyan irányíthatom át a Joomla! alapú oldalam HTTP protokollról HTTPS protokollra?](#)
- [1.5.x Joomla! verzió karakterkódolási probléma](#)

# Redmine telepítése

- Létre kell hozni egy adatbázist, és egy DB usert hozzá.
- Következő lépésként a cPanel felületén a `Setup Ruby App` menüpontban meg kell adni a Ruby verzióját (2.1), valamint az applikáció elérési könyvtárát, valamint az URI-t **Az URI nem tartalmazhat semmilyen spec. karaktert („/” „.” „,” „” stb,).** Az adatok megadását követően a **Setup** gombra kell kattintani.
- Ezt követően a Ruby on Rails alkalmazásokhoz a modulokat külön kell telepítenie az ügyfélnek. Ehhez a modules sorban kell a **show** feliratra kell kattintani. A pontos modullistáról az adott alkalmazás rendszerkövetelményeiben tájékozódhat. A telepítés hosszabb rövidebb időt is igénybe vehet (Redmine esetében ez a teszteléskor 07:48.96 perc volt) **Redmine esetében az alábbi modulokat kell telepíteni:**
  - bundle
  - i18n#0.6.11
  - builder#3.0.4
  - rails#3.2.19
  - mime-types#1.25.1
  - mocha#1.0.0
  - jquery-rails#3.1.2
  - coderay
  - fastercsv
  - request\_store
  - rbpdf
  - mysql2
  - selenium-webdriver
  - rmagick
  - shoulda#3.3.2
  - ruby-openid#2.3.0
  - request\_store#1.0.5
  - capybara#2.1.0
  - net-ldap#0.3.1
  - rack-openid
  - shoulda-matchers#1.4.1
  - redcarpet#2.3.0
  - yard
  - rake#10.4.2
  - bigdecimal
- Modul név beírása, majd az **Add** gombra kell kattintani. Ha az összes modult hozzáadtuk, akkor utána az oldal alján az **Update** gombra kell kattintani. Ezt követően ki kell várni ameddig az adott modul fordítása és telepítése megtörténik, amint a telepítés megtörtént akkor a `Extensions set successfully updated` üzenetet kapjuk.
- kattintsunk a **Restart** gombra ha elkészült.

- Ezt követően SSH konzolon töltsük le a Redmine csomagot:

```
wget url http://www.redmine.org/releases/redmine-x.x.x.tar.gz
```

- Majd tömörítsük ki:

```
tar xzf redmine-x.x.x.tar.gz
```

- Ezt követően másoljuk át a tartalmat a korábban létrehozott redmine mappánkba

```
cp -R ~/redmine-x.x.x/* ~/redmine
```

- Készítsünk másolatot, majd állítsuk be az adatbázis csatlakozáshoz szükséges adatokat a production résznél:

```
cp ~/redmine/config/database.yml.example ~/redmine/config/database.yml
```

```
nano ~/redmine/config/database.yml
```

A konfigurációt F2-vel mentjük el.

- Ezt követően másoljuk át a public mappát a /public\_html/redmine/ mappába.

```
cp -R ~/redmine/public/* ~/public_html/redmine/
```

- Lépjünk be a Redmine mappába.

```
cd ~/public_html/redmine
```

- Másoljuk át a htaccess.fcgi.example, valamint a dispatch.fcgi.example fájlokat.

```
cat htaccess.fcgi.example >> .htaccess
```

```
cp dispatch.fcgi.example dispatch.fcgi
```

- Ezt követően lépünk vissza abba a mappába ahová kitömörítettük a Redmine-t

```
cd ~/redmine
```

- A `Gemfile` fájlhoz adjuk hozzá az alábbi sort:

```
gem "bigdecimal"
```

- Futtassuk le az alábbi parancsokat:

Aktiválás:

```
source ~/rubyenv/redmine/2.1/bin/activate
```

Telepítés:

```
~/rubyvenv/redmine/2.1/bin/bundle install
```

Titkos kulcs generálása:

```
~/rubyvenv/redmine/2.1/bin/rake generate_secret_token
```

Adatbázis betöltése:

```
RAILS_ENV=production ~/rubyvenv/redmine/2.1/bin/rake db:migrate
```

Mintaadatok telepítése:

```
RAILS_ENV=production ~/rubyvenv/redmine/2.1/bin/rake redmine:load_default_data
```

- Végül a cPanel felületén indítsuk újra az alkalmazást a **Restart** gomb segítségével.

# Hogyan irányíthatom át a WordPress alapú oldalam HTTP protokollról HTTPS protokollra?

A WordPress tartalom kezelő rendszer esetében az adminisztrációs felületen keresztül lehet beállítani azt, hogy az érintett oldal `https://` címen keresztül legyen elérhető.

A beállítás menete a következő:

- Bejelentkezést követően A **Vezérlőpult** -> **Beállítások** -> **Általános** részen a **WordPress cím (URL)** és a **Honlap cím (URL)** mezőkben kell módosítani a jelenlegi `http://` előtagot `https://`-re.

A mentést követően a WordPress ki fogja jelentkeztetni a bejelentkezett felhasználót, annak újra be kell jelentkeznie.

- Ezt követően egy FTP kliens segítségével be kell jelentkezni a tárhelyre, és a domain névhez tartozó dokumentum root könyvtárban található **.htaccess** fájlt, illetve a **wp-config.php** fájlt kell szerkeszteni.

A .htaccess fájl elejére az alábbi kódot kell beilleszteni:

```
RewriteEngine On
RewriteCond %{SERVER_PORT} 80
RewriteRule ^(.*)$ https://domain-nev.hu/$1 [R,L]
```

Az utolsó sorban található `domain-nev.hu`-t módosítani kell arra a domain névre amely alatt a WordPress üzemel.

- A `wp-config.php` fájlban meg kell keresni az „Ennyi volt kellemes bloggolást” sort , és fölé kell elhelyezni az alábbi kódrészletet:

```
define('FORCE_SSL_ADMIN', true);  
// in some setups HTTP_X_FORWARDED_PROTO might contain  
// a comma-separated list e.g. http,https  
// so check for https existence  
if (strpos($_SERVER['HTTP_X_FORWARDED_PROTO'], 'https') !== false)  
    $_SERVER['HTTPS']='on';
```

- Ahhoz, hogy a közvetlen hivatkozások is helyesen jelenjenek meg a elegendő ha a Közvetlen hivatkozások menüpontban rákattintunk a mentés gombra. Ekkor a WordPress automatikusan újra fogja generálni a .htaccess fájlt.

**Amennyiben bármilyen további kérdése van, forduljon Ügyfélszolgálatunkhoz bizalommal.**

# Hogyan irányíthatom át a weboldalam HTTP protokollról HTTPS protokollra?

## AZ ÖSSZES FORGALOM ÁTIRÁNYÍTÁSA

Ahhoz, hogy az összes HTTP-re érkező kérést HTTPS protokollra tudja irányítani az alábbi kódsorozatot kell a domain névhez tartozó dokumentum\_root könyvtárban található .htaccess fájlban szerepeltetnie:

```
RewriteCond %{REQUEST_URI} !^[0-9]+\.\.+\.cpanel$
RewriteCond %{REQUEST_URI} !^\.well-known/pki-validation/[A-F0-9]{32}\.txt(?:\ Comodo\ DCV)?$
RewriteEngine On
RewriteCond %{SERVER_PORT} 80
RewriteRule ^(.*)$ https://www.példa.hu/$1 [R=301,L]
```

Az utolsó sorban szereplő *www.példa.hu* domain nevet mindenképpen cserélje ki arra a domain névre amelynél szeretné beállítani a HTTPS protokollt.

## ADOTT DOMAIN NÉV ÁTIRÁNYÍTÁSA

Ha csak egy adott domain név esetében szeretné átírányítani HTTPS protokollra akkor az alábbi kódsorozatot kell a domain névhez tartozó dokumentum\_root könyvtárban található .htaccess fájlban szerepeltetnie:

```
RewriteCond %{REQUEST_URI} !^[0-9]+\.\.+\.cpanel$dcv$
RewriteCond %{REQUEST_URI} !^\.well-known/pki-validation/[A-F0-9]{32}\.txt(?:\ Comodo\ DCV)?$
RewriteEngine On
RewriteCond %{HTTP_HOST} ^példa.hu [NC]
RewriteCond %{SERVER_PORT} 80
RewriteRule ^(.*)$ https://www.példa.hu/$1 [R=301,L]
```

Figyeljen arra, hogy a *példa.hu* és a *www.példa.hu* domain nevet mindenképpen cserélje ki arra a domain névre amelynél szeretné beállítani a HTTPS protokollt.

# ÁTIRÁNYÍTÁS EGY SPECIÁLIS KÖNYVTÁRRA.

Ahhoz, hogy az összes HTTP-re érkező kérést HTTPS protokollra, és azon belül egy megadott könyvtárra tudja irányítani az alábbi kódsorozatot kell a domain névhez tartozó `document_root` könyvtárban található `.htaccess` fájlban szerepeltetnie:

```
RewriteCond %{REQUEST_URI} !^[0-9]+\.\.+\.cpanel$dcv$
RewriteCond %{REQUEST_URI} !^\.well-known/pki-validation/[A-F0-9]{32}\.txt(?:\ Comodo\ DCV)?$
RewriteEngine On
RewriteCond %{SERVER_PORT} 80
RewriteCond %{REQUEST_URI} konyvtar
RewriteRule ^(.*)$ https://www.példa.hu/konyvtar/$1 [R=301,L]
```

A *konyvtar* könyvtár nevet, valamint a *www.példa.hu/konyvtar* domain nevet mindenképpen cserélje ki arra a domain névre, és könyvtárra amelynél szeretné beállítani a HTTPS protokollt.

# Hogyan irányíthatom át a Joomla! alapú oldalam HTTP protokollról HTTPS protokollra?

1. Az átirányítás beállításához a domain névhez tartozó document root könyvtárban található *configuration.php* fájlt, valamint a *.htaccess* fájlt kell szerkeszteni. Egy FTP klines program segítségével jelentkezzen be a tárhelyére, majd nyissa meg egy szövegszerkesztővel a *configuration.php* fájlt. A fájlban keresse meg az alábbi változót:

```
var $live_site ='';
```

Ezt a változót kell módosítania a következők szerint:

```
var $live_site = 'https://www.your-domain.com';
```

A módosításokat követően mentse el a fájlt, majd írja felül az eredeti fájlt. Következő lépésként nyissa meg a *.htaccess* fájlt, és a fájl végére illessze be az alábbi kódrészletet:

```
RewriteEngine On
RewriteCond %{HTTPS} OFF
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
```

2. A további beállításokat a Joomla! adminisztrációs felületén keresztül lehet elvégezni.
  - Bejelentkezést követően navigáljon a **Globális konfiguráció** menüpontba
  - A **Kiszolgáló** fülön a **Kiszolgáló beállításai** szekcióban be kell állítani a **HTTPS kényszerítése** résznél a **Teljes webhelyre** vonatkozóan.

**Amennyiben bármilyen további kérdése van, forduljon Ügyfélszolgálatunkhoz bizalommal.**

# 1.5.x Joomla! verzió

## karakterkódolási probléma

A tárhelyes szervereink esetében az 1.5.x-s Joomla! telepítések esetében sajnálatos módon katakterkódolási probléma jelentkezik. A problémát az okozza, hogy a szervereinken a közelmúltban a MySQL 5.6-os szerver szoftver lecserélésre került a MariaDB 10.1.21-es verziójára. Az 1.5-ös Joomla verziók esetében sajnos az UTF támogatást legalább a MySQL 4.0-ás verziójának meglétéhez kötik.

A probléma megoldásához módosítani kell az alábbi fájlokat:

libraries/joomla/database/database/mysql.php  
libraries/joomla/database/database/mysqli.php

A fájlokban meg kell keresni az alábbi kódrészletet:

```
/**
 * Determines UTF support
 *
 * @access public
 * @return boolean True - UTF is supported
 */
function hasUTF()
{
    $verParts = explode( '.', $this->getVersion() );
    return ($verParts[0] == 5 || ($verParts[0] == 4 && $verParts[1] == 1 && (int)$verParts[2] >= 2));
}
```

Majd ki kell kommentezni a `return ($verParts[0] == 5 || ($verParts[0] == 4 && $verParts[1] == 1 && (int)$verParts[2] >= 2);` sort, és alá beírni a következő kódrészletet:

```
return 1;
```

Módosítást követően a következőképpen néz ki a kódrészlet:

```
/**
```

```
* Determines UTF support
*
* @access public
* @return boolean True - UTF is supported
*/
function hasUTF()
{
    $verParts = explode( '.', $this->getVersion() );
//    return ($verParts[0] == 5 || ($verParts[0] == 4 && $verParts[1] == 1 && (int)$verParts[2] >= 2));
return 1;
}
```

A módosítást követően az oldal ismételten rendben meg fogja jeleníteni az ékezetes karaktereket is.

**Amennyiben bármilyen további kérdése van, forduljon Ügyfélszolgálatunkhoz bizalommal.**