TYPO3 Core - Bug #29274

Regression on session handling for security fix

2011-08-26 13:59 - Ernesto Baschny

Status:	Closed	**Start date:**	2011-08-26
Priority:	Must have	**Due date:**	
Assignee:	Helmut Hummel	**% Done:**	100%
Category:	Frontend	**Estimated time:**	0.00 hour
Target version:	4.3.14		
TYPO3 Version:	4.3	**Complexity:**	
PHP Version:		**Is Regression:**	
Tags:		**Sprint Focus:**	

Description

After upgrading from 4.3.11 to 4.3.12, an embedded application (run as a TYPO3 extension) did not work anymore. After some research, I discovered it was due to the change introduced in #24456, which moved the call of "session_start()" from a place where it was only called on demand (when doing a challenge/response login) to a place where it is **always** being called (even on the frontend).

Two issues with this changes:

1) My embedded application for a misfortune also does a session_start. But it also includes lots of Objects into this session. The classes for this objects are loaded by the application before calling session_start(), so PHP can build the objects just fine.

But now when TYPO3 calls a session_start on **every hit** and very early: the classes of my applications are not loaded yet! Thus the session is filled with "__PHP_Incomplete_Class" objects! The application no longer works.

2) Another issue which happened after this change is that several customer sites began running over quota, simply because every FE hit (even from Google & Co) created new PHP session (files in phptmp). This was not so before and will cause annoyances for bigger sites, which are tuned for fast FE rendering explicitly without Cookies / Sessions.

In my situation the result is worse than the "security gain" obtained by this change. So please consider either reverting it again (also in 4.4 and 4.5) or apply it somewhere else.

Related issues:

Related to TYPO3 Core - Bug #24456: Information disclosure during backend login	**Closed**	**2011-01-03**
Related to TYPO3 Core - Bug #28900: All links have Parameter PHPSESSID at fir...	**Closed**	**2011-08-10**
Related to TYPO3 Core - Feature #29750: Pre-Session Hook in t3lib_userauth	**Rejected**	**2011-09-13**
Related to TYPO3 Core - Bug #28694: PHP Warning: session_start()	**Closed**	**2011-08-03**
Related to TYPO3 Core - Bug #29927: Remove occurences of session_start()	**Closed**	**2011-09-17**
Has duplicate TYPO3 Core - Bug #28948: Session is always started	**Closed**	**2011-08-12**

Associated revisions

Revision 3e18ab87 - 2011-09-19 21:24 - Helmut Hummel

[BUGFIX] Don't unnecessarily start PHP session

Because of an information disclosure problem in the backend login
we moved the session_start() in t3lib_userauth in a place which caused
unwanted side effects with 3rd party extensions.

Revert that change to avoid compatibility and performance problems
and instead send no cache headers earlier in t3lib_userauth
to also fix the information disclosure.

Releases: 4.3, 4.4, 4.5, 4.6
Resolves: #29274
Related: #24456, #28694

Change-Id: I87226a21d9b1955773ceb3c377fa1b4c9938e6b2
Reviewed-on: http://review.typo3.org/5007
Reviewed-by: Christopher Hlubek

Reviewed-by: Dmitry Dulepov
Tested-by: Dmitry Dulepov
Reviewed-by: Xavier Perseguers
Reviewed-by: Jigal van Hemert
Tested-by: Jigal van Hemert

Revision 3863b1be - 2011-09-19 21:41 - Helmut Hummel

[BUGFIX] Don't unnecessarily start PHP session

Because of an information disclosure problem in the backend login
we moved the session_start() in t3lib_userauth in a place which caused
unwanted side effects with 3rd party extensions.

Revert that change to avoid compatibility and performance problems
and instead send no cache headers earlier in t3lib_userauth
to also fix the information disclosure.

Releases: 4.3, 4.4, 4.5, 4.6
Resolves: #29274
Related: #24456, #28694

Change-Id: I87226a21d9b1955773ceb3c377fa1b4c9938e6b2
Reviewed-on: http://review.typo3.org/5070
Reviewed-by: Helmut Hummel
Tested-by: Helmut Hummel

Revision 3e1cd735 - 2011-09-19 22:06 - Helmut Hummel

[BUGFIX] Don't unnecessarily start PHP session

Because of an information disclosure problem in the backend login
we moved the session_start() in t3lib_userauth in a place which caused
unwanted side effects with 3rd party extensions.

Revert that change to avoid compatibility and performance problems
and instead send no cache headers earlier in t3lib_userauth
to also fix the information disclosure.

Releases: 4.3, 4.4, 4.5, 4.6
Resolves: #29274
Related: #24456, #28694

Change-Id: I87226a21d9b1955773ceb3c377fa1b4c9938e6b2
Reviewed-on: http://review.typo3.org/5071
Reviewed-by: Helmut Hummel
Tested-by: Helmut Hummel

Revision f8902b0b - 2011-09-19 22:06 - Helmut Hummel

[BUGFIX] Don't unnecessarily start PHP session

Because of an information disclosure problem in the backend login
we moved the session_start() in t3lib_userauth in a place which caused
unwanted side effects with 3rd party extensions.

Revert that change to avoid compatibility and performance problems
and instead send no cache headers earlier in t3lib_userauth
to also fix the information disclosure.

Releases: 4.3, 4.4, 4.5, 4.6
Resolves: #29274
Related: #24456, #28694

Change-Id: I87226a21d9b1955773ceb3c377fa1b4c9938e6b2
Reviewed-on: http://review.typo3.org/5072
Reviewed-by: Helmut Hummel
Tested-by: Helmut Hummel

History

#1 - 2011-08-27 19:46 - Ingmar Schlecht

- Assignee set to Helmut Hummel

#2 - 2011-08-27 19:46 - Ingmar Schlecht

- Status changed from New to Accepted

#3 - 2011-08-31 23:13 - Helmut Hummel

- File session-fix.diff added

Please try the attached patch.

#4 - 2011-09-08 18:28 - Frederic Gaus

The Patch is working for me.

This solves the following error in conjunction with TypoGento:
Fatal error: Mage_Core_Model_Session_Abstract::getMessages() [mage-core-model-session-abstract.getmessages]: The script tried to execute a method or access a property of an incomplete object. Please ensure that the class definition "Mage_Core_Model_Message_Collection" of the object you are trying to operate on was loaded *before* unserialize() gets called or provide a __autoload() function to load the class definition in XXX on line 215

#5 - 2011-09-09 09:05 - Roland Hager

We updated to 4.4.10 yesterday and run into exactly that issue. I would even consider it a severe vulnerability to a DoS attack. Since the network filesystem we are using for our multi server environment has a hard limit for the maximum number of files per directory our site was getting real slow after about two hours running the new version. A BE/FE-Login was not possible any more, because no new sessions could be made due to a filled up session directory.

It should be quite easy to generate loads of sessions on the serverside and depending on the filesystem used by the server causing issues related to quota or filesystemlimits. This way an attacker can prevent normal users from logging in to the BE and/or FE.

The good news: The provided patch seems to do the trick. No null-byte sessions per hit and the login still works -Thanks!

#6 - 2011-09-12 22:01 - Christian Opitz

I wrote an extension that is a middleware between Zend Framework and TYPO3 - since the call to session_start mentioned above, a later call to Zend_Session::start() rightly leads to this exception:

```
session has already been started by session.auto-start or session_start()

Zend_Session_Exception thrown in file
[...]Zend/Session.php in line 462.
```

The patch makes that this exception is not thrown when nobody is logged in but when someone is trying to log in it's there again. Could you pleeaase wrap the call to session_start() in a protected function so that it's possible to override it from a XCLASS? I provide a patch if you like...

#7 - 2011-09-17 18:41 - Helmut Hummel

The problem with the information disclosure was, that if a valid user was found the session_start() (which also sends headers) was called before noCacheHeaders are sent, while in all other situations (invalid username) session_start() is called **after** noCacheHeaders are sent.

In the initial approach (#24456) we made sure that the order was always the same by starting the session always before sending the noCacheHeaders.

As this obviously caused a lot of problems I now suggest to do it the other way around and send the noCacheHeaders in any case before we may start a session.

#8 - 2011-09-17 18:44 - Mr. Hudson

Patch set 1 of change I87226a21d9b1955773ceb3c377fa1b4c9938e6b2 has been pushed to the review server.
It is available at http://review.typo3.org/5007

#9 - 2011-09-19 21:30 - Anonymous

- Status changed from Accepted to Resolved

- % Done changed from 0 to 100

Applied in changeset 3e18ab8726e5586d9ef8888ffce49a6cf7e03b53.

#10 - 2017-10-24 20:19 - Riccardo De Contardi

- Status changed from Resolved to Closed

Files