TYPO3.Flow - Bug # 47995

Status:	Closed	**Priority:**	Should have
Author:	Dominique Feyer	**Category:**	
Created:	2013-05-06	**Assigned To:**	Dominique Feyer
Updated:	2013-05-08	**Due date:**	

PHP Version:	
Has patch:	Yes
Complexity:	
Affected Flow version: Git 2.0	
Subject:	The method hasRole() in TYPO3/Flow/Security/Account is currently broken
Description	

Currently this code always return FALSE:

    $account->hasRole(new \TYPO3\Flow\Security\Policy\Role('Your.Package:Role')

The hasRole method use the contains method from Doctrine ArrayCollection. Has the two object are different (even is they have the same identifier), the method return FALSE.

I propose to change this method by the same feature found in the SecurityContext hasRole method (who compore the identifer, and the full object).

History

#1 - 2013-05-06 21:00 - Gerrit Code Review

- Status changed from New to Under Review

Patch set 1 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20555

#2 - 2013-05-06 21:04 - Gerrit Code Review

Patch set 2 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20555

#3 - 2013-05-07 09:48 - Gerrit Code Review

Patch set 3 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20555

#4 - 2013-05-07 10:33 - Bastian Waidelich

- Status changed from Under Review to Needs Feedback

The actual "issue" is that roles are entities since #28862
Account::hasRole() won't be the only place where this breaks compatibility.

For instance:

```
1 $account->addRole(new \TYPO3\Flow\Security\Policy\Role('Your.Package:Role');
```

won't work either - as well as Account::setRoles(), Account::removeRole() and so on.
The way to work around this is not to create new instances but fetch existing roles from the PolicyService:

```
1 $role = $policyService->getRole('Your.Package:Role');
2 $account->hasRole($role);
```

IMO we need to document this before the release an add **visible** pointers to the documentation in the release notes

Note: with ValueObjects this might be less breaking but they're currently not properly supported AFAIR

#5 - 2013-05-07 11:10 - Christian Müller

We could try to use VO here. That should solve the mentioned problem. At least for doctrine our implementation should do what is needed.

#6 - 2013-05-07 11:26 - Karsten Dambekalns

One thing we could discuss is what methods should (alternatively) be able to take a string (role identifier). But then again, **checking for roles is not really a clever thing in most cases.**

Why? **Because it ignores the fact that roles might get reconfigured permissions.** So if you want to check for access to something, you should rather check if permission to access things is granted, not if a certain role is assigned!

#7 - 2013-05-07 11:30 - Bastian Waidelich

Karsten Dambekalns wrote:

> *if you want to check for access to something, you should rather check if permission to access things is granted, not if a certain role is assigned!*

I agree!
Unfortunately the API for this is not very usable yet (at least some convenience method would do good). Currently I have to add this snippet in my base controllers:

```
1 /**
2  * Check if we currently have access to the given resource
3  *
4  * @param string $resource The resource to check
5  * @return boolean TRUE if we currently have access to the given resource
6  */
7 protected function hasAccessToResource($resource) {
```

```
 8   try {
 9       $this->accessDecisionManager->decideOnResource($resource);
10   } catch (AccessDeniedException $e) {
11       return FALSE;
12   }
13   return TRUE;
14}
```

#8 - 2013-05-07 12:19 - Karsten Dambekalns

Bastian Waidelich wrote:

> *Unfortunately the API for this is not very usable yet (at least some convenience method would do good). Currently I have to add this snippet in my*
> *base controllers:*
> *[...]*

Full Ack. What about adding that to some clever place?

#9 - 2013-05-07 16:43 - Dominique Feyer

Thanks for all those information, I update my application based on it, and everything work fine.

I need to spend more time on the decisionMaker, but currently my role configuration are really simple so I can rely on simple hasRole logic.

All the information in this issue will have a better visibility in the official documentation I think. The change around Role can be frustrating without that.

#10 - 2013-05-08 12:33 - Bastian Waidelich
- Status changed from Needs Feedback to Closed

Thanks for the feedback Dominique,

We'll definitely have to add a comprehensive "breaking changes" section to documentation & release notes with code examples and such.