

## Password protecting web pages in Apache

There are four steps (as root or su):

1. Create the file **.htaccess** in the directory you wish to protect.
2. Edit the configuration file **httpd.conf**.
3. Run the utility **htpasswd** to establish a username and password for the directory.
4. Stop and restart Apache.

### **.htaccess**

Move to the directory you wish to protect. Create, with vi or other text editor, a file called .htaccess (don't forget the period) and enter (or paste and edit) the following:

```
AuthUserFile /full/path/to/protected/directory/.htpasswd
AuthGroupFile /dev/null
AuthType Basic
AuthName "Name of Realm"
```

```
<Limit GET POST>
require valid-user
</Limit>
```

### ***Explanation:***

The first line, AuthUserFile, tells apache where to look for the username and password: specifically, in a file called .htpasswd, located (by convention) in the directory with .htaccess. The utility, **htpasswd**, will create the file .htpassword for you, when you run it to choose a username and password.

The fourth line, AuthName, will appear in the password pop up box, letting the user know which username and password to enter. It can be any name. Perhaps "Astronomy 301". The 'realm', can protect disparately located directories for a single group of users.

The <Limit> directive tells apache to require authentication and must be switched on in the main Apache configuration file, httpd.conf.

### **httpd.conf**

Now move to Apache's configuration directory, conf, usually installed under Apache's main directory, arranged perhaps like:

```
/opt/apache-2.0.40/conf
```

In vi, or other text editor, open httpd.conf.

Perhaps a hundred lines or more down is a directive, `AllowOverride`, which tells apache which directives to honor in `.htaccess` files. In the uncommented line (without the `#`) ensure that `Limit` is included. If the line reads `AllowOverride All`, `Limit` is included in `All` (see the obvious instructions within `httpd.conf`). Otherwise, add it.

Save and close `httpd.conf`.

### **htpasswd**

Now move out of the `conf` directory to Apache's `bin` directory. Consistent with the above example, it would be:

```
/opt/apache-2.0.40/bin
```

Run the utility **htpasswd** with the command:

```
htpasswd -c /full/path/to/protected/directory/.htpasswd username
```

#### ***Explanation:***

The `-c` switch will create the `.htpasswd` file if it does not exist, and overwrite it if it exists, unless it cannot be read or written to (check permissions if error).

Include the full path to the `.htpasswd` file.

The **htpasswd** utility will prompt for a password, and encrypt it (by default) for whatever user is designated by *username*.

### **Stopping and starting Apache**

Use the utility **apachectl** to stop and start Apache:

```
apachectl stop  
apachectl start
```

#### ***Explanation:***

**apachectl** is often located in the `bin` directory with **htpasswd** (as above, `/opt/apache-2.0.40/bin`). It may run only with the full pathname.

*ex.*  
`/opt/apache-2.0.40/bin/apachectl stop`  
`/opt/apache-2.0.40/bin/apachectl start`