

Password-Protecting Your Site

NOTE: The following section is excerpted from the *ServInt Webhosting User Manual*. Please see the manual for the full version of this and many other articles covering essential topics for webmasters.

Security on the Internet is a big concern, and one of the issues most important to you, as webmasters, is the ability to restrict access to portions of your web content. Luckily, it's very easy to add password-protection and other access control features to your site. **.htaccess** is a program that restricts access to certain areas of your site by requiring a login name and password to be used for entrance to those areas. **.htaccess** is the most common form of user verification used on the Internet. D.C. FreeNet includes a basic **.htaccess** script that is placed in the directory of your choice. The **.htaccess** script depends on a file called **.htpasswd**, where the allowed users have their login name and password stored. The password is stored encrypted and cannot be identified by looking at this file.

When a person tries to gain access to a site with the **.htaccess** script, you'll see a dialog box pop up, asking for a **Login Name** and **Password**. The server will then verify this login name and password by checking it against the **.htpasswd** file. If the user's input matches the **.htpasswd** information, access is granted. If not, an HTML page stating `Access ID Restricted` appears. This then puts the user back to the original page and asks for the Login Name and Password again. Access is not permitted until a valid Login Name and Password are entered.

Password-Secured Areas

The most common use for **.htaccess** is to protect directories from unauthorized access. To set up a protected area on your site you will need three files: **.htpasswd**, **.htaccess**, and **addpw**.

The **.htpasswd** file is a file that will store you users names and passwords.

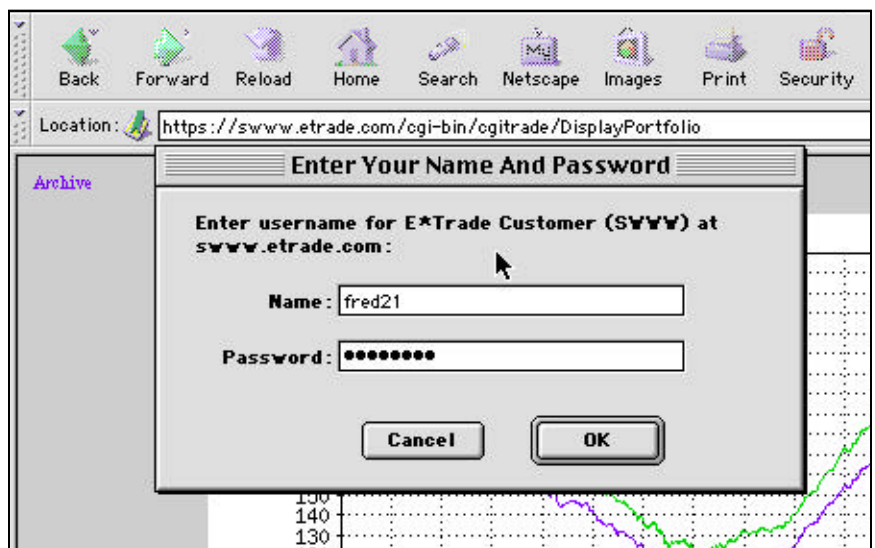
The **.htaccess** file is a file that operates the protection script. This file looks like this:

```
AuthName
AuthType Basic
AuthUserFile /usr/home/yourusername/.htpasswd
<Limit GET>
order deny,allow
deny from none
allow from all
</Limit>
```

The **addpw** file is the file you will use to add the names and passwords to the **.htpasswd** file.

These three files need to be placed into your directory. If you do not know how to install these files, please E-mail D.C. FreeNet Tech Support (help@dcfree.net), and we will be happy to set up your account with the files. Once you have the files, you will need to place the **.htaccess** file in the directory that you would like to password-protect. This will protect any files within that directory. If there are sub-directories in the directory with the **.htaccess** file, then those directories will be password protected as well.

You can easily modify the **.htaccess** file with a text editor to match the protection you want to



An **htaccess**-protected directory requires a username and password.

give. For example, we can use **pico** (see pico in the *ServInt User Manual*, page 87) to modify the file like this:

```
server: % pico -w .htaccess
```

Add the following to the file:

```
AuthName Private Area
AuthType Basic
AuthDBMUserFile /usr/home/yourusername/.htpasswd
require valid-user
```

This will allow only people who know a username and password from the `.htpasswd.db` file to access the files in the directory or sub-directories that the `.htaccess` file is located. The `.htpasswd.db` file is created using **dbmmanage** (an easy-to-use Unix database program).

The `.htpasswd` file is where you will store the names and passwords for individuals who will have access to the password protected files. This file can be read, but the names and passwords will be stored encrypted.

We have also included a script called `addpw` that will allow you to add people to your password file. This command uses information that you plug in to encrypt the password and add it to the `.htpasswd` file. To use the `addpw` command you will need to telnet to the D.C. FreeNet server, log in, and be in your home directory (`/usr/home/yourusername`). Once there, type:

```
server: % addpw .htpasswd (user-name) (user-password)
```

This command will add a user and password to the `.htpasswd` file. You select the user name and the user password. No two people may have either the same user name or user password. You can create one generic user name and password that you can give out to many people. For example:

```
server: % addpw .htpasswd adduser test a1b2c3
```

The protected directory can now be accessed by the user `test` with the password `a1b2c3`.

Extending htaccess

Error Redirection

Another thing you can do is redirect the browser to a URL when an error occurs. The most common use is redirecting when a document fails to load (404 Error). You would need something like the following in your `.htaccess` file (using a text editor, as shown above:

```
ErrorDocument 500 http://foo.example.com/cgi-bin/tester
ErrorDocument 404 /cgi-bin/bad_urls.pl
ErrorDocument 401 /subscription_info.html
ErrorDocument 403 "Sorry, can't allow you access today"
```

Learning More

For additional information on `.htaccess` and how to use it to your advantage, please read the Apache Directives section of the Apache Web Site at <http://www.apache.org/docs/directives.html>.