

NativeSPFastCGIConfig

FastCGI SP Application

It consists of 2 modules:

- a FastCGI responder (shibresponder) that handles the HandlerURL
- a FastCGI authorizer (shibauthorizer) that acts as a filter and does the usual (authN, export assertions and authZ).

Bellow is a sample lighttpd snip of the configuration needed to get this to work:

```
server.name = "your_server_name"

server.document-root = "/servers/tags/www/"

fastcgi.server = (
    "/Shibboleth.sso" => (("socket" => "/tmp/fcgi-resp.sock", "bin-path" => "/servers/sapo-sp/lib/shibboleth
/shibresponder", "check-local" => "disable", "mode" => "responder")),
    "/" => (("socket" => "/tmp/fcgi-auth.sock", "bin-path" => "/servers/sapo-sp/lib/shibboleth
/shibauthorizer", "check-local" => "disable", "mode" => "authorizer")),
    ".php" => (( "socket" => "/tmp/fcgi-php.sock", "broken-scriptfilename" => "enable", "mode" =>
"responder"))
)
```

I have a PHP application being served behind shibboleth. Also, in the case of lighttpd, you need the [patch for bug #322](#).

This opens up some interesting possibilities as it can be used with any web server that supports the fastcgi specification and can be run remotely. I haven't yet thought much about these issues but am open to suggestions.

A standard install of Nginx has FastCGI responder support, but not the "authorizer" mode needed for the filter. Until this support exists Nginx cannot be used in this way.

It is worth noting that with several customisations to Nginx, it is possible to leverage the Shibboleth applications above to achieve authentication with a backend application - for more information see [Integrating Nginx and a Shibboleth SP with FastCGI](#).