Problem redirecting 403 Forbidden to 404 Not Found

The pertinent part of my .htaccess looks like this:

```
Options -Indexes
<FilesMatch include>
    Order allow,deny
    Deny from all
</FilesMatch>
RedirectMatch 404 ^/include(/.*)$
```

And it's generating the following responses:

- /include 403

- /include/ 404

- /include/config.inc 403

I can tell by looking at my pattern that problem is likely in the (/.*) part but everything I have tried gives me the same results; instead of consistently getting 404 I get a 404 for the one case and 403 for everything else. **What is wrong with the expression I'm using? Alternatively since I have to do this for a few directories is there a blanket approach that would allow me to convert all 403 responses to 404?**

UPDATE: I've found that by removing the FileMatch I get better results, so my .htaccess now looks like this:

```
Options -Indexes
RedirectMatch 404 ^/include(/.*)?$ # Added dlamblin's first suggestion
```

And generates the following responses:

- /include 404

- /include/ 404

- /include/config.inc 403

UPDATE: Interestingly enough I have discovered that the following produces different output:

```
RedirectMatch 404 ^/include(/?|/.*)$
RedirectMatch 404 ^/template(/?|/.*)$
```

The template pattern works on all cases however include is still generating 403 for all files in include (e.g. /include/config.inc) **Could this be an issue with the directory name and not a problem with the .htaccess file itself?**

UPDATE: The following in my .htaccess was conflicting with redirect when accessing /include/config.inc.

```
<FilesMatch config>
    Order allow,deny
    Deny from all
</FilesMatch>
```

apache   .htaccess   mod-rewrite   http-status-code-404   http-status-code-403

2   It's a long time ago now, but I'd like to suggest that you could resolve all of these issues by just putting your includes and templates somewhere outside your document root; it's the hardest way to screw up and it has slightly less load on Apache to boot! – El Yobo Jul 30 '10 at 1:48

Quickest and simplest solution that only needs `ErrorDocument` directive and a PHP page for the 404 error: Just use `http_response_code` . No need to mess with `RedirectMatch` , `RewriteEngine` , etc. – ADTC Mar 27 '16 at 8:00

7 Answers

I can understand why the /include isn't caught by your RedirectMatch, you aren't making the end '/' optional, however the /include/config.inc part is a bit on the puzzling side.

Here is what I got to work on Apache 2.2:

```
<FilesMatch /include(/?|/.*)>
    Order allow,deny
    Deny from all
```

```
</FilesMatch>
```

```
RedirectMatch 404 ^/include(/?|/.*)$
```

This handles these cases:

```
/include 404
/include/ 404
/include/config.inc 404
```

I had to change the FilesMatch part in order for the /include part to work properly.

EDIT:

The match line also works without the <FilesMatch> section in .htaccess and gives the expected results.

edited Feb 14 '09 at 0:43   answered Feb 14 '09 at 0:23

Bryan Rehbein
6,892 3 29 41

There has to be something else causing the problem because using this /include/config.inc still generates a 403. I'm going to do some digging in my httpd.conf and see if something there is interfering. — Kevin Loney Feb 14 '09 at 1:03

What would be wrong about `RewriteRule ^/include(/?|/.*)$ - [R=404,NC,L]` ? — Frank Nocke Sep 6 '15 at 6:23

Another possibility is not to bother matching the whole path:

```
RedirectMatch 404 ^/include
```

If there are publicly visible URL paths that might start with "/include" (say, "/includeMe"), a small addition will separate the private from the public URLs:

```
RedirectMatch 404 ^/include(/|$)
```

answered Jul 8 '10 at 22:30   community wiki
outis

With rewrite mod:

```
RewriteEngine on
```

```
RewriteCond %{THE_REQUEST} ^.*/\.
RewriteRule ^(.*)$ - [R=404]
```

Every file or dir who begin with a dot will be redirected to 404.

```
/myDir/.svn => 404
/.gitignore => 404
/dir1/dir2_dir3/
```

Or to change all 403,400 errors into 404 errors, put this at the end of **/etc/apache2/conf.d/localized-error-pages** OR into a **.htaccess**

```
# Will raise a 404 error, because the file <fake_file_for_apache_404.php> doesn't exist.
# We change 403 or 400 to 404 !
ErrorDocument 400 /fake_file_for_apache_404.php
ErrorDocument 403 /fake_file_for_apache_404.php
# We need to rewrite 404 error, else we will have "fake_file_for_apache_404.php not found"
ErrorDocument 404 "<!DOCTYPE HTML PUBLIC \"-//IETF//DTD HTML 2.0//EN\"><html><head>
<title>404 Not Found</title></head><body><h1>Not Found</h1><p>The requested URL <script
type=\"text/javascript\">document.write(document.location.pathname);</script> was not
found on this server.</p></body></html>"
ErrorDocument 500 "Server in update. Please comme back later."
```

edited Dec 17 '13 at 19:17   community wiki
2 revs
ImmortalPC

Don't you want '^/include(/.*)?$'

This part is a guess, but what would happen if you put the RedirectMatch above the block. That way you wouldn't by denying (forbidding) access to a request before you redirect that request to 404.

Good point, unfortunately I get the same results. – Kevin Loney Feb 14 '09 at 0:03

A better and simple way is as follow :

1 - Insert in your main conf (ie Outside VHOST world) :

```
<IfModule mod_alias.c>
RedirectMatch 404 ^/myDirectory(?|\/)$
</IfModule>
```

with myDirectory = css, js, images ...

2 - You can set a per directory -Indexes as follow :

Allow your "DirectoryIndex" Apache directive content to be served :

```
<Directory "/myPathtoMyWebDocRoot/">
    Options Indexes
    AllowOverride None
    Order Allow, Deny
    Allow from all
</Directory>
```

and Deny directory index for others :

```
<Directory "/myPathtoMyWebDocRoot/myDirectory/">
    Options -Indexes
    AllowOverride None
    Order Allow, Deny
    Allow from all
</Directory>
```

with myDirectory = *, css, images, js, secret, ... following your needs.

edited Oct 9 '13 at 9:44

community wiki
2 revs
user2862072

This one works as expected for all files/dirs with name started by dot even in subdirectories:

```
<FilesMatch "^\..+">
    Order allow,deny
    Deny from all
    Satisfy All
</FilesMatch>
RedirectMatch 404 \/\..+$
```

answered May 15 '12 at 22:19

community wiki
kbec

In that case you do not need to use `Options -Indexes` My solution to stop displaying directory's content as list and display 404 error is simple. Create .htaccess file in root directory of your project and write which directories should be protected.

Directories structure

```
- your_root_directory
  - .htaccess
  - 404.html
  - index.html
  - app
    - other files I do not want to show
  - models
    - other files I do not want to show
```

.htaccess

```
RewriteEngine On
RewriteRule   ^app/ - [R=404,L]
RewriteRule   ^models/ - [R=404,L]
```

ErrorDocument 404 /your_root_directory/404.html The second line of .htaccess disallow access to list items in app directory and all its subridectories.

The third line of .htaccess disallow access to list items in models directory and all its subridectories.

The fourth of .htaccess line sets our own 404 error (if you do not want to show apache default error).

Remember to clear cache on your browser when you work with htaccess.