

21 Reglas de .htaccess que debes saber

La configuración del archivo .htaccess de Apache (hypertext access) puede ser una herramienta muy ponderosa en el desarrollo web si se utiliza de la forma adecuada. 

Mediante .htaccess podremos realizar redirecciones, proteger archivos y directorios con contraseña, prevenir el hotlinking y el robo de ancho de banda, comprimir archivos y mucho, mucho más. Entérate cómo. 

Asegúrate de hacer un backup de tu archivo .htaccess actual antes de aplicar cualquiera de estas reglas.

1. Prevenir el Hotlinking

¿Estás cansado de que la gente utilice tu ancho de banda poniendo las imágenes albergadas en tu servidor en su sitio web? Añade el siguiente código al final de tu archivo .htaccess para prevenir el hotlinking:

```
1 Options +FollowSymlinks
2 #Protect against hotlinking
3 RewriteEngine On
4 RewriteCond %{HTTP_REFERER} !^$
5 RewriteCond %{HTTP_REFERER} !^http://(www.)?nombrededominio.com/ [nc]
6 RewriteRule .*.(gif|jpg|png)$
  http://nombrededominio.com/img/no_robar_ancho_de_banda.gif[nc]
```

2. Bloquea todas las peticiones del user-agent

Es posible bloquear a todos los user-agents indeseados que pueden ser potencialmente dañinos o quizá simplemente para mantener la carga del servidor lo más baja posible.

```
1 #Block bad bots
2 SetEnvIfNoCase user-Agent ^FrontPage [NC,OR]
3 SetEnvIfNoCase user-Agent ^Java.* [NC,OR]
4 SetEnvIfNoCase user-Agent ^Microsoft.URL [NC,OR]
5 SetEnvIfNoCase user-Agent ^MSFrontPage [NC,OR]
6 SetEnvIfNoCase user-Agent ^Offline.Explorer [NC,OR]
7 SetEnvIfNoCase user-Agent ^[Ww]eb[Bb]andit [NC,OR]
8 SetEnvIfNoCase user-Agent ^Zeus [NC]
9
10 Order Allow,Deny
11 Allow from all
```

12 Deny from env=bad_bot

3. Redirecciona a todos, excepto a ciertas IPs

Si por alguna razón deseas denegar el acceso a todos los usuarios o sólo permitir a un cierto grupo de direcciones IP para que tengan acceso a tu sitio, añade el siguiente código a tu archivo .htaccess:

```
1 ErrorDocument 403 http://www.nombrededominio.com
2 Order deny,allow
3 Deny from all
4 Allow from 124.34.48.165
5 Allow from 102.54.68.123
```

4. Redirecciones 301 amigables con SEO

Si has transferido nombres de dominio o deseas redireccionar a una página específica sin que afecte a los resultados de los motores de búsqueda como Google, utiliza el siguiente código:

```
1 Redirect 301 /d/file.html http://www.nombrededominio.com/r/file.html
```

5. Crea una página personalizada para cada tipo de error

Si te cansaste del diseño por defecto de las páginas de error 401, 403, 404, 500, etc. puedes fácilmente crear la tuya propia y referirte hacia ella de esta forma:

```
1 ErrorDocument 401 /error/401.php
2 ErrorDocument 403 /error/403.php
3 ErrorDocument 404 /error/404.php
4 ErrorDocument 500 /error/500.php
```

6. Crea una lista de IPs prohibidas

¿Te cansaste de obtener siempre los mismos comentarios de ciertos usuarios una y otra vez? Simplemente prohíbe el acceso a determinadas IPs, añadiendo el siguiente código:

```
1 allow from all
2 deny from 145.186.14.122
3 deny from 124.15
```

7. Setea una dirección de e-mail por defecto para el administrador de servidor

Con el siguiente código puedes especificar la dirección de correo electrónico por defecto para el administrador de servidor:

```
1 ServerSignature EMail
2 SetEnv SERVER_ADMIN default@dominio.com
```

8. Deshabilita la ventana de aviso de descarga

Usualmente al descargar algo de un sitio web, saltará una ventana que pregunta si deseas abrir el archivo o guardarlo en tu PC. Para prevenir esto, utiliza el siguiente código:

- 1 AddType application/octet-stream .pdf
 - 2 AddType application/octet-stream .zip
 - 3 AddType application/octet-stream .mov
-

9. Protege un archivo específico

El código que sigue te permite denegar el acceso a cualquier archivo que desees, mostrando una página de error 403 cuando alguien intenta acceder al mismo:

- 1 #Protect the .htaccess File
 - 2
 - 3 order allow,deny
 - 4 deny from all
-

10. Comprime componentes con mod_deflate

Como alternativa a la compresión de archivos mediante Gzip, puedes implementar mod_deflate (que se supone es más rápido). Coloca el siguiente código en la parte superior de tu archivo .htaccess:

- 1 SetOutputFilter DEFLATE
-

11. Agrega headers de expiración a los archivos

El siguiente código muestra cómo agregar fechas de expiración en las cabeceras de los archivos:

- 1 Header set Expires "Wed, 21 May 2010 20:00:00 GMT"
-

12. Seteando la página por defecto

Puedes hacer que la página a mostrar ni bien un usuario llegue a tu sitio no sea la archiconocida "index". Aquí seteamos "about.html" como página de inicio:

- 1 #Serve Alternate Default Index Page
 - 2 DirectoryIndex about.html
-

13. Protege tus archivos y directorios con una contraseña

- 1 #proteger un archivo con contraseña
- 2
- 3 AuthType Basic
- 4 AuthName "Prompt"
- 5 AuthUserFile /home/path/.htpasswd
- 6 Require valid-user
- 7

```
8 # proteger un directorio con contraseña
9 resides
10 AuthType basic
11 AuthName "This directory is protected"
12 AuthUserFile /home/path/.htpasswd
13 AuthGroupFile /dev/null
14 Require valid-user
```

14. Hacer una redirección 301 de un dominio viejo a uno nuevo

Incluyendo este snippet en el .htaccess de tu viejo dominio, conseguirás redirigir a otro nuevo (sin ser penalizado por Google por duplicar contenido, dado que el 301 es un aviso de "redirección permanente") :

```
1 # Redirigir de un dominio viejo a uno nuevo
2 RewriteEngine On
3 RewriteRule ^(.*)$ http://www.nuevodominio.com/$1 [R=301,L]
```

15. Forzar el cacheo

El siguiente snippet no va a incrementar directamente la velocidad de carga de tu sitio en general, pero sí va a cargar más rápido cuando el mismo usuario vuelva a visitarlo al enviar un status 304 cuando se pidan elementos que no han sido modificados. Puedes cambiar la frecuencia de cacheo al cambiar el número de segundos (en este ejemplo se da una vez al día):

```
1 FileETag MTime Size
2 ExpiresActive on
3 ExpiresDefault "access plus 86400 seconds"
```

16. Comprimir componentes al activar Gzip

Esto hace que sea más rápida la carga de tu sitio:

```
1 AddOutputFilterByType DEFLATE text/html text/plain text/xml application/xml
2 application/xhtml+xml text/javascript text/css application/x-javascript
3 BrowserMatch ^Mozilla/4 gzip-only-text/html
4 BrowserMatch ^Mozilla/4.[678] no-gzip
5 BrowserMatch bMSIE !no-gzip !gzip-only-text/html
```

17. Quitar palabras o strings de una URL

En este ejemplo, quitaremos "category" de nuestras URL, es decir: <http://tudominio.com/category/blue> -> <http://tudominio.com/blue>

Agrega este código al final de tu .htaccess:

```
1 RewriteRule ^category/(.+)$ http://www.tudominio.com/$1 [R=301,L]
```

18. Deshabilitar la navegación por los directorios de tu sitio

```
1 Options All -Indexes
```

19. Redirigir los Feeds de WordPress a FeedBurner

```
1 #Redirect wordpress content feeds to feedburner
2
3 RewriteEngine on
4 RewriteCond %{HTTP_USER_AGENT} !FeedBurner [NC]
5 RewriteCond %{HTTP_USER_AGENT} !FeedValidator [NC]
6 RewriteRule ^feed/?([_0-9a-z-]?)?/?$ http://feeds.feedburner.com/tufeed [R=302,NC,L]
```

20. Remover la extensión del archivo de una URL

Para que sea más amigable al usuario y a los buscadores, por ejemplo
"http://www.tudominio.com/autos.html" -> "http://www.tudominio.com/autos"

```
1 RewriteRule ^(([/+])*(^.)+)$ /$1.php [L]
```

21. Quitar el "www" del dominio

```
1 #remove www from URI
2 RewriteEngine On
3 RewriteCond %{HTTP_HOST} ^www\.dominio\.com$ [NC]
4 RewriteRule ^(.*)$ http://dominio.com/$1 [L,R=301]
5
```

Sigue mis publicaciones:

Para poder ver los links necesitas estar [Registrado](#) . O.. ya tienes cuenta? [Logueate!](#)

Para poder ver los links necesitas estar [Registrado](#) . O.. ya tienes cuenta? [Logueate!](#)

La paciencia es el arbol de raices amargas, pero da dulces frutos.

- [Compartir](#)
- 0
- Compartir en: