

# \$\_SERVER Array

Whenever a script is executed, a \$\_SERVER array is created and passed to the script. This provides information about the context of the script and environment.

The elements of the array are read-only. You can actually modify them from code, but they will not have any effect on the system once the script terminates.

The following table describes the various elements of the array.

Key	Description
SCRIPT_NAME	Name of the script being executed
SCRIPT_FILENAME	Full path and file name of script being executed
SCRIPT_TYPE	Can be: http, cron or shell.
	http scripts are initiated via a web request.
	cron scripts are initiated via the scheduler or exec() shell scripts are executed via the telnet console.
SCRIPT_FLAGS	Only valid for cron scripts. Flags for the script as follows:
	+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+   Bit 8   Bit 7   Bit 6   Bit 5   Bit 4   Bit 3   Bit 2   Bit 1   Bit 0             KILL   SHELL   RUN_NOW   ONE_SHOT   IN_MEM   +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
	IN_MEM - The script is loading in memory and executing or ready to.
	ONE_SHOT - This script is not recurring, and has been triggered by a call to exec() or an action, or is the startup script.
	RUN_NOW - The script should be run immediately
	SHELL - The script is run from the telnet console KILL - Signals a kill to the script
REMOTE_ADDR	Only valid for http scripts. IP address of device initiating request
SERVER_ADDR	Only valid for http scripts. IP address of <a href="#">Wattmon</a> device
REQUEST_METHOD	Only valid for http scripts. Can be: GET, POST, PUT
HW_PLATFORM	Hardware platform currently executing on
	0 - WM-60 (the original <a href="#">Wattmon</a> )
	1 - WM-PRO <a href="#">WattmonPRO</a>
	2 - WM-MINI <a href="#">WattmonMINI</a>
	10 - WM-MEGA <a href="#">WattmonMEGA</a>

## Examples

### Checking the Hardware Platform

```
<?
// check if the device is running on a MINI and issue an alert
if ($_SERVER['HW_PLATFORM']==2) {
    print("This cannot run on the WM-MINI Device");
}
?>
```

### Checking the HTTP Method

```
<?
// check that the request method is correct

if ($_SERVER['REQUEST_METHOD']=='POST') {

    print("You have sent data via the POST method!");

    // at this point the $_POST array will contain variables submitted
    print_r($_POST);
} else {
    print("You need to submit data to this script via POST.");
    die();
}
?>
```

## Checking the Script Initiator

```
<?
if ($_SERVER['SCRIPT_TYPE']=='shell') {
    print("This was started through the shell.");
}

if ($_SERVER['SCRIPT_TYPE']=='http') {
    print("This was started through an http request.");
}

if ($_SERVER['SCRIPT_TYPE']=='cron') {
    print("This was started through a scheduled task.");

    // check the flags to see if it is a one-time call or a recurring
    execution
    if ($_SERVER['SCRIPT_FLAGS'] & 2 > 0) {
        print("This was started with exec() or via an action, and will only
run once.");
    }
}
?>
```

From:  
<https://www.wattmon.com/dokuwiki/> - **Wattmon Documentation Wiki**

Permanent link:  
<https://www.wattmon.com/dokuwiki/uphp/server>

Last update: **2021/09/13 05:57**

