Tartalomjegyzék

Weboldalüzemeltetői praktikák

Itt próbáljuk összefoglalni azokat a - szerintünk - „best practices”, azaz jó gyakorlatot megvalósító beállításokat egy weboldal esetén. A javaslatok a web gyökérmappájában található .htaccess fájlt érintik. Többnyire az oldal biztonságát vagy és/vagy az oldal betöltési sebességét növeli minden beállítás. A leírt beállítások nem új keletűek, de igyekszünk egy áttekinthető összefoglalót adni. A CMS rendszerek (Joomla, Wordpress, stb.) egyéb beállításaival és használatával itt nem foglalkozunk, mert azt már nagyon sok helyen leírták az interneten. A doimain.hu -t mindig cserélje ki a saját domainjére.

Átlag weboldal

www átirányítása www nélküli oldalra, pl.: www.mav-it.hu ⇒ mav-it.hu

```
#### www => non-www ####
RewriteCond %{HTTP_HOST} ^www\.domain\.hu [NC]
RewriteRule ^(.*)$ https://domain.hu/$1 [L,R=301]
```

http: *oldal átirányítása https:* oldalra, ha erőltetni szeretnénk a titkosított adatforgalmat

```
#### http => https ####
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

Gzip oldaltömörítés bekapcsolása. Drasztikusan csökkentheti az oldal betöltési sebességét (a szerveren is telepítve kell lennie a modulnak!).

```
#### GZIP ####
<ifmodule mod_deflate.c>
AddOutputFilterByType DEFLATE text/text text/html text/plain text/xml
text/css application/x-javascript application/javascript
</ifmodule>
```

Hotlink védelem. Ez a kicsi kódocska sok sávszélességet spórolhat nekünk, illetve a webszerver üzemeltetőjének. Megakadályozza, hogy a weboldal fájljait más weboldalak is használják, ezáltal „ellopják” az Ön sávszélességét.

```
#### Hotlink protection ####
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^https://(.+\.)?domain\.hu/ [NC]
RewriteRule \.(gif|jpg|jpeg|bmp|zip|rar|xml|png|css|pdf|js)$ - [F]
```

Az Apache alapértelmezésben (ha a könyvtár nem tartalmaz index.html vagy index.php fájlt) megmutatja a könyvtár tartalmát. Amennyiben ezt nem tartjuk kívánatosnak, akkor helyezzük ezl az alábbi kódpt a .htaccess fájlban:

```
Options -Indexes
```

Az alábbi kód megakadályozza számos exploit becsempészését az URL-be.

```
# Szűrünk minden base64 kódolt tartalmat az URL-ben
RewriteCond %{QUERY_STRING} base64_encode[^(]*\([^)]*\) [OR]
#
# Szűrjük a <script></script> tageket az URL-ben
RewriteCond %{QUERY_STRING} (<|%3C)([^s]*s)+cript.*(>|%3E) [NC,OR]
#
# Megakadályozzuk a PHP GLOBALS változók (át)írását
RewriteCond %{QUERY_STRING} GLOBALS(=|\[|\%[0-9A-Z]{0,2}) [OR]
#
# Megakadályozzuk a _REQUEST változó átírását
RewriteCond %{QUERY_STRING} _REQUEST(=|\[|\%[0-9A-Z]{0,2})
#
# a fenti feltételek esetén 403 Forbidden header-t ad vissza a böngészőnek
RewriteRule .* index.php [F]
```

Wordpress praktikák

A wordpress egyik sebezhetősége, hogy a gyökérmappában található xmlrpc.php fájlon keresztül az adatbázishoz próbálnak hozzáférni akár brute force technikával. Ez a kis kód elérhetetlenné teszi ezt a fájlt (illetve átirányítja, és „no” szócskát láthatunk a fájl betöltésekor). (Természetesen ha használni akarjuk az RPC szolgáltatást akkor NE használjuk ezt a kódocskát!)

```
#### xmlrpc attack protection ####
Redirect 403 /xmlrpc.php
ErrorDocument 403 "no"
```

Vannak bizonyos könyvtárak, amikben nem szabad .php fájlokat futtatni. Ilyen pl a /wp-includes és a /wp-content/uploads könyvtárak. Ezekbe tegyünk bele egy-egy üres .htaccess fájlt, és másoljuk bele az alábbi kódsort. Ezzel megakadályozzunk, hogy a rosszindulatú PHP kódot futtasson valaki ezekben a könyvtárakban.

```
<Files *.php>
deny from all
</Files>
```

Talán az egyik legkényesebb fájl a gyökérmappában lévő wp-config.php. Sok kényes adat között tartalmazza pl. az adatbázis hozzáférési adatait is. Ezt az alábbi kódocskát elhelyezve a gyökérkönyvtárban lévő .htaccess file-ban megakadályozzuk, hogy a kiszolgáló IP címétől eltérő IP címről elérjék a fájlt.

```
<files wp-config.php>
order allow,deny
deny from all
</files>
```