Jetty and PHP

> **"Warning!"**
>
> These instructions are for using PHP with Jetty 5.x only.

Jetty and PHP

Running PHP applications using the Jetty CGI servlet.
TODO: Find a FASTCGI servlet which by all accounts will have a significant impact on throughput and performance

Jetty as a general web server

You have Jetty configured to serve static files from the root context, and you want *.php files to be processed by PHP. Add the CGI servlet into the context as shown below.

*TODO: Find a way to configure jetty so that *.php/some/path/info is passed to PHP with SCRIPT_NAME=*.php and PATH_INFO=/some/path/info*

```
<!-- Set up the root context -->
<Call name="addContext">
   <Arg>/</Arg>
   <Set
name="ResourceBase">path/to/html/Or/PHP/file
s</Set>
   <!-- add miscellaneous handlers eg
HtAccessHandler--->
   <Call name="addServlet">
      <Arg>Common Gateway Interface</Arg>
      <Arg>*.php</Arg>
      <Arg>org.mortbay.servlet.CGI</Arg>
      <Put
name="commandPrefix">/usr/local/bin/php-cgi-
fix</Put>
      <!-- Possibly required for Windows
installs
      <Put
name="ENV_SystemRoot">C:\WINDOWS</Put>
      <Put
name="ENV_REDIRECT_STATUS">200</Put>
      -->
   </Call>
   <Call name="addWelcomeFile">
      <Arg>index.php</Arg>
   </Call>
   <!-- add more handlers, resource handler,
not found etc... -->
</Call>
```

PHP as a webapp

Your php app needs special environment variables for configuration or uses PATH_INFO information.

For example, yourapp.php/path/info expects yourapp.php to be processed by PHP with `PATH_INFO=/path/info`

Assuming jetty is configured to auto-discover web applications in a `webapps` directory and your php application files are in a directory `yourapp`.

Create a directory `webapps/yourapp/WEB-INF` and the standard `webapps/yourapp/WEB-INF/web.xml`

```
<code xml>
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app
  xmlns="http://java.sun.com/xml/ns/j2ee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"

xsi:schemaLocation="http://java.sun.com/xml/
ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.
xsd"
  version="2.4"
>

 <display-name>YourApp</display-name>

 <servlet>
   <servlet-name>PHP</servlet-name>

<servlet-class>org.mortbay.servlet.CGI</serv
let-class>
   <init-param>
```

```
<param-name>commandPrefix</param-name>

<param-value>/usr/local/bin/php-cgi-fix</par
am-value>
   </init-param>
   <init-param>

<param-name>ENV_yourRequiredEnvironmentVaria
ble</param-name>
        <param-value>yourValue</param-value>
   </init-param>
   <!- Path, other ENV_variables including
ENV_SystemRoot, ENV_REDIRECT_STATUS on
Windows -->
 </servlet>

 <servlet-mapping>
   <servlet-name>PHP</servlet-name>
   <url-pattern>/yourapp.php/*</url-pattern>
   <!-- Any other URL patterns that are
needed by your app to be processed by PHP
-->
 </servlet-mapping>

 <!-- If you want http://yourhost/yourapp to
call yourapp/yourapp.php then make a welcome
file -->
 <welcome-file-list>
```

```
    <welcome-file>yourapp.php</welcome-file>
  </welcome-file-list>
</web-app>
```

mod_rewrite

If your app makes use of Apache's mod_rewrite functionality they have a look at URL Rewrite Filter and add the configuration to `yourapp/WEB-INF/web.xml`

```
<code xml>
 <filter>

<filter-name>UrlRewriteFilter</filter-name>

<filter-class>org.tuckey.web.filters.urlrewr
ite.UrlRewriteFilter</filter-class>
 </filter>
 <filter-mapping>

<filter-name>UrlRewriteFilter</filter-name>
       <url-pattern>*</url-pattern>
 </filter-mapping>
</code>
```

htaccess Authentication

`WEB-INF/jetty-web.xml` can be used as normal to configure the HTAccessHandler. The index form is required because you need the authentication check to occur before the PHP scripts are called.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay
Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure
class="org.mortbay.jetty.servlet.WebApplicat
ionContext">

  <Call name="addHandler">
     <Arg type="int">0</Arg>
     <Arg>
        <New
class="org.mortbay.http.handler.HTAccessHand
ler"/>
     </Arg>
   </Call>
</Configure>
```

PHP Configuration

For this purpose PHP must be compiled as a cgi binary.

Refer to the PHP documentation for security considerations but be mindful that they are likely to be referencing the default Apache configuration and your Jetty configuration is likely to be highly customised.

php.ini

- **doc_root** If you are only using the general web server arrangement then **doc_root** can be set to the ResourceBase of your root context. If you want to also use the webapp arrangement then leave **doc_root** bla nk and take any necessary security precautions (use **cgi.force_redirect** or remove the php binary/link from your cgi-bin)
- **cgi.fix_pathinfo** Try using this setting (=0 or 1) if PATH_INFO is not as you expect

php-cgi-fix

In both scenarios above the command prefix is pointing to a script **php-cgi-fix** that works around the fact that PHP running as cgi uses a non-standard CGI environment variable **SCRIPT_FILENAME** to reference the script, rather

than taking the first argument. In certain circumstances PHP will get a bad SCRIPT_FILENAME, then use SCRIPT_NAME to find and parse the file. This produces the expected HTML but also provides a "Not Found" header, (Status: 404) which can confuse some browsers.

```
$!/bin/sh
export SCRIPT_FILENAME=$1

/your/path/to/php-cgi
```

Since Jetty 5.1.5 the CGI servlet sets SCRIPT_FILENAME, but not correctly in all cases

Testing

The **phpinfo()** function is **very handy** for testing your configuration as it will dump lots of information about the configuration.

Save the following to a file `phpinfo.php` into your webserver directory and navigate your browser to `http://<yourjettyhost>/phpinfo.php` and check the results are what you expect.

```
<html>
<head>
  <title>PHP Test</title>
</head>
 <body>
 <?php phpinfo(); ?>
 </body>
</html>
</code>
```