**.htaccess**

1. **Introduction**

.htaccess is a configuration file for use on web servers running the Apache Web Server software. When a .htaccess file is placed in a directory which is in turn 'loaded via the Apache Web Server', then the .htaccess file is detected and executed by the Apache Web Server software. These .htaccess files can be used to alter the configuration of the Apache Web Server software to enable/disable additional functionality and features that the Apache Web Server software has to offer.

1. **[How to use .htaccess](http://www.htaccess-guide.com/how-to-use-htaccess/" \o "How to use .htaccess)**

'.htaccess' is the filename in full, it is not a file extension. For instance, you would not create a file called, 'file.htaccess', it is simply called, '.htaccess'. This file will take effect when placed in any directory which is then in turn loaded via the Apache Web Server software. The file will take effect over the entire directory it is placed in and all files and subdirectories within the specified directory.

Here is an example of what you might include in a .htaccess file.

|  |  |
| --- | --- |
| |  | | --- | | AuthName "Member's Area Name" AuthUserFile /path/to/password/file/.htpasswd AuthType Basic require valid-user ErrorDocument 401 /error\_pages/401.html AddHandler server-parsed .html | |

1. **Error Documents**

To set-up custom error documents, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | ErrorDocument 404 /error\_pages/404.html | |

To setup further error documents, for example for '401 Unauthorised', '403 Forbidden', and '500 Internal Server' error messages, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | ErrorDocument 401 /error\_pages/401.html ErrorDocument 404 /error\_pages/404.html ErrorDocument 500 /error\_pages/500.html | |

1. **Redirects**

Redirects enable us to direct web site visitors from one document within your web site to another. This is useful for example, if you have moved your web site content and would like to redirect visitors from old links to the new content location.

To set-up redirects, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | Redirect /old\_dir/ http://www.yourdomain.com/new\_dir/index.html | |

1. **Password Protection**

The password protection and authentication systems offered by the [Apache Web Server](http://httpd.apache.org/) are probably the most important use of .htaccess files. Very easily, we can password protect a directory (or multiple) of a web site which require a username and password to access. The login procedure for these secure directories is handled automatically by the web browser using a pop-up login interface.

To begin, decide which directory you would like to password protect (note that all files and subdirectories within the directory will be password protected), then create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | AuthName "Member's Area Name" AuthUserFile /path/to/password/file/.htpasswd AuthType Basic require valid-user | |

The first line tells the Apache Web Server the secure directory is called 'Member's Area Name', this will be displayed when the pop-up login prompt appears. The second line specifies the location of the password file. The third line specifies the authentication type, in this example we are using 'Basic' because we are using basic HTTP authentication and finally the fourth line specifies that we require valid login credentials, this line can also be used to specify a specific username, e.g. 'require user username' would require the username 'username'.

The location of the password file can be anywhere on your web server, the '/location/of/password/file/' must be replaced with the full/absolute path to the directory containing the password file, and the '.htpasswd' file must exist, this can however be called anything. We use the filename '.htpasswd' because the server will recognise the filename and will hide it from visitors.

The password file would contain something similar to the following text:

|  |  |
| --- | --- |
| |  | | --- | | username:encryptedpassword fred\_smith:oCF9Pam/MXJg2 | |

Now, you cannot just make up the password, on Unix/Linux servers they must be encrypted by the server, on Windows servers you do just use a plain text password as Windows does not offer any encryption methods. You can have any number of user records in your password file, one account per row, separating the username and password with a colon.

1. **Deny Visitors by IP Address**

The visitor blocking facilities offered by the Apache Web Server enable us to deny access to specific visitors, or allow access to specific visitors.

To set-up visitors restrictions and blocking, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | order allow,deny deny from 255.0.0.0 deny from 123.45.6. allow from all | |

To set-up blocking of all visitors except yourself, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | order allow,deny allow from 255.0.0.0 deny from all | |

1. **Deny Visitors by Referrer**

If you've ever looked at your logs and noticed a surprising increase in traffic, yet no increases in actual file requests it's probably someone pinching content (such as CSS files) or someone attempting to hack your web site (this may simply mean trying to find non public content).

To set-up block a single referrer, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | RewriteEngine on # Options +FollowSymlinks RewriteCond %{HTTP\_REFERER} otherdomain\.com [NC] RewriteRule .\* - [F] | |

The above lines tell the Apache Web Server to block traffic from the URL 'otherdomain.com'. The '[NC]' text after the referrer specifies it as not case-sensitive. Which prevents traffic from 'OtherDomain.com', 'otherdomain.com', 'OTHERDOMAIN.COM' and so on.

To set-up block multiple referrers, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | RewriteEngine on # Options +FollowSymlinks RewriteCond %{HTTP\_REFERER} otherdomain\.com [NC,OR] RewriteCond %{HTTP\_REFERER} anotherdomain\.com RewriteRule .\* - [F] | |

1. **Hotlink Prevention Techniques**

Hot link prevention refers to stopping web sites that are not your own from displaying your files or content, e.g. stopping visitors from other web sites. This is most commonly used to prevent other web sites from displaying your images but it can be used to prevent people using your JavaScript or CSS (cascading style sheet) files. The problem with hot linking is it uses your bandwidth, which in turn costs money, hot linking is often referred to as 'bandwidth theft'.

To set-up hot link prevention for '.gif', '.jpg' and '.css' files, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | RewriteEngine on RewriteCond %{HTTP\_REFERER} !^$ RewriteCond %{HTTP\_REFERER} !^http://(www\.)?yourdomain.com/.\*$ [NC] RewriteRule \.(gif|jpg|css)$ - [F] | |

The above lines tell the Apache Web Server to block all links to '.gif', '.jpg' and '.css' files which are not from the domain name 'http://www.yourdomain.com/'. Before uploading your .htaccess file ensure you replace 'yourdomain.com' with the appropriate web site address.

To set-up hot link prevention for '.gif', '.jpg' files which displays alternate content (such as an angry man image), create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | RewriteEngine on RewriteCond %{HTTP\_REFERER} !^$ RewriteCond %{HTTP\_REFERER} !^http://(www\.)?yourdomain.com/.\*$ [NC] RewriteRule \.(gif|jpg)$ http://www.yourdomain.com/hotlink.jpg [R,L] | |

The above lines tell the Apache Web Server to block all links to '.gif' and '.jpg' files which are not from the domain name 'http://www.yourdomain.com/' and to display the file 'http://www.yourdomain.com/hotlink.jpg' instead. Before uploading your .htaccess file ensure you replace 'yourdomain.com' with the appropriate web site address.

1. **Blocking Offline Browsers and Bad Bots**

Offline browsers are pieces of software which download your web page, following the links to your other web pages, downloading all the content and images. The purpose of this is innocent, so the visitor can log off the Internet and browse the site without a connection, but the demand on the server and bandwidth usage can be expensive. Bad bots as they are often called refers to programs which visit your web site, either to source content, look for security holes or to scan for email addresses. This is often how your email address ends up on 'Spam' databases, because they have set a 'bot' to scan the Internet and collect email addresses. These programs and 'bots' often ignore the rules set out in 'robot.txt' files.

Below is a useful example of how to block some common 'bots' and site rippers. Create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | RewriteEngine On  RewriteCond %{HTTP\_USER\_AGENT} ^BlackWidow [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Bot\ mailto:craftbot@yahoo.com [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^ChinaClaw [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Custo [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^DISCo [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Download\ Demon [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^eCatch [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^EirGrabber [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^EmailSiphon [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^EmailWolf [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Express\ WebPictures [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^ExtractorPro [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^EyeNetIE [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^FlashGet [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^GetRight [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^GetWeb! [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Go!Zilla [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Go-Ahead-Got-It [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^GrabNet [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Grafula [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^HMView [OR]  RewriteCond %{HTTP\_USER\_AGENT} HTTrack [NC,OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Image\ Stripper [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Image\ Sucker [OR]  RewriteCond %{HTTP\_USER\_AGENT} Indy\ Library [NC,OR]  RewriteCond %{HTTP\_USER\_AGENT} ^InterGET [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Internet\ Ninja [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^JetCar [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^JOC\ Web\ Spider [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^larbin [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^LeechFTP [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Mass\ Downloader [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^MIDown\ tool [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Mister\ PiX [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Navroad [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^NearSite [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^NetAnts [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^NetSpider [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Net\ Vampire [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^NetZIP [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Octopus [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Offline\ Explorer [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Offline\ Navigator [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^PageGrabber [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Papa\ Foto [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^pavuk [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^pcBrowser [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^RealDownload [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^ReGet [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^SiteSnagger [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^SmartDownload [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^SuperBot [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^SuperHTTP [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Surfbot [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^tAkeOut [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Teleport\ Pro [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^VoidEYE [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Web\ Image\ Collector [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Web\ Sucker [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebAuto [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebCopier [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebFetch [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebGo\ IS [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebLeacher [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebReaper [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebSauger [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Website\ eXtractor [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Website\ Quester [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebStripper [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebWhacker [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WebZIP [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Wget [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Widow [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^WWWOFFLE [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Xaldon\ WebSpider [OR]  RewriteCond %{HTTP\_USER\_AGENT} ^Zeus  RewriteRule ^.\* - [F,L] | |

1. **DirectoryIndex Uses**

The directoryindex command allows you to specify a default page to display when a directory is accessed. For instance, if a visitor requests a directory on your web site, you can specify the file to load when the directory is accessed (if a filename is not specified in the initial request). For example, to display a 'index.html' file rather than showing directory listings or to load a 'index.php' file rather than an 'index.html' file.

To set-up a directoryindex, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | DirectoryIndex index.html | |

The above lines tell the Apache Web Server to display the 'index.html' file, whenever the directory containing this .htaccess file (or any subdirectory) is accessed.

We can setup a directoryindex to call multiple files using the following text:

|  |  |
| --- | --- |
| |  | | --- | | DirectoryIndex index.html index.cgi index.php | |

The above lines tell the Apache Web Server to display the 'index.html' file as the directoryindex, if this file is not available then display 'index.cgi', and if this is not available then display 'index.php'.

1. **Adding MIME Types**

MIME types set what a file is, or rather what file extensions refer to what file types. For example, a '.html' file extension refers to a HTML document, a '.zip' file extension refers to a ZIP archive file. The server needs to know this so it knows how to deal with the file. This is often used to create custom file extension for common file types.

To setup a MIME type, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | AddType text/html htm0 | |

'AddType' specifies that you are adding a MIME type. The second part is the MIME type, in this case text or HTML, and the final part is the file extension, in this example 'htm0'.

A common issue with MP3 or SWF files not playing can be resolved with the following text:

|  |  |
| --- | --- |
| |  | | --- | | AddType application/x-shockwave-flash swf | |

A handy trick, to force a file to be downloaded, via the 'Save As' feature in the web browser, set the MIME type to application/octet-stream and the browser will immediately prompt for download. Note, this does not work consistently in some versions of Microsoft Internet Explorer.

Here is a list of various MIME types and some associations:

|  |  |
| --- | --- |
| |  | | --- | | AddType text/html .html .htm  AddType text/plain .txt  AddType text/richtext .rtx  AddType text/tab-separated-values .tsv  AddType text/x-setext .etx  AddType text/x-server-parsed-html .shtml .sht  AddType application/macbinhex-40 .hqx  AddType application/netalivelink .nel  AddType application/netalive .net  AddType application/news-message-id  AddType application/news-transmission  AddType application/octet-stream .bin .exe  AddType application/oda .oda  AddType application/pdf .pdf  AddType application/postscript .ai .eps .ps  AddType application/remote-printing  AddType application/rtf .rtf  AddType application/slate  AddType application/zip .zip  AddType application/x-mif .mif  AddType application/wita  AddType application/wordperfect5.1  AddType application/x-csh .csh  AddType application/x-dvi .dvi  AddType application/x-hdf .hdf  AddType application/x-latex .latex  AddType application/x-netcdf .nc .cdf  AddType application/x-sh .sh  AddType application/x-tcl .tcl  AddType application/x-tex .tex  AddType application/x-texinfo .texinfo .texi  AddType application/x-troff .t .tr .roff  AddType application/x-troff-man .man  AddType application/x-troff-me .me  AddType application/x-troff-ms .ms  AddType application/x-wais-source .src  AddType application/x-bcpio .bcpio  AddType application/x-cpio .cpio  AddType application/x-gtar .gtar  AddType application/x-shar .shar  AddType application/x-sv4cpio .sv4cpio  AddType application/x-sv4crc .sv4crc  AddType application/x-tar .tar  AddType application/x-ustar .ustar  AddType application/x-director .dcr  AddType application/x-director .dir  AddType application/x-director .dxr  AddType application/x-onlive .sds  AddType application/x-httpd-cgi .cgi  AddType image/gif .gif .GIF  AddType image/ief .ief  AddType image/jpeg .jpeg .jpg .jpe .JPG  AddType image/tiff .tiff .tif  AddType image/x-cmu-raster .ras  AddType image/x-portable-anymap .pnm  AddType image/x-portable-bitmap .pbm  AddType image/x-portable-graymap .pgm  AddType image/x-portable-pixmap .ppm  AddType image/x-rgb .rgb  AddType image/x-xbitmap .xbm  AddType image/x-xpixmap .xpm  AddType image/x-xwindowdump .xwd  AddType audio/basic .au .snd  AddType audio/x-aiff .aif .aiff .aifc  AddType audio/x-wav .wav  AddType audio/x-pn-realaudio .ram  AddType audio/x-midi .mid  AddType video/mpeg .mpeg .mpg .mpe  AddType video/quicktime .qt .mov  AddType video/x-msvideo .avi  AddType video/x-sgi-movie .movie  AddType message/external-body  AddType message/news  AddType message/partial  AddType message/rfc822  AddType multipart/alternative  AddType multipart/appledouble  AddType multipart/digest  AddType multipart/mixed  AddType multipart/parallel  AddType x-world/x-vrml .wrl | |

1. **Enable SSI with .htaccess**

SSI stands for server side includes, these are special HTML tags which you can include in your HTML documents to call CGI scripts or other HTML content. This is particularly useful, for example to include a navigation menu in your HTML documents, it allows you to use one document to display the navigation menu in all your other documents. This saves disk space and means if you need to update the content, you only need to modify one file.

Two examples of HTML tags you would use to call SSI documents are shown below, these would be placed in your HTML document:

|  |  |
| --- | --- |
| |  | | --- | | <!--#exec cgi="/cgi-bin/script.cgi"--> | |

This would load the CGI script 'script.cgi' which is located in the 'cgi-bin' directory.

|  |  |
| --- | --- |
| |  | | --- | | <!--#include virtual="/files/document.html"--> | |

This example would call the HTML document 'document.html' which is located in the 'files' directory. It is important to use a relative URL, not a path or full URL.

It is likely SSI will work on your web server, but you will probably need to use '.shtml' file extensions rather than '.html'. This can be frustrating if you already have a web site setup which uses '.html' extensions. In this case, you can enable SSI by following the instructions below.

To enable SSI, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | AddHandler server-parsed .html | |

The above lines tell the Apache Web Server to allow server side includes in documents with the file extension '.html'.

To enable SSI for multiple file extensions, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | AddHandler server-parsed .html AddHandler server-parsed .shtml AddHandler server-parsed .htm | |

1. **Enable CGI outside of the CGI-bin**

If your web server does not allow you to run CGI scripts outside of the 'cgi-bin' directory, you can enable CGI. Check with your system administrator or web hosting company before doing so.

To enable CGI, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | AddHandler cgi-script .cgi Options +ExecCGI | |

The above lines tell the Apache Web Server to allow firstly, process '.cgi' files as CGI scripts, and secondly to enable CGI within the directory.

1. **Disable Directory Listings**

Preventing directory listings can be very useful if for example, you have a directory containing important '.zip' archive files or to prevent viewing of your image directories. Alternatively it can also be useful to enable directory listings if they are not available on your server, for example if you wish to display directory listings of your important '.zip' files.

To prevent directory listings, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | IndexIgnore \* | |

The above lines tell the Apache Web Server to prevent directory listings of directories and files within the directory containing the .htaccess file. The '\*' represents a wildcard, this means it will not display any files. It is possible to prevent listings of only certain file types, so for example you can show listings of '.html' files but not your '.zip' files.

To prevent listing '.zip' files, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | IndexIgnore \*.zip | |

The above line tells the Apache Web Server to list all files except those that end with '.zip'.

To prevent listing multiple file types, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | IndexIgnore \*.zip \*.jpg \*.gif | |

The above line tells the Apache Web Server to list all files except those that end with '.zip', '.jpg' or '.gif'.

Alternatively, if your server does not allow directory listings and you would like to enable them, create a .htaccess file following the main instructions and guidance which includes the following text:

|  |  |
| --- | --- |
| |  | | --- | | Options +Indexes | |

The above line tells the Apache Web Server to enable directory listing within the directory containing this .htaccess file. You can also reverse this to disable directory listings by replacing the plus sign before the text 'Indexes' with a minus sign. e.g. 'Options -Indexes'.

1. **Setting Server Time zone**

To set your web servers date timezone, for example for Eastern Standard Time (EST) use the following code:

|  |  |
| --- | --- |
| |  | | --- | | SetEnv TZ America/Indianapolis | |

For example, for Los Angeles time (Pacific time), use the following code:

|  |  |
| --- | --- |
| |  | | --- | | SetEnv TZ America/Los\_Angeles | |

Other location examples include:

|  |  |
| --- | --- |
| |  | | --- | | America/New\_York - Eastern Time  America/Detroit - Eastern Time - Michigan (most locations)  America/Louisville - Eastern Time (Louisville, Kentucky)  America/Indianapolis - Eastern Standard Time (Indiana, most locations)  America/Indiana/Marengo - Eastern Standard Time (Indiana, Crawford County)  America/Indiana/Knox - Eastern Standard Time (Indiana, Starke County)  America/Indiana/Vevay - Eastern Standard Time (Indiana, Switzerland County)  America/Chicago - Central Time  America/Menominee - Central Time (Michigan, Wisconsin border)  America/Denver - Mountain Time  America/Boise - Mountain Time (South Idaho, East Oregon)  America/Shiprock - Mountain Time (Navajo)  America/Phoenix - Mountain Standard Time (Arizona)  America/Los\_Angeles - Pacific Time  America/Anchorage - Alaska Time  America/Juneau - Alaska Time (Alaska panhandle)  America/Yakutat - Alaska Time (Alaska panhandle neck)  America/Nome - Alaska Time (west Alaska)  America/Adak - Aleutian Islands  Pacific/Honolulu - Hawaii | |

1. **Changing the Server Signature**

To change the server signature which is displayed as part of the default Apache error documents, use the following code:

|  |  |
| --- | --- |
| |  | | --- | | ServerSignature EMail SetEnv SERVER\_ADMIN nospace@pleasenospace.com | |

The example above will simply change the email address which is displayed, this is useful if the default address is not set correctly.

To remove the server signature completely, use the following code:

|  |  |
| --- | --- |
| |  | | --- | | ServerSignature Off | |

1. **Preventing Access to your PHP Includes Files**

If you have a directory containing PHP includes, that you do not wish to be accessed directly from the browser, there is a way of disabling the directory using Mod\_Rewrite.   
   
To enable this, create a .htaccess file following the main instructions and guidance, and include the following text: 

|  |  |
| --- | --- |
| |  | | --- | | ## Enable Mod Rewrite, this is only required once in each .htaccess file RewriteEngine On  RewriteBase /  ## Test for access to includes directory RewriteCond %{THE\_REQUEST} ^[A-Z]{3,9}\ /includes/ .\*$ [NC]  ## Test that file requested has php extension  RewriteCond %{REQUEST\_FILENAME} ^.+\.php$  ## Forbid Access  RewriteRule .\* - [F,NS,L] | |

1. **Prevent Access to PHP.ini**

If you run the risk of someone accessing your php.ini or php.cgi files directly through their browsers, you can limit access to them using .htaccess.   
   
To enable this, create a .htaccess file following the main instructions and guidance, and include the following text: 

|  |  |
| --- | --- |
| |  | | --- | | <FilesMatch "^php5?\.(ini|cgi)$"> Order Deny,Allow  Deny from All  Allow from env=REDIRECT\_STATUS  </FilesMatch> | |

1. **Forcing Scripts to display as source code**

If you need to display scripts as source code, instead of executing, for example to allow review, this can be achieved with the Remove Handler function   
   
To enable this, create a .htaccess file following the main instructions and guidance, and include the following text: 

|  |  |
| --- | --- |
| |  | | --- | | RemoveHandler cgi-script .pl .cgi .php .py  AddType text/plain .pl .cgi .php .py | |