- Partners
- Support
- Community
- Ubuntu.com

- Login to edit

Search

EnablingUseOfApacheHtaccessFiles

WARNING: This page was written based on Ubuntu 10.04 (Lucid), although the same may apply to other releases. You are encouraged to also read:

1. The related ubuntu server guide (Choose your distribution version > "Ubuntu Server Guide" > "Web servers").

2. The apache htaccess tutorial: http://httpd.apache.org/docs/2.0/howto/htaccess.html

By **default**, Ubuntu's Apache will **ignore** the directives in your `.htaccess` files.

When (not) to use .htaccess files

According to Apache.org's Apache Tutorial,

"In general, you should never use .htaccess files unless you don't have access to the main server configuration file. There is, for example, a prevailing misconception that user authentication should always be done in .htaccess files. This is simply not the case. You can put user authentication configurations in the main server configuration, and this is, in fact, the preferred way to do things."

".htaccess files should be used in a case where the content providers need to make configuration changes to the server on a per-directory basis, but do not have root access on the server system. In the event that the server administrator is not willing to make frequent configuration changes, it might be desirable to permit individual users to make these changes in .htaccess files for themselves."

On Ed/X/Ubuntu 6.06 and Ubuntu Edgy Eft, the *"main server configuration file"* is `/etc/apache2/apache2.conf` .

OK, I know it is not recommended -- how do I do it

anyway?

To make `.htaccess` files work as expected, you need to edit this file:

```
/etc/apache2/sites-available/default
```

Look for a section that looks like this:

```
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
                # Uncomment this directive is you want to see apache2's
                # default start page (in /apache2-default) when you go to /
                #RedirectMatch ^/$ /apache2-default/
        </Directory>
```

You need to modify the line containing `AllowOverride None` to read `AllowOverride All`. This tells Apache that it's okay to allow `.htaccess` files to over-ride previous directives. You must *reload* Apache before this change will have an effect:

```
sudo /etc/init.d/apache2 reload
```

2009.12.08 note: in the LAMP download about a week ago with Ubuntu 9.10 (Karmic) the default configuration file was `/etc/apache2/sites-available/000-default` and it included AllowOverride None under `<Directory />` in addition to `<Directory /var/www/>`. Also, directories in `/www/var/` containing `.htaccess` files defaulted to not giving the Apache server read access, resulting in the Apache error

> *(13)Permission denied: /var/www/webapp/.htaccess pcfg_openfile: unable to check htaccess file, ensure it is readable.*

To fix, `$ sudo nautilus` then right click on the directory with the .htaccess file, select Properties, then select Permissions, and give the user group you log in as at least read permission.

See http://httpd.apache.org/docs/2.0/mod/core.html#allowoverride for more info on `AllowOverride`.

Password-Protect a Directory With .htaccess

Warning: On at least some versions of Ubuntu, `.htaccess` files will not work by default. See EnablingUseOfApacheHtaccessFiles for help on enabling them.

Create a file called `.htaccess` in the directory you want to password-protect with the follwing content:

```
AuthUserFile /your/path/.htpasswd
AuthName "Authorization Required"
AuthType Basic
require valid-user
```

instead of `valid-user`, you can also add the users you want directly

If you want to password protect just a single file in a folder add the following lines to the `.htaccess` file:

```
<Files "mypage.html">
  Require valid-user
</Files>
```

Then create the file `/your/path/.htpasswd` which contains the users that are allowed to login and their passwords. We do that with the `htpasswd` command:

```
htpasswd -c /path/to/your/.htpasswd user1
```

The `-c` flag is used only when you are creating a new file. After the first time, you will omit the `-c` flag, when you are adding new users to an already-existing password file. Otherwise you will overwrite the file!!

Nevertheless, you should store the file in as secure a location as possible, with whatever minimum permissions on the file so that the web server itself can read the file.

Finally we need to add the following lines to `/etc/apache2/apache2.conf`:

```
<Directory /your/path>
AllowOverride All
</Directory>
```

You have to adjust `/your/path/.htpasswd`

Restart your webserver:

```
sudo /etc/init.d/apache2 restart
```

Troubleshooting

If you can't access your stuff and the dialog keeps popping up, check that you entered the username and password correctly. If it still doesn't work, check the path to your `.htpasswd` and make sure the path specified in the `AuthUserFile directive` is correct. Also make sure that both the `.htpasswd` and `.htaccess` files are readable by the web server user `chmod 644` should do the trick!

Example

Here is an example on how to prevent users from access the directory, password-protect a specific file and allow userse to view a specific file:

```
AuthUserFile /your/path/.htpasswd
AuthName "Authorization Required"
AuthType Basic
Order Allow,Deny
<Files myfile1.html>
 Order Allow,Deny
 require valid-user
</Files>

<Files myfile2.html>
 Order Deny,Allow
</Files>
```

Redirect requests using .htaccess and mod_rewrite

1. Make sure Apache .htaccess is enabled (by default it is enabled in Ubuntu)

2. Make sure the Apache module `mod_rewrite` is enabled. Execute:

```
sudo a2enmod rewrite
```

..and see if rewrite is listed here:

```
sudo apache2ctl -M
```

and then you can redirect requests using RewriteRules. Example:

```
RewriteEngine On

RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ index.php?_REQUEST=$1 [L]
```

EnablingUseOfApacheHtaccessFiles (last edited 2012-05-04 18:54:07 by medigeek @ 156.247.106.109.adsl.dyn.beotel.net[109.106.247.156]:medigeek)