

4 Simple and Effective Ways to Use Your .htaccess File

The .htaccess file is an effective way to simplify tasks on your web site. This simple text-based file holds server directives (commands used by the server). When a web request comes in, this file tells the web server how to process the request.

Understanding how to manipulate this file gives you more control and the added abilities to improve the functionality of some basic web server operations. Ready to get started? Check out these 4 ways to get more out of your .htaccess file:

Spice Up Those Errors

Standard web error messages are confusing and boring to your site's visitors. How many times do you think they want to see 404 Page not found?

Why not spice up your error pages with your .htaccess file?

With .htaccess, you can add paths to custom error pages instead of those old and boring error pages. In your custom error pages, you can have links to drive your visitors to other pages in your site instead of them leaving.

For instance, you want to send someone to a customized 404 page called 404.html in your home directory, you would just add:

```
ErrorDocument 404 /404.html
```

That's it!

Now when someone encounters a not found error on your site, the .htaccess file will redirect them to the 404.html page instead of the standard error message.

You can do this for any of the http errors on your site.

Keeping Things Private

The .htaccess file is great for adding password protection to specific areas of your site. Areas you may want to password protect would be: admin areas, work not ready for production or just private stuff.

When you implement this in your .htaccess file, anyone accessing that area will receive a message for their username and password. Without login information, you can't get in.

Say you want to create a login for someone named Mack, and allow him to access a directory - here's how you do it:

- Login to your UNIX shell with administrator rights.
- Execute the command: `htpasswd -c .htpasswd Mack`
- When prompted, type the password twice.
- Create a blank .htaccess file.
- Add the following to your .htaccess file:
AuthType Basic
AuthUserFile /pathtoprotecteddirectory/.htaccess
AuthName "Authorization Required"
Require Valid-User
- Put the .htaccess file in the directory you want to protect.

Now if Mack wants to access that directory, he will need to enter his username and password.

Detour Up Ahead

Now here's a neat little trick.

Have you ever had to change a URL on your site or move content from one web site to another? With the .htaccess file, you can automate this easily.

For internal redirects, you can just add the following:

Redirect /oldfile.html /newfile.html

Redirecting to a new site? No problem! Use this:

Redirect /file.html http://www.newsite.com/newfile.html

See? Simple huh?

This is a simple and effective way to get your visitors to the right content.

Keep Unwanted Guests Out

There are troublemakers everywhere - this includes online as well. They just don't want to stop causing trouble and won't get off of your lawn (or web site in this case).

Guess what? You can use that handy .htaccess file again to put a stop to those pesky troublemakers.

If you know where they are coming from, such as their IP address or domain, you can ban them from your site by adding this to your .htaccess file:

order allow,deny

deny from 168.91.22.19 (this would be the IP address they are coming from)

deny from troublemakers.net (this would be the domain they are coming from)

allow from all

Now you are blocking both IP address and domain of those troublemakers from accessing your site. I do want to warn you to check to see if the IP address and domain are correct before you execute this.

Effective and Simple

There they are, some useful and yet, simple ways to get more out of your.htaccess file. From moving web sites to keeping troublemakers away, this little file will help you manage your web site effectively and with minimal effort.