Per-user web directories

On systems with multiple users, each user can be permitted to have a web site in their home directory using the `UserDir` directive. Visitors to a URL `http://example.com/~username/` will get content out of the home directory of the user "`username`", out of the subdirectory specified by the `UserDir` directive.

Note that, by default, access to these directories is **not** enabled. You can enable access when using `UserDir` by uncommenting the line:

```
#Include conf/extra/httpd-userdir.conf
```

in the default config file `conf/httpd.conf`, and adapting the `httpd-userdir.conf` file as necessary, or by including the appropriate directives in a `<Directory>` block within the main config file.

▲ Per-user web directories

Related Modules	Related Directives
mod_userdir	UserDir
	DirectoryMatch
	AllowOverride

▲ Setting the file path with UserDir

The `UserDir` directive specifies a directory out of which per-user content is loaded. This directive may take several different forms.

If a path is given which does not start with a leading slash, it is assumed to be a directory path relative to the home directory of the specified user. Given this configuration:

```
UserDir public_html
```

the URL `http://example.com/~rbowen/file.html` will be translated to the file path `/home/rbowen/public_html/file.html`

If a path is given starting with a slash, a directory path will be constructed using that path, plus the username specified. Given this configuration:

```
UserDir /var/html
```

the URL `http://example.com/~rbowen/file.html` will be translated to the file path `/var/html/rbowen/file.html`

If a path is provided which contains an asterisk (*), a path is used in which the asterisk is replaced with the username. Given this configuration:

```
UserDir /var/www/*/docs
```

the URL `http://example.com/~rbowen/file.html` will be translated to the file path `/var/www/rbowen/docs/file.html`

Multiple directories or directory paths can also be set.

```
UserDir public_html /var/html
```

For the URL `http://example.com/~rbowen/file.html`, Apache will search for `~rbowen`. If it isn't found, Apache will search for `rbowen` in `/var/html`. If found, the above URL will then be translated to the file path `/var/html/rbowen/file.html`

Redirecting to external URLs

The `UserDir` directive can be used to redirect user directory requests to external URLs.

```
UserDir http://example.org/users/*/
```

The above example will redirect a request for `http://example.com/~bob/abc.html` to `http://example.org/users/bob/abc.html`.

Restricting what users are permitted to use this feature

Using the syntax shown in the UserDir documentation, you can restrict what users are permitted to use this functionality:

```
UserDir disabled root jro fish
```

The configuration above will enable the feature for all users except for those listed in the `disabled` statement. You can, likewise, disable the feature for all but a few users by using a configuration like the following:

```
UserDir disabled
UserDir enabled rbowen krietz
```

See `UserDir` documentation for additional examples.

Enabling a cgi directory for each user

In order to give each user their own cgi-bin directory, you can use a `<Directory>` directive to make a particular subdirectory of a user's home directory cgi-enabled.

```
<Directory "/home/*/public_html/cgi-bin/">
    Options ExecCGI
    SetHandler cgi-script
</Directory>
```

Then, presuming that `UserDir` is set to `public_html`, a cgi program `example.cgi` could be loaded from that directory as:

```
http://example.com/~rbowen/cgi-bin/example.cgi
```

Allowing users to alter configuration

If you want to allows users to modify the server configuration in their web space, they will need to use `.htaccess` files to make these changes. Ensure that you have set `AllowOverride` to a value sufficient for the directives that you want to permit the users to modify. See the .htaccess tutorial for additional details on how this works.