

Scripts and Scripting Languages FAQ

For cPanel & WHM version 58

[Which versions of Perl are compatible with cPanel & WHM?](#)

[How can I install a Perl Module?](#)

[What is wrong with my CGI and Perl scripts?](#)

[Check the script permissions](#)

[Check the script code for issues](#)

[Add a content type to the script](#)

[Use the suexec module to sanity test permissions](#)

[Use the suexec module to sanity check ownership](#)

[How do I write JavaScript code on my website?](#)

[What if the suexec module breaks my CGI scripts?](#)

[What if all Perl or CGI scripts that do not run as the root user return the "getgrgid: invalid groupid XXXXX" error?](#)

[Can I run PHP 5 and PHP 7 at the same time?](#)

[How can I install PHP 7, and what changes must I make?](#)

[How do I upgrade the system Perl?](#)

[How do I activate or install Zend Guard Loader?](#)

[If I enabled suPHP, how can I override the default settings in the php.ini file?](#)

[How can I add my own modules to cPanel & WHM?](#)

Which versions of Perl are compatible with cPanel & WHM?

cPanel & WHM servers include multiple Perl environments.

Your version of cPanel & WHM determines which Perl environments are available:

cPanel & WHM versions 62 and later include the following Perl environments:

Perl environment	Perl version	Location	Applications that use this Perl environment	Additional information
The server's Perl binary	<ul style="list-style-type: none"> Perl 5.8.8 on CentOS, CloudLinux™, and Red Hat® Enterprise Linux (RHEL) 5 (cPanel & WHM version 56 only). Perl 5.10 on CentOS, CloudLinux, and RHEL 6, and Amazon® Linux. Perl 5.16 on CentOS, CloudLinux, and RHEL 7. 	/usr/bin/perl	<ul style="list-style-type: none"> Operating system maintenance scripts. Some custom modules. 	<ul style="list-style-type: none"> Manage modules for this Perl environment with the <code>cpan</code> command or with WHM's Module Installers interface (<i>WHM >> Home >> Software >> Module Installers</i>).
cPanel-included Perl installation	Perl 5.24	<ul style="list-style-type: none"> This environment self-embeds and compiles in the cPanel binary. <code>/usr/local/cpanel/3rdparty/bin/perl</code> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p>Note:</p> <p>This location is a symlink.</p> </div> <ul style="list-style-type: none"> Install custom modules for this Perl environment into the <code>/opt/cpanel/perl5/524/site_lib</code> directory. <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p>Note:</p> <p>@INC walks this directory after it walks all other directories for modules.</p> </div>	<ul style="list-style-type: none"> The cPanel & WHM software. cPanel system maintenance scripts (in the <code>/usr/local/cpanel/scripts</code> directory). CGI scripts. 	<ul style="list-style-type: none"> This environment allows you to manage the system's Perl binary independently from cPanel & WHM. Use it when you write Perl scripts for a cPanel & WHM server. For more information about how to modify scripts and modules to use this Perl environment, read our Prepare Perl Scripts for cPanel & WHM's Internal Perl documentation.

How can I install a Perl Module?

For directions on how to install a Perl module, read our [Perl Module Installation](#) documentation.

What is wrong with my CGI and Perl scripts?

Problems with CGI and Perl scripts are common, and may be due to a wide variety of reasons.

To troubleshoot the problem, use following methods.



Note:

The following examples troubleshoot the `example.cgi` script.

Check the script permissions

Make **certain** that the script is executable. To display the permissions for a script, run the `ls -la example.cgi` command as the `root` user, where `example.cgi` represents the name of the script:

```
# ls -la example.cgi
-rw-r--r-- 1 burst wheel 41 May 29 16:04 example.cgi
```

In this example, the script has the following permissions:

- The script is readable by everyone.
- The script is writable only by the owner.
- The script is not executable by anyone.

To change the permissions of the script, run the following command:

```
chmod 755 example.cgi
```

This command changes the permissions of the script to the following permissions:

- The script is readable by everyone.
- The script is executable by everyone.
- The script is writable only by the owner.



Note:

For more information about these file system permissions, read about [numeric notation](#).

Check the script code for issues

If the server still returns a 500 error, a code error may be the reason.

Add the following code to the top of your script:

```
#!/usr/bin/perl
use CGI::Carp qw(fatalsToBrowser);
```

This change redirects errors to the browser and does **not** display a 500 error.

To check the code for errors, run the following command:

```
perl -w ./example.cgi
```

This command runs the script with the warnings flag, which displays any errors in the code:

```
# perl -w ./example.cgi
Can't find string terminator '"' anywhere before EOF at ./example.cgi line 3.
```

This message indicates that the Perl script cannot run because it contains fatal errors.

**Note:**

Search online for tutorials on Perl and how to troubleshoot specific errors.

Add a content type to the script

If you found the error within the code, run the script from the command line again:

```
# perl -w ./example.cgi
Hello World !
```

If the script works at the command line but does **not** work in a browser, the script may not send the correct content type to the browser.

In the above example, assume that the `example.cgi` script works in the shell, but does not display in the browser. To fix this problem, add the following line near the top of the script:

```
print "Content-type: text/html\n\n";
```

When you run the updated script, you will see the following output:

```
# perl -w ./example.cgi
Content-type: text/html
Hello World !
```

Use the suexec module to sanity test permissions

If the script does not work in the browser with the correct content type, the `suexec` module may cause the problem. The Apache webserver often includes the `suexec` module as a security measure. It forces a sanity test on the permissions of CGI scripts before it runs them. A sanity test checks the permissions of scripts and determines whether they are accessible.

After the `suexec` module completes the sanity test, it forces the scripts to run with the permissions of the account owner.

To check for proper permissions, run the `ls -la` command.

The output will resemble the following example:

```
# ls -la
total 6
drwxr-xr-x 2 burst wheel 512 May 29 16:04 .
drwxr-xr-x 3 burst wheel 512 May 29 16:03 ..
-rwxr-xr-x 1 burst wheel 78 May 29 16:24 example.cgi
```

The directory "." is the current working directory. Set this directory to the following permissions:

- The script is readable by everyone.
- The script is executable by everyone.
- The script is writable only by the owner.

If you are unsure about problems with a script, examine the `suexec` log file in the `/usr/local/apache/logs/` directory. Use the `tail -f suexec_log` command to watch the log file, and then try to load the script in your browser to see the error that the script caused.

Use the suexec module to sanity check ownership

Confirm that the account's user owns the file. Sometimes, when a password or group file becomes corrupted, you will see a number rather than a username.

When you run the `ls -la` command and a file is corrupted, the output will resemble the following example:

```
# ls -la
total 6
drwxr-xr-x 2 8840 8840 512 May 29 16:04 .
drwxr-xr-x 3 burst wheel 512 May 29 16:03 ..
-rwxr-xr-x 1 8840 8840 78 May 29 16:24 example.cgi
```

To fix this error, run the `chown burst.burst *` command, where `burst` represents the name of the account's owner, to change the owner of the script **and** the directory back to the correct owner.

Even if the owner appears to be correct, it is possible for a script to show the correct name but to actually be associated with a different user ID. This issue appears in the `suexec_log` file.

If you have performed all of these steps and the script still does not work, [submit a support ticket](#).

How do I write JavaScript code on my website?

Search for a tutorial on JavaScript. JavaScript is a language that has many rules and conventions. Because of this, it may take some time to learn.

When you are ready to add your code, edit your HTML files directly to add JavaScript code with a text editor. You can also edit them through cPanel's [File Manager](#) interface (*Home >> Files >> File Manager*).

What if the suexec module breaks my CGI scripts?

If the `suexec` module breaks your CGI scripts, run the `tail -f` command on the `/usr/local/apache/logs/suexec_log` log file to find the error.

What if all Perl or CGI scripts that do not run as the root user return the "getgrgid: invalid groupid XXXXX" error?

The following output is an example of error output from a Perl script that is found in Apache's error log (`/usr/local/apache/logs/error_log`):

```
[Tue Mar 26 09:13:16 2002] [error] [client x.x.x.x] (2)No such file or directory: getgrgid: invalid groupid
32015
[Tue Mar 26 09:13:16 2002] [error] (2)No such file or directory: exec of /home/username/public_html/utility.cgi
failed
[Tue Mar 26 09:13:16 2002] [error] [client x.x.x.x] Premature end of script headers: /home/username/public_html
/script.cgi
```

To fix this error, run the following command:

```
chmod 644 /etc/group
```

If you continue to experience an issue, check the permissions on the `passwd` file and `shadow` file.

If that fails to fix the issue, [submit a support ticket](#).

Can I run PHP 5 and PHP 7 at the same time?

You can run multiple versions of PHP 5 **only** if your system runs EasyApache 4. EasyApache 4 is available in cPanel & WHM versions 11.52 and later.

You can also run PHP 7 if your system runs EasyApache 4 and cPanel & WHM version 54 or later.

How can I install PHP 7, and what changes must I make?

To install PHP 7, perform the following steps:

1. Run the following command on the command line: `yum install ea-php70`. For more information, read our [How to Locate and Install a PHP Version or Extension](#) documentation.
2. Assign PHP 7 to your desired users in WHM's [MultiPHP Manager](#) interface (*Home >> Software >> MultiPHP Manager*). You can also set the system default to PHP 7 in this interface.

**Notes:**

- Make **certain** that the PHP programs and scripts that you wish to run are compatible with PHP 7.
- You can **only** use PHP 7 on systems that run EasyApache 4.

How do I upgrade the system Perl?

We recommend that you allow your operating system's distribution to control the system Perl installation.

How do I activate or install Zend Guard Loader?

To install and activate Zend Guard Loader in EasyApache 3, run the `/scripts/installzendopt` script at the command line as the `root` user.

To install and activate Zend Guard Loader in EasyApache 4, use yum or WHM's [EasyApache 4](#) interface (`Home >> Software >> EasyApache 4`).

**Notes:**

- Zend Guard Loader was previously known as Zend Optimizer.
- Zend Guard Loader does **not** work with the PHP debugger.
- Zend Guard Loader is **not** available with PHP 7.

If I enabled suPHP, how can I override the default settings in the php.ini file?

Because the PHP module does not have control over the `httpd.conf` or `.htaccess` files, place a customized `php.ini` file in the directory that contains your PHP scripts.

How can I add my own modules to cPanel & WHM?

For more information about how to install modules to cPanel & WHM, read our [Perl Module Installation](#) documentation.