

PHP – HTTP Authentication

In PHP, the `header()` function is used to send an "Authentication Required" message to the client browser causing it to pop up a Username/Password input window. In fact `header()` allows you to send any raw HTTP header.

```
header(string $header, bool $replace = true, int $response_code = 0): void
```

The string parameter is passed to the `header()` function. For example

```
header("HTTP/1.1 404 Not Found");
```

It is used to figure out the HTTP status code to send.

You can also use `header()` function to redirect the browser to another URL.

Once the user has filled in a username and a password, the URL containing the PHP script will be called again with the predefined variables `PHP_AUTH_USER`, `PHP_AUTH_PW`, and `AUTH_TYPE` set to the user name, password and authentication type respectively. These predefined variables are found in the `$_SERVER` array. Only "Basic" and "Digest" authentication methods are supported.

```
<?php

/* Redirect browser */
header("Location: http://www.example.com/");

/* Make sure that code below does not get executed when we redirect. */
exit;

?>
```

The optional `replace` parameter indicates whether the header should replace a previous similar header, or add a second header of the same type, and `response_code` parameter forces the HTTP response code to the specified value.

To be able to force the client authentication, you need a `.htaccess` file in document root folder. Open a new text file, put the following text in it, and save it with `.htaccess` as its name.

```
CGIPassAuth On
```



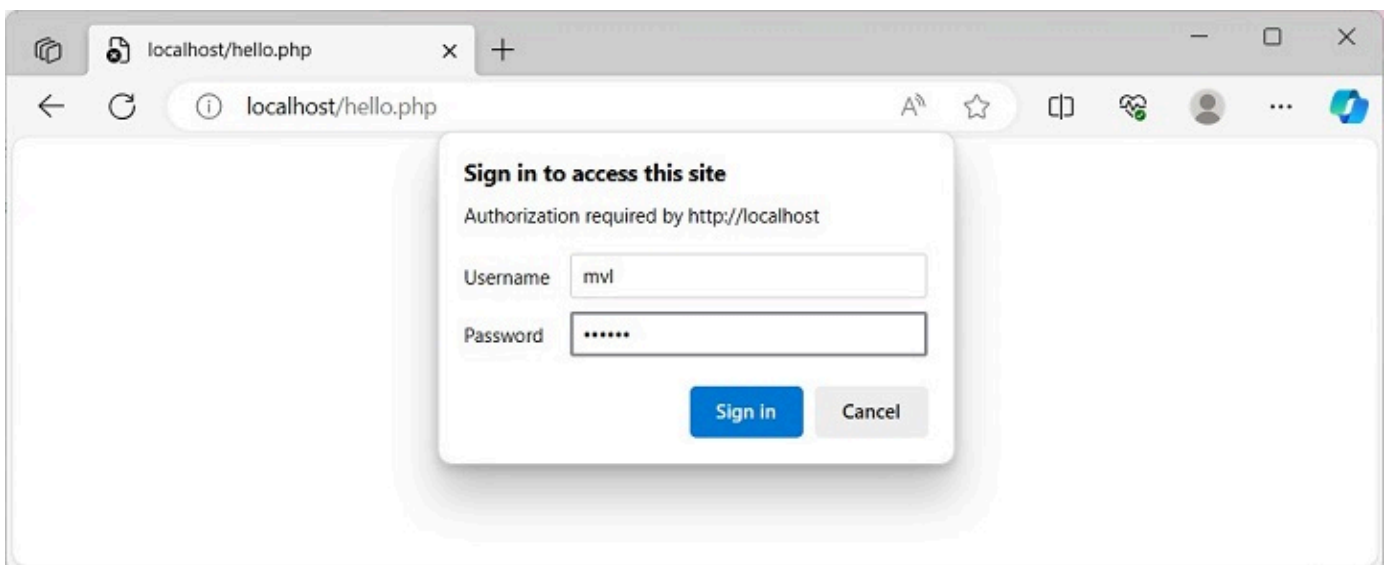
Example

An example script fragment which would force client authentication on a page is as follows –

```
<?php
    if (!isset($_SERVER['PHP_AUTH_USER'])) {
        header('WWW-Authenticate: Basic realm="My Realm"');
        header('HTTP/1.0 401 Unauthorized');
        echo 'User hits Cancel button';
        exit;
    } else {
        echo "<p>Hello {$_SERVER['PHP_AUTH_USER']}</p>";
        echo "<p>You entered {$_SERVER['PHP_AUTH_PW']} as your password.</p>";
    }
?>
```

Output

When you visit the script in a browser, it pops up a dialog box as shown –



Once you click on the sign in button, there may be a backend script to authenticate the login credentials. Once authenticated, two server variables will be created with the keys `PHP_AUTH_USER` and `PHP_AUTH_PW`, which can be verified with the output of `phpinfo()` function.

<code>\$_SERVER['PHP_AUTH_USER']</code>	mvl
<code>\$_SERVER['PHP_AUTH_PW']</code>	asd123