TYPO3.Fluid - Bug # 60069

Status:	Resolved	**Priority:**	Must have
Author:	Philipp Maier	**Category:**	ViewHelpers
Created:	2014-07-03	**Assigned To:**	Bastian Waidelich
Updated:	2014-08-26	**Due date:**	
Has patch:	No		
Affected Flow version:	Git master		
Subject:	Objects cast to string are not escaped		

Description

Basically if you have a class like this:

```
class HelloWorld {
public function __toString() { return '<script>alert("hello world");</script>' }
}
```

and you assign it as a fluid variable like this:

```
$this->view->assign('helloworld', new HelloWorld());
```

and have a template like this:

```
{helloworld}
```

you're going to have a bad time.

Related issues:

related to Core - Bug # 60082: Backport: Objects cast to string are not escaped	**New**	**2014-07-03**

Associated revisions

Revision 315f3753 - 2014-07-03 15:37 - Bastian Waidelich

[!!!][BUGFIX] Enforce escaping on string-casted objects

This change assures that the escape interceptor is active for objects
that are casted to strings implicitly.

Background:
For HTML requests Fluid internally applies the
``HtmlspecialcharsViewHelper`` on variables before rendering them.
An ``is_string()`` check in the escaping ViewHelpers effectively
disabled this behavior for objects that are converted to strings
implicitly via a ``__toString()`` method.

This is a breaking change if you relied on the previous behavior that
escaping is disabled for objects. In this case you can apply the
format.raw ViewHelper to achieve the old behavior::

```
{object -> f:format.raw()}
```

But be aware that this might pose a security issue if
``$object->__toString()`` returns an unsecure string.

Change-Id: I7c66d3247ffda8f5dc5a03a823f0a05a56ff686b
Fixes: #60069
Releases: master, 2.2, 2.1

Revision 9744e768 - 2014-08-18 21:55 - Bastian Waidelich

[!!!][BUGFIX] Enforce escaping on string-casted objects

This change assures that the escape interceptor is active for objects
that are casted to strings implicitly.

Background:
For HTML requests Fluid internally applies the
``HtmlspecialcharsViewHelper`` on variables before rendering them.
An ``is_string()`` check in the escaping ViewHelpers effectively
disabled this behavior for objects that are converted to strings
implicitly via a ``__toString()`` method.

This is a breaking change if you relied on the previous behavior that
escaping is disabled for objects. In this case you can apply the
format.raw ViewHelper to achieve the old behavior::

    {object -> f:format.raw()}

But be aware that this might pose a security issue if
``$object->__toString()`` returns an unsecure string.

Change-Id: I7c66d3247ffda8f5dc5a03a823f0a05a56ff686b
Fixes: #60069
Releases: master, 2.2, 2.1

History

#1 - 2014-07-03 10:41 - Bastian Waidelich

- Category set to Core
- Status changed from New to Accepted
- Assigned To set to Bastian Waidelich
- Affected Flow version changed from (any) to Git master

This is bad, thanks for reporting!

#2 - 2014-07-03 10:52 - Philipp Maier

I forgot to mention that the CMS version behaves the very same way.
Should I create an issue in that bugtracker as well?

#3 - 2014-07-03 12:57 - Bastian Waidelich

- Category changed from Core to ViewHelpers

#4 - 2014-07-03 12:59 - Bastian Waidelich

Philipp Maier wrote:

> *I forgot to mention that the CMS version behaves the very same way.*
> *Should I create an issue in that bugtracker as well?*

> *es that would be great!*

FYI: the culprit is line 66 of
https://git.typo3.org/Packages/TYPO3.Fluid.git/blob/HEAD:/Classes/TYPO3/Fluid/ViewHelpers/Format/HtmlspecialcharsViewHelper.php#l66

and a possible fix is to replace

```
1if (!is_string($value)) {
```

by

```
1if (!is_string($value) && !(is_object($value) && method_exists($value, '__toString'))) {
```

#5 - 2014-07-03 13:13 - Philipp Maier

Cool that you found the issue already!

Copied the Bug to the CMS Tracker:
http://forge.typo3.org/issues/60082

#6 - 2014-07-03 15:26 - Gerrit Code Review

- Status changed from Accepted to Under Review

Patch set 1 for branch **master** of project **Packages/TYPO3.Fluid** has been pushed to the review server.
It is available at https://review.typo3.org/31312

#7 - 2014-07-03 15:37 - Gerrit Code Review

Patch set 2 for branch **master** of project **Packages/TYPO3.Fluid** has been pushed to the review server.
It is available at https://review.typo3.org/31312

#8 - 2014-07-07 20:36 - Bastian Waidelich

- Status changed from Under Review to Resolved
- % Done changed from 0 to 100

Applied in changeset commit:315f375362dd2f7964af756205e5cb08fd1f9763.

#9 - 2014-08-18 21:55 - Gerrit Code Review

- Status changed from Resolved to Under Review

Patch set 1 for branch **2.2** of project **Packages/TYPO3.Fluid** has been pushed to the review server.
It is available at http://review.typo3.org/32230

#10 - 2014-08-18 21:55 - Gerrit Code Review

Patch set 1 for branch **2.1** of project **Packages/TYPO3.Fluid** has been pushed to the review server.
It is available at http://review.typo3.org/32231

#11 - 2014-08-26 08:33 - Bastian Waidelich

- Status changed from Under Review to Resolved

Applied in changeset commit:9744e768fdab93cadf97fe0c3e8f523fddc95b14.