

## TYPO3.Flow - Bug # 42601

<b>Status:</b>	Under Review	<b>Priority:</b>	Must have
<b>Author:</b>	Robert Lemke	<b>Category:</b>	Security
<b>Created:</b>	2012-11-01	<b>Assigned To:</b>	Robert Lemke
<b>Updated:</b>	2013-08-14	<b>Due date:</b>	
<b>PHP Version:</b>	5.4		
<b>Has patch:</b>	No		
<b>Complexity:</b>	medium		
<b>Affected Flow version:</b>	Git master		
<b>Subject:</b>	Content Security: QOM rewriting is omitted if used in certain cases in an Action Controller		
<b>Description</b>	<p>The QOM Query Rewriting Aspect checks if the security context is initialized. If it is not yet initialized, it will suspend query rewriting and just proceed to call the execute() or count() method.</p> <p>This may be a problem because it is not defined when the security context is initialized. It can does happen that if no getRole() etc. methods have been called previously (no user is logged in), content is shown which must not be visible.</p> <p>This issue is, however, quite predictable and becomes apparent during development already.</p>		
<b>Related issues:</b>			
related to TYPO3.Flow - Bug # 42758: Unit test for PersistenceQueryRewritingA...	<b>Resolved</b>	<b>2012-11-07</b>	
related to TYPO3.Flow - Bug # 44765: Functional test broken	<b>Resolved</b>	<b>2013-01-23</b>	

### Associated revisions

#### Revision 9af3204b - 2012-11-01 19:58 - Robert Lemke

[BUGFIX] Enforce Query Rewriting more reliably

This initializes the security context if it hasn't been initialized when the PersistenceQueryRewritingAspect becomes active.

Previously it could happen that entities which should be covered by a policy are visible to anonymous users.

Change-Id: I44838de1503cbe49cf3fee51921b731bfaa0cfc5

Resolves: #42601

Releases: 1.1, 1.2

#### Revision ce08c301 - 2013-01-18 14:37 - Sebastian Kurfuerst

[BUGFIX] The security context is only allowed to be initialized after routing took place

This bugfix solves the root-cause for the following two symptoms:

- two logins needed in Neos until the Site is shown
- if the Flow\_Mvc\_Routing\_FindMatchResults cache is deactivated completely, the login does not work at all.

The problem is as follows:

- The security context needs the current **request** for working properly; such that it can separate the active and inactive tokens correctly in `\TYPO3\Flow\Security\Context::separateActiveAndInactiveTokens()`
- The current request is built during **routing**. Thus, the routing mechanism (f.e. `RoutePart` handlers) is not allowed to access the Security Context in any way. If it does (like in this example), things might break in various ways.
- For Neos, the following call chain takes place:
  - Routing
  - `FrontendNodeRoutePartHandler->matchValue` line 51
  - `NodeService->getNodeByContextNodePath()` line 57
  - `new ContentContext()` calls "initializeObject"
  - `ContentContext->initializeObject` does `$this->domainRepository->findByHost()`
  - this internally uses `Repository->findAll()`
  - this executes the `TYPO3\Flow\Security\Aspect\PersistenceQueryRewritingAspect->rewriteQomQuery`
  - because Neos has policy entries for entities (`TYPO3\TYPO3CR\Domain\Model\Node`), `$this->securityContext->initialize()` is called, WITHOUT HAVING A REQUEST SET BEFORE.
  - This results in a half- and wrongly-initialized Security Context set up, with `activeTokens` not properly set, and also only the standard roles assigned ("Everybody").
  - Thus, the check in `TYPO3\Neos\Controller\Frontend\NodeController->showAction()` fails: `$this->accessDecisionManager->decideOnResource('TYPO3_Neos_Backend_BackendController');`
  - This redirects the user back to the login (`$this->redirect('index', 'Login')`)
  - Now, if the routing cache is activated, the aspect kicks in (in the second iteration) and directly returns the match result, without triggering a database query before.

Thus, we need to enforce that the security context is not initialized during the routing phase.

The attached patch is just a quick fix; with not really the clean solution. But at least it works and the problem is properly described ;-)

This is a follow-up to issue #42601; where the according code has been implemented.

Change-Id: I724c1b352dd1807ba53b1e336f2d90e90360ff4d

Releases: master, 2.0

## Revision 9a3433d9 - 2013-01-23 11:08 - Karsten Dambekalns

[BUGFIX] Fix security-related unit test failures

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing unit tests.

Change-Id: Idff32284a4e8c4e935b7d7c132da769f5dd47140

Related: #42601

Releases: master

#### **Revision 55a312f2 - 2013-01-28 14:43 - Karsten Dambekalns**

[BUGFIX] Fix security-related functional test failures

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing functional tests.

It does that by:

- changing the order in which security is set up in the FunctionalTestCase provided by Flow.
- setting the "current request" again after a call to clearContext() in ContentSecurityTest
- adjusting the expected exception in MethodSecurityTest in two places

Change-Id: I353e2cba11473cf9ddef82f96b6a79d9d6fefbba

Fixes: #44765

Related: #42601

Releases: master

#### **Revision 56c6d852 - 2013-01-28 16:04 - Karsten Dambekalns**

[BUGFIX] Fix security-related functional test failures, part 2

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing functional tests.

It does that by:

- setting the "current request" again after a call to clearContext() in ContentSecurityTest
- adjusting the expected exception in MethodSecurityTest in two places

This is a followup to I353e2cba11473cf9ddef82f96b6a79d9d6fefbba which was broken after having fixed those already.

Change-Id: I09182972baf668abbb2cf583e8deebdc31e90205

Related: #42601

Fixes: #44765

Releases: master

#### **Revision ff2a4117 - 2013-04-05 10:58 - Sebastian Kurfuerst**

[BUGFIX] The security context is only allowed to be initialized after routing took place

This bugfix solves the root-cause for the following two symptoms:

- two logins needed in Neos until the Site is shown
- if the Flow\_Mvc\_Routing\_FindMatchResults cache is deactivated completely, the login does not work at all.

The problem is as follows:

- The security context needs the current **request** for working properly; such that it can separate the active and inactive tokens correctly in `\TYPO3\Flow\Security\Context::separateActiveAndInactiveTokens()`
- The current request is built during **routing**. Thus, the routing mechanism (f.e. RoutePart handlers) is not allowed to access the Security Context in any way. If it does (like in this example), things might break in various ways.
- For Neos, the following call chain takes place:
  - Routing
  - `FrontendNodeRoutePartHandler->matchValue` line 51
  - `NodeService->getNodeByContextNodePath()` line 57
  - `new ContentContext()` calls "initializeObject"
  - `ContentContext->initializeObject` does `$this->domainRepository->findByHost()`
  - this internally uses `Repository->findAll()`
  - this executes the `TYPO3\Flow\Security\Aspect\PersistenceQueryRewritingAspect->rewriteQomQuery`
  - because Neos has policy entries for entities (`TYPO3\TYPO3CR\Domain\Model\Node`), `$this->securityContext->initialize()` is called, WITHOUT HAVING A REQUEST SET BEFORE.
  - This results in a half- and wrongly-initialized Security Context set up, with `activeTokens` not properly set, and also only the standard roles assigned ("Everybody").
  - Thus, the check in `TYPO3\Neos\Controller\FrontendNodeController->showAction()` fails:  
`$this->accessDecisionManager->decideOnResource('TYPO3_Neos_Backend_BackendController');`
  - This redirects the user back to the login (`$this->redirect('index', 'Login')`)
  - Now, if the routing cache is activated, the aspect kicks in (in the second iteration) and directly returns the match result, without triggering a database query before.

Thus, we need to enforce that the security context is not initialized during the routing phase.

The attached patch is just a quick fix; with not really the clean solution. But at least it works and the problem is properly described ;-)

This is a follow-up to issue #42601; where the according code has been implemented.

Change-Id: I724c1b352dd1807ba53b1e336f2d90e90360ff4d

Releases: master, 2.0

#### Revision 26a56543 - 2013-04-05 10:58 - Karsten Dambekalns

[BUGFIX] Fix security-related functional test failures

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing functional tests.

It does that by:

- changing the order in which security is set up in the FunctionalTestCase provided by Flow.
- setting the "current request" again after a call to clearContext() in ContentSecurityTest
- adjusting the expected exception in MethodSecurityTest in two places

Change-Id: I353e2cba11473cf9ddef82f96b6a79d9d6fefbba

Fixes: #44765

Related: #42601

Releases: master, 2.0

#### **Revision 5612a561 - 2013-04-05 10:58 - Karsten Dambekalns**

[BUGFIX] Fix security-related functional test failures, part 2

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing functional tests.

It does that by:

- setting the "current request" again after a call to clearContext() in ContentSecurityTest
- adjusting the expected exception in MethodSecurityTest in two places

This is a followup to I353e2cba11473cf9ddef82f96b6a79d9d6fefbba which was broken after having fixed those already.

Change-Id: I09182972baf668abbb2cf583e8deebdc31e90205

Related: #42601

Fixes: #44765

Releases: master, 2.0

#### **Revision 18fa6d16 - 2013-04-05 10:58 - Karsten Dambekalns**

[BUGFIX] Fix security-related unit test failures

The change I724c1b352dd1807ba53b1e336f2d90e90360ff4d introduced some test failures. This change takes care of the failing unit tests.

Change-Id: Idf32284a4e8c4e935b7d7c132da769f5dd47140

Related: #42601

Releases: master, 2.0

## **History**

---

### **#1 - 2012-11-01 19:59 - Gerrit Code Review**

- *Status changed from Accepted to Under Review*

Patch set 1 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/16106>

**#2 - 2012-11-07 14:35 - Robert Lemke**

- Status changed from *Under Review* to *Resolved*

- % Done changed from 0 to 100

Applied in changeset commit:9af3204b3ceb08b488370d6d85802cac87821154.

**#3 - 2012-12-10 22:35 - Gerrit Code Review**

- Status changed from *Resolved* to *Under Review*

Patch set 1 for branch **FLOW3-1.1** has been pushed to the review server.

It is available at <https://review.typo3.org/17084>

**#4 - 2012-12-12 09:21 - Karsten Dambekalns**

- Target version changed from *2.0 beta 1* to *2.0*

**#5 - 2012-12-13 12:35 - Karsten Dambekalns**

- Affected Flow version changed from *Git 1.2 (master)* to *Git master*

**#6 - 2013-02-07 11:39 - Robert Lemke**

- Status changed from *Under Review* to *Resolved*

**#7 - 2013-04-05 16:26 - Gerrit Code Review**

- Status changed from *Resolved* to *Under Review*

Patch set 2 for branch **FLOW3-1.1** has been pushed to the review server.

It is available at <https://review.typo3.org/17084>

**#8 - 2013-08-14 15:35 - Karsten Dambekalns**

- Target version changed from *2.0* to *2.0.1*