Apache Module mod_userdir

Description:	User-specific directories
Status:	Base
Module Identifier:	userdir_module
Source File:	mod_userdir.c

Summary

This module allows user-specific directories to be accessed using the `http://example.com/~user/` syntax.

Topics

Directives

See also

- Mapping URLs to the Filesystem[1]
- public_html tutorial[2]

UserDir Directive

Description:	Location of the user-specific directories
Syntax:	`UserDir` *`directory-filename`*
Default:	`UserDir public_html`
Context:	server config, virtual host
Status:	Base
Module:	mod_userdir

The `UserDir` directive sets the real directory in a user's home directory to use when a request for a document for a user is received. *Directory-filename* is one of the following:

- The name of a directory or a pattern such as those shown below.
- The keyword `disabled`. This turns off *all* username-to-directory translations except those explicitly named with the `enabled` keyword (see below).
- The keyword `disabled` followed by a space-delimited list of usernames. Usernames that appear in such a list will *never* have directory translation performed, even if they appear in an `enabled` clause.
- The keyword `enabled` followed by a space-delimited list of usernames. These usernames will have directory translation performed even if a global disable is in effect, but not if they also appear in a `disabled` clause.

If neither the `enabled` nor the `disabled` keywords appear in the `Userdir` directive, the argument is treated as a filename pattern, and is used to turn the name into a directory specification. A request for `http://www.foo.com/~bob/one/two.html` will be translated to:

UserDir directive used	**Translated path**
UserDir public_html	~bob/public_html/one/two.html

UserDir directive used	**Translated path**
UserDir /usr/web	/usr/web/bob/one/two.html
UserDir /home/*/www	/home/bob/www/one/two.html

The following directives will send redirects to the client:

UserDir directive used	**Translated path**
UserDir http://www.foo.com/users	http://www.foo.com/users/bob/one/two.html
UserDir http://www.foo.com/*/usr	http://www.foo.com/bob/usr/one/two.html
UserDir http://www.foo.com/~*/	http://www.foo.com/~bob/one/two.html

Be careful when using this directive; for instance, "`UserDir ./`" would map "`/~root`" to "`/`" - which is probably undesirable. It is strongly recommended that your configuration include a "`UserDir disabled root`" declaration. See also the `Directory` directive and the Security Tips[3] page for more information.

Additional examples:

To allow a few users to have `UserDir` directories, but not anyone else, use the following:

```
UserDir disabled
UserDir enabled user1 user2 user3
```

To allow most users to have `UserDir` directories, but deny this to a few, use the following:

```
UserDir enabled
UserDir disabled user4 user5 user6
```

It is also possible to specify alternative user directories. If you use a command like:

```
Userdir public_html /usr/web http://www.foo.com/
```

With a request for http://www.foo.com/~bob/one/two.html, will try to find the page at ~bob/public_html/one/two.html first, then /usr/web/bob/one/two.html, and finally it will send a redirect to http://www.foo.com/bob/one/two.html.

If you add a redirect, it must be the last alternative in the list. Apache cannot determine if the redirect succeeded or not, so if you have the redirect earlier in the list, that will always be the alternative that is used.

See also

- public_html tutorial[2]

URI References

[1] http://httpd.apache.org/docs-2.1/urlmapping.html
[2] http://httpd.apache.org/docs-2.1/howto/public_html.html
[3] http://httpd.apache.org/docs-2.1/misc/security_tips.html