Grant permissions for php-cli users on IBM i

Issue

When running a php script using php-cli, what authorities need to be granted?

When running a script in a web browser, the default Apache user QTMHHTTP has authority to run scripts. Is there an easy way to insure that a user other than QTMHHTTP will have authority to run a script using php-cli?

Environment

Any version of Zend Server for IBM i running on any supported version of IBM i.

Resolution

Upgrades and clean installs will require redoing these permissions

When you upgrade Zend Server, you often replace some of the directories that will be given new permissions in this exercise. And when you do a clean install or install a new version that you need to migrate to, again there will be all new directories without these permissions. In these cases, you will need to redo the steps in this article to provide the needed permissions. So, it is important to keep notes on the permissions set so you can go back and do this again when needed. You could even create a CL program to do this task for you and run it whenever needed.

Grant the user *RX permissions to directory /usr/local/zendphp74 and all underlying directories. From the 5250 command line, signed on as QSECOFR:

Note

```
Replace PHPUSER in the following examples with the actual user profile you need to run
the script via CLI.
```

For Zend Server 2020.x and higher

```
CHGAUT OBJ('/usr/local/zendphp74') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

For Zend Server 9 or higher:

```
CHGAUT OBJ('/usr/local/zendphp7') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

For Zend Server 6 through 8.5:

```
CHGAUT OBJ('/usr/local/zendsvr6') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

For Zend Server 5:

```
CHGAUT OBJ('/usr/local/zendsvr') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

The user will need to be able to write to the log files, so that messages do not display on the terminal or in a QPRINT spool file:

For Zend Server 9 or higher:

```
CHGAUT OBJ('/usr/local/zendphp7/var/log') USER(PHPUSER) DTAAUT(*RWX) SUBTREE(*ALL)
```

For Zend Server 6 through 8.5:

```
CHGAUT OBJ('/usr/local/zendsvr6/var/log') USER(PHPUSER) DTAAUT(*RWX) SUBTREE(*ALL)
```

For Zend Server 5:

```
CHGAUT OBJ('/usr/local/zendsvr/var/log') USER(PHPUSER) DTAAUT(*RWX) SUBTREE(*ALL)
```

Note: Security level 30 may also require the Object Alter authority

We have a report from a customer at Security Level 30 who found they needed to also set the object authority to alter the log files:

For Zend Server 9 or higher:

CHGAUT OBJ('/usr/local/zendphp7/var/log') USER(PHPUSER) DTAAUT(*RWX) OBJAUT(*OBJALTER) SUBTREE(*ALL)

For Zend Server 6 through 8.5:

CHGAUT OBJ('/usr/local/zendsvr6/var/log') USER(PHPUSER) DTAAUT(*RWX) OBJAUT(*OBJALTER) SUBTREE(*ALL)

For Zend Server 5:

CHGAUT OBJ('/usr/local/zendsvr/var/log') USER(PHPUSER) DTAAUT(*RWX) OBJAUT(*OBJALTER) SUBTREE(*ALL)

The user will also need *RX authority to the PHP scripts and other web content. For example, to grant permissions for the user to the default document root and all underlying directories:

For Zend Server 9 or higher:

```
CHGAUT OBJ('/www/zendphp7/htdocs') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

For Zend Server 6 through 8.5:

```
CHGAUT OBJ('/www/zendsvr6/htdocs') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

For Zend Server 5:

```
CHGAUT OBJ('/www/zendsvr/htdocs') USER(PHPUSER) DTAAUT(*RX) SUBTREE(*ALL)
```

Details

It can be more convenient to simply run the above commands for user *PUBLIC. This would allow any user successfully signed in with valid credentials to use PHP scripts run via php-cli. However, this is generally considered less secure than specifying allowed users individually.

Users with the *ALLOBJ special authority do not need to have permissions granted in order to run scripts using php-cli. Sometimes a developer will not have any problem running scripts in php-cli, but will discover that the users in production are having permissions problems. This is usually because the developer has *ALLOBJ special authority, while typical users in production do not.