

Your web server cluster is almost ready to deploy. The only thing left is to take a few gotchas into account.

Module Gotchas

When creating modules, watch out for these gotchas:

- File paths
- Inline blocks

File Paths

In [Chapter 3](#), you moved the User Data script for the web server cluster into an external file, `user-data.sh`, and used the `templatefile` built-in function to read this file from disk. The catch with the `templatefile` function is that the filepath you use must be a relative path (you don't want to use absolute file paths, as your Terraform code may run on many different computers, each with a different disk layout)—but what is it relative to?

By default, Terraform interprets the path relative to the current working directory. That works if you're using the `templatefile` function in a Terraform configuration file that's in the same directory as where you're running `terraform apply` (that is, if you're using the `templatefile` function in the root module), but that won't work when you're using `templatefile` in a module that's defined in a separate folder (a reusable module).

To solve this issue, you can use an expression known as a *path reference*, which is of the form `path.<TYPE>`. Terraform supports the following types of path references:

`path.module`

Returns the filesystem path of the module where the expression is defined.