

Table Of Contents

Table Of Contents 1

Step # 1: Make sure Apache is configured to use .htaccess file 2

Step # 2: Create a password file with htpasswd 2

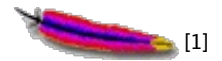
Step # 3: Test your configuration 3

Troubleshooting 4

See also: 4

[Home](#) > [Faq](#) > [Apache](#)

Set Apache Password Protected Directories With .htaccess File

Posted by [Vivek Gite](#) <vivek@nixcraft.com>

Q. How do I protecting a directory in Apache on linux?

A. There are many ways you can password protect directories under Apache web server. This is important to keep your file privates from both unauthorized users and search engines (when you do not want to get your data indexed). Here you will see the basics of password protecting a directory on your server. You can use any one of the following method:

1. Putting authentication directives in a <Directory> section, in your main server configuration httpd.conf file, is the preferred way to implement this kind of authentication.
2. If you do not have access to Apache httpd.conf file (for example shared hosting) then with the help of file called .htaccess you can create password protect directories. .htaccess file provide a way to make configuration changes on a per-directory basis.

In order to create apache password protected directories you need:

- Password file
- And Directory name which you would like to password protect (/var/www/docs)

Step # 1: Make sure Apache is configured to use .htaccess file

You need to have AllowOverride AuthConfig directive in httpd.conf file in order for these directives to have any effect. Look for DocumentRoot Directory entry. In this example, our DocumentRoot directory is set to /var/www. Therefore, my entry in httpd.conf looks like as follows:

```
<Directory /var/www>
Options Indexes Includes FollowSymLinks MultiViews
AllowOverride AuthConfig
Order allow,deny
Allow from all
</Directory>
```

Save the file and restart Apache

If you are using Red Hat/Fedora Linux:

```
# service httpd restart
```

If you are using Debian Linux:

```
# /etc/init.d/apache-perl restart
```

Step # 2: Create a password file with htpasswd

htpasswd command is used to create and update the flat-files (text file) used to store usernames and password for basic authentication of Apache users. General syntax:
htpasswd -c password-file username

Where,

- -c : Create the password-file. If password-file already exists, it is rewritten and truncated.
- username : The username to create or update in password-file. If username does not exist in this file, an entry is added. If it does exist, the password is changed.

Create directory outside apache document root, so that only Apache can access password file. The password-file should be placed somewhere not accessible from the web. This is so that people cannot download the password file:

```
# mkdir -p /home/secure/
```

Add new user called vivek

```
# htpasswd -c /home/secure/apasswords vivek
```

Make sure /home/secure/apasswords file is readable by Apache web server. If Apache cannot read your password file, it will not authenticate you. You need to setup a correct permission using chown command. Usually apache use www-data user. Use the following command to find out Apache username. If you are using Debian Linux use apache2.conf, type the following command:

```
# grep -e '^User' /etc/apache2/apache2.conf
```

Output:

```
www-data
```

Now allow apache user www-data to read our password file:

```
# chown www-data:www-data /home/secure/apasswords
# chmod 0660 /home/secure/apasswords
```

If you are using RedHat and Fedora core, type the following commands :

```
# grep -e '^User' /etc/httpd/conf/httpd.conf
```

Output:

```
apache
```

Now allow apache user apache to read our password file:

```
# chown apache:apache /home/secure/apasswords
# chmod 0660 /home/secure/apasswords
```

Now our user vivek is added but you need to configure the Apache web server to request a password and tell the server which users are allowed access. Let us assume you have directory called /var/www/docs and you would like to protect it with a password.

Create a directory /var/www/docs if it does not exist:

```
# mkdir -p /var/www/docs
```

Create .htaccess file using text editor:

```
# cd /var/www/docs
# vi .htaccess
```

Add following text:

```
AuthType Basic
AuthName "Restricted Access"
AuthUserFile /home/secure/apasswords
Require user vivek
```

Save file and exit to shell prompt.

Step # 3: Test your configuration

Fire your browser type url <http://yourdomain.com/docs/> or <http://localhost/docs/> or http://ip-address/docs



[2]

When prompted for username and password please supply username vivek and password. You can add following lines to any file <Directory> entry in httpd.conf file:

```
AuthType Basic
AuthName "Restricted Access"
AuthUserFile /home/secure/apasswords
Require user vivek
```

To change or setup new user use htpasswd command again.

Troubleshooting

If password is not accepted or if you want to troubleshoot authentication related problems, open and see apache access.log/error.log files:

Fedora Core/CentOS/RHEL Linux log file location:

```
# tail -f /var/log/httpd/access_log
# tail -f /var/log/httpd/error_log
```

Debian Linux Apache 2 log file location:

```
# tailf -f /var/log/apache2/access.log
# tailf -f /var/log/apache2/error.log
```

See also:

- [Apache Tutorial: .htaccess files](#) [3]

4000+ howtos and counting! Want to read more Linux / UNIX howtos, tips and tricks? Subscribe to our [daily email](#) newsletter or [weekly newsletter](#) to make sure you don't miss a single tip/tricks. Alternatively, subscribe via [RSS/XML](#) feed.

Article printed from Frequently Asked Questions About Linux / UNIX: <http://www.cyberciti.biz/faq/>

URL to article: <http://www.cyberciti.biz/faq/howto-setup-apache-password-protect-directory-with-htaccess-file/>

URLs in this post:

[1] Image: <http://www.cyberciti.biz/faq/faq/category/apache/>

[2] Image: http://www.cyberciti.biz/faqs/uploaded_images/apache-password-protected-dir-782077.png

[3] Apache Tutorial: .htaccess files: <http://httpd.apache.org/docs/2.2/howto/htaccess.html>

