

Chamilo LMS - Bug #7020

Cpanel conflicts

21/03/2014 13:44 - Julio Montoya

Status:	Bug resolved	Start date:	21/03/2014
Priority:	Normal	Due date:	24/04/2014
Assignee:	Julio Montoya	% Done:	100%
Category:	Installation / Migration	Estimated time:	0.00 hour
Target version:	1.9.8	Spent time:	0.45 hour
Complexity:	Normal	SCRUM pts - complexity:	

Description

I had this error when trying to enter a course created in a CHamilo inside a Cpanel hosting.

```
[Fri Mar 21 13:13:20 2014] [error] [client ****] SoftException in Application.cpp:256: File "/*  
*/  
*/courses/TESTCHAMILO/index.php" is writeable by group, referer: http://*/user_portal.php?nos  
ession=true
```

With an Internal server error 500:

After searching I found that cpanel doesn't "like" the **666** default permission:

I changed to **644**. And now courses work great!

This is the setting:

```
main/admin/settings.php?search_field=permissions_for_new_files&submit_button=&qf__search_settings  
=&category=search_setting
```

"Permissions for new files".

Why we don't use 644?

Related issues:

Related to Chamilo LMS - Support #7018: Chamilo download (best for cpanel) d...

Bug resolved 20/03/2014

Associated revisions

Revision 68e449f2 - 22/04/2014 14:03 - Julio Montoya

Improving course creation message see #7020

History

#1 - 21/03/2014 14:07 - Yannick Warnier

- Status changed from New to Assigned

- Assignee set to Julio Montoya

It's kind of funny that you (of all people) would have this problem, given this is the most common installation problem that we have (see, between other references, including the installation guide: <http://beeznest.wordpress.com/2012/04/21/problemas-al-crear-cursos-en-chamilo-despues-de-la-instalacion-error-500/>). The correct permissions depend on the configuration of the server and might change dramatically from one server to another.

This is why the installer for Chamilo (since 1.9.6, I think) includes the creation **and execution** of a PHP file inside a fictitious course directory. If the script does not execute, we consider the permissions to be wrong and subsequently update these permissions in Chamilo **during the installation**.

If this Chamilo installation was installed through the 1.9.6 installer (maybe it was installed as a previous version and then upgraded?), then it means that this procedure is not working, and it would be great to find out what exactly prevents it to work (and you have a use case at your disposal, which is difficult to have as not all CPanel installations have the same problem).

You can find more about the permissions check here:

- <https://github.com/chamilo/chamilo-lms/commit/2e7a124ab650f41ff0e2124398c47c911e40147b>
- <https://github.com/chamilo/chamilo-lms/commit/4925b9ebd4d727f42b943af8eca5e4203bf8b75b>

You'll see that we're actually "crawling" the php file (instead of reading the source on disk) to make sure it is **interpreted** by the web server and, if there is an error, it pops up at this point.

I'm assigning it to you because you have the right tools in hand right now to fix it definitively.

#2 - 21/03/2014 14:15 - Julio Montoya

- Assignee changed from Julio Montoya to Yannick Warnier

Yannick Warnier wrote:

It's kind of funny that you (of all people) would have this problem, given this is the most common installation problem that we have (see, between other references, including the installation guide: <http://beeznest.wordpress.com/2012/04/21/problemas-al-crear-cursos-en-chamilo-despues-de-la-instalacion-error-500/>). The correct permissions depend on the configuration of the server and might change dramatically from one server to another.

I **never** had that problem before.

This is why the installer for Chamilo (since 1.9.6, I think) includes the creation **and execution** of a PHP file inside a fictitious course directory. If the script does not execute, we consider the permissions to be wrong and subsequently update these permissions in Chamilo **during the installation**.

This doesn't work.

If this Chamilo installation was installed through the 1.9.6 installer (maybe it was installed as a previous version and then upgraded?), then it means that this procedure is not working, and it would be great to find out what exactly prevents it to work (and you have a use case at your disposal, which is difficult to have as not all CPanel installations have the same problem).

It was an special "cpanel version" (God knows what that mean version) downloaded here:

<http://www.chamilo.org/en/download>

See also for the cpanel version naming [#7018](#).

You can find more about the permissions check here:

- <https://github.com/chamilo/chamilo-lms/commit/2e7a124ab650f41ff0e2124398c47c911e40147b>
- <https://github.com/chamilo/chamilo-lms/commit/4925b9ebd4d727f42b943af8eca5e4203bf8b75b>

You'll see that we're actually "crawling" the php file (instead of reading the source on disk) to make sure it is **interpreted** by the web server and, if there is an error, it pops up at this point.

I'm assigning it to you because you have the right tools in hand right now to fix it definitively.

I will need a Cpanel test enviroment to try this, this plays with a php + some security suEXEC modules?

#3 - 21/03/2014 14:56 - Julio Montoya

- SCRUM pts - complexity deleted (?)

If chamilo is inside a sub folder the permission to this folder must be 755 too

#4 - 21/03/2014 14:59 - Yannick Warnier

Julio Montoya wrote:

Yannick Warnier wrote:

It's kind of funny that you (of all people) would have this problem, given this is the most common installation problem that we have (see, between other references, including the installation guide: <http://beeznest.wordpress.com/2012/04/21/problemas-al-crear-cursos-en-chamilo-despues-de-la-instalacion-error-500/>). The correct permissions depend on the configuration of the server and might change dramatically from one server to another.

I **never** had that problem before.

... which is a lesson as to why we should be patient with our community :-)

This being said, you never had the problem because you work in a dedicated-only servers environment, with little cases of CPanel (be grateful for that :-p)

This is why the installer for Chamilo (since 1.9.6, I think) includes the creation **and execution** of a PHP file inside a fictitious course directory. If the script does not execute, we consider the permissions to be wrong and subsequently update these permissions in Chamilo **during the installation**.

This doesn't work.

Well, we did a series of tests that proved the contrary, but there are always new cases, so let's enjoy this one.

If this Chamilo installation was installed through the 1.9.6 installer (maybe it was installed as a previous version and then upgraded?), then it means that this procedure is not working, and it would be great to find out what exactly prevents it to work (and you have a use case at your disposal, which is difficult to have as not all CPanel installations have the same problem).

It was an special "cpanel version" (God knows what that mean version) downloaded here:

<http://www.chamilo.org/en/download>

Oh, that's only a difference in settings to make it directly use the right permissions (because it's .tar.gz, which maintains unix permissions inside the compressed archive) for directories **outside** of those that need to be writeable.

See also for the cpanel version naming [#7018](#).

You can find more about the permissions check here:

- <https://github.com/chamilo/chamilo-lms/commit/2e7a124ab650f41ff0e2124398c47c911e40147b>
- <https://github.com/chamilo/chamilo-lms/commit/4925b9ebd4d727f42b943af8eca5e4203bf8b75b>

You'll see that we're actually "crawling" the php file (instead of reading the source on disk) to make sure it is **interpreted** by the web server and, if there is an error, it pops up at this point.

I'm assigning it to you because you have the right tools in hand right now to fix it definitively.

I will need a Cpanel test enviroment to try this, this plays with a php + some security suEXEC modules?

It is most likely to be something like this, yes. There are a few combinations possible. One of them is SELinux (Security Enhanced Linux), other is suEXEC, others do other stuff still, but independent of what you have there, the procedure that we used in the installer should work in all circumstances. There's probably something wrong while reading the results of the PHP script.

#5 - 21/03/2014 15:43 - Julio Montoya

permissions_for_new_directories = 755 too.

#6 - 21/03/2014 16:14 - Julio Montoya

- Assignee changed from Yannick Warnier to Julio Montoya

Yannick Warnier wrote:

Julio Montoya wrote:

Yannick Warnier wrote:

It's kind of funny that you (of all people) would have this problem, given this is the most common installation problem that we have (see, between other references, including the installation guide: <http://beeznest.wordpress.com/2012/04/21/problemas-al-crear-cursos-en-chamilo-despues-de-la-instalacion-error-500/>). The correct permissions depend on the configuration of the server and might change dramatically from one server to another.

I **never** had that problem before.

... which is a lesson as to why we should be patient with our community :-)

This being said, you never had the problem because you work in a dedicated-only servers environment, with little cases of CPanel (be grateful for that :-p)

This is why the installer for Chamilo (since 1.9.6, I think) includes the creation **and execution** of a PHP file inside a fictitious course directory. If the script does not execute, we consider the permissions to be wrong and subsequently update these permissions in Chamilo **during the installation**.

This doesn't work.

Well, we did a series of tests that proved the contrary, but there are always new cases, so let's enjoy this one.

That's why I reported **this** task :

If this Chamilo installation was installed through the 1.9.6 installer (maybe it was installed as a previous version and then upgraded?), then it means that this procedure is not working, and it would be great to find out what exactly prevents it to work (and you have a use case at your disposal, which is difficult to have as not all CPanel installations have the same problem).

It was an special "cpanel version" (God knows what that mean version) downloaded here:

<http://www.chamilo.org/en/download>

Oh, that's only a difference in settings to make it directly use the right permissions (because it's .tar.gz, which maintains unix permissions inside the compressed archive) for directories **outside** of those that need to be writeable.

That explanation need to be added in the download page, you can't assume that people know that.

See also for the cpanel version naming [#7018](#).

You can find more about the permissions check here:

- <https://github.com/chamilo/chamilo-lms/commit/2e7a124ab650f41ff0e2124398c47c911e40147b>
- <https://github.com/chamilo/chamilo-lms/commit/4925b9ebd4d727f42b943af8eca5e4203bf8b75b>

You'll see that we're actually "crawling" the php file (instead of reading the source on disk) to make sure it is **interpreted** by the web server and, if there is an error, it pops up at this point.

I'm assigning it to you because you have the right tools in hand right now to fix it definitively.

I will need a Cpanel test enviroment to try this, this plays with a php + some security suEXEC modules?

It is most likely to be something like this, yes. There are a few combinations possible. One of them is SELinux (Security Enhanced Linux), other is suEXEC, others do other stuff still, but independent of what you have there, the procedure that we used in the installer should work in all circumstances. There's probably something wrong while reading the results of the PHP script.

I will try to reproduce the error with the latest 198 version.

#7 - 21/03/2014 16:33 - Julio Montoya

- *Subject changed from Cpanel conflict when creating courses to Cpanel conflicts*

I just download a zip file from here <https://github.com/chamilo/chamilo-lms/tree/1.9.x>

Unzip it and and I received this "Internal Server Error".

All files have **664** permissions and all folders have **775** permissions.

In the installation guide there's nothing that helps.

https://stable.chamilo.org/documentation/installation_guide.html

I have this in my error_log:

```
SoftException in Application.cpp:256: File "/*/*/test/chamilo-lms-1.9.x/index.php" is writeable by group,
```

#8 - 21/03/2014 16:36 - Julio Montoya

- *Assignee changed from Julio Montoya to Yannick Warnier*

I also download Wordpress in the server and the default permissions are **755** and **644** that just works. I think is not good idea of having a "best for cpanel" version, the best version must be only one zip file ...

#9 - 21/03/2014 16:41 - Julio Montoya

I have the same issue trying with the first package here <http://www.chamilo.org/download>.

#10 - 21/03/2014 16:57 - Julio Montoya

With the cpanel I can go to the installation process.

#11 - 21/03/2014 17:14 - Julio Montoya

- Status changed from Assigned to Needs more info

I found some bugs. The installer shows "course folder is writable" but the course folder creation failed!!

#12 - 21/03/2014 18:11 - Julio Montoya

I found a problem here. The function "check_course_script_interpretation()" (in main/install/install.lib.php) tries to reach a file using fsockopen/curl_init/fopen folder via the web. But in my case my URL was not yet available.

So I have this nonsense, it says the folder have permissions to write, but the folder creation was not succeeded.

In my case the folder creation was fine. The error was that it was not reachable outside and it blocks the installation process.

I thought that your blog post resolves this task lol :)

```
function check_course_script_interpretation($course_dir, $course_attempt_name, $file = 'test.php')
{
    $output = false;
    //Write in file
    $file_name = $course_dir.'/'.$file;
    $content = '<?php echo "123"; exit;';

    if (is_writable($file_name)) {
        if ($handler = @fopen($file_name, "w")) {
            //write content
            if (fwrite($handler, $content)) {
                $sock_errno = '';
                $sock_errmsg = '';
                $url = api_get_path(WEB_COURSE_PATH).$course_attempt_name.'/'.$file;

                $parsed_url = parse_url($url);
                // $scheme = isset($parsedUrl['scheme']) ? $parsedUrl['scheme'] : ''; //http
                $host = isset($parsed_url['host']) ? $parsed_url['host'] : '';
                $path = isset($parsed_url['path']) ? $parsed_url['path'] : '/';
                $port = isset($parsed_url['port']) ? $parsed_url['port'] : '80';

                //Check fsockopen (doesn't work with https)
                if ($fp = @fsockopen(str_replace('http://', '', $url), -1, $sock_errno, $sock_errmsg, 60)) {
                    $out = "GET $path HTTP/1.1\r\n";
                    $out .= "Host: $host\r\n";
                    $out .= "Connection: Close\r\n\r\n";

                    fwrite($fp, $out);
                    while (!feof($fp)) {
                        $result = str_replace("\r\n", '', fgets($fp, 128));
                        if (!empty($result) && $result == '123') {
                            $output = true;
                        }
                    }
                    fclose($fp);
                }
                //Check allow_url_fopen
            } elseif (ini_get('allow_url_fopen')) {
                if ($fp = @fopen($url, 'r')) {
                    while ($result = fgets($fp, 1024)) {
                        if (!empty($result) && $result == '123') {
                            $output = true;
                        }
                    }
                    fclose($fp);
                }
            } // Check if has support for cURL
        } elseif (function_exists('curl_init')) {
            $ch = curl_init();
            curl_setopt($ch, CURLOPT_HEADER, 0);
            curl_setopt($ch, CURLOPT_URL, $url);
            //curl_setopt($ch, CURLOPT_TIMEOUT, 30);
            curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
            $result = curl_exec ($ch);
            if (!empty($result) && $result == '123') {
                $output = true;
            }
        }
    }
}
```

```

    }
    curl_close($ch);
}
}
@fclose($handler);
}
}
return $output;
}

```

#13 - 15/04/2014 19:12 - Yannick Warnier

- % Done changed from 0 to 40

I'm adding some progress here, because there is.

#14 - 16/04/2014 20:02 - Yannick Warnier

- Assignee changed from Yannick Warnier to Erick Ocrospoma

#15 - 20/04/2014 01:38 - Yannick Warnier

- Due date set to 24/04/2014

#16 - 21/04/2014 20:04 - Yannick Warnier

- Status changed from Needs more info to Assigned

- Assignee changed from Erick Ocrospoma to Julio Montoya

- Priority changed from Normal to High

- % Done changed from 40 to 50

OK, I understand, Julio.

So the issue here is the following situation:

- if we just check the permissions on disk, it says it's ok (but they are probably **not** ok because of the non-interpretation of the script, which is not checked)
- we check the URL to see if the script is interpreted, but the URL is not available, resulting in a "not interpretable" error, which denies the right to install (however, the script is probably just fine)

So there is no perfect solution here... we should show a warning if it fails but **not** block the installation process.

So we should:

- show a green box (label-success) if the install could validate in both ways
- show an orange box (label-warning) if the check on dirs was OK but not the check on the file execution
- show a red box (label-danger) if none of the two options passed (in this case **only**, block the install procedure)

Under the hood, we should

- check the first permission settings that allow to write the file and save them temporarily.
- Then check the URL through fsockopen/curl/wget.
 - If the second method succeeded, assign permissions as tried with the is_writeable() option and all that.
 - If the second method fails, assign the permissions anyway, show the warning message and continue

This way we add value (information) in comparison to the install process of 1.9.6, but we don't block anyone.

The orange message should be:

\$InstallWarningCouldNotInterpretPHP = "Warning: the installer detected an error while trying to reach the test file at %s. It looks like the PHP script could not be interpreted. This could be a warning sign for future problems when creating courses. Please check the installation guide for more information about permissions. If you are installing a site with a URL that doesn't resolve yet, you can probably ignore this message.";

The %s is the URL of the tested file.

#17 - 22/04/2014 14:04 - Julio Montoya

- Status changed from Assigned to Needs testing

- Assignee deleted (Julio Montoya)

- % Done changed from 50 to 80

I added new changes, I added the variable in translate.chamilo

#18 - 04/05/2014 23:27 - Yannick Warnier

- *Priority changed from High to Normal*

I installed several times and couldn't find an issue, but this is clearly a task that should be left open until the stable release, just to give people time to review.

#19 - 05/06/2014 13:49 - Yannick Warnier

- *Status changed from Needs testing to Bug resolved*

- *Assignee set to Julio Montoya*

- *% Done changed from 80 to 100*

I received no complaint in a month, despite the fact that beta and RC packages have been released. Closing.