

Arvados - Bug #10557

Perform full user setup procedure when (incl. can_read permission link to all_users) when a user is activated by updating the is_active flag.

11/18/2016 01:19 PM - Nico César

Status:	Resolved	Start date:	11/18/2016
Priority:	Normal	Due date:	
Assigned To:	Tom Clegg	% Done:	50%
Category:		Estimated time:	0.00 hour
Target version:	2017-06-21 sprint		
Description			
<p>Currently, the setup button (in the Admin tab that gives login permissions on shell nodes) is responsible for creating the can_read link permission to all_users. The link is also added when the user is created if the "auto setup new user" feature is enabled.</p> <p>If auto-setup is disabled, and the admin doesn't want to set up VM access (e.g., external LDAP and other tools provide the users), the admin might activate users by changing the is_active flag to true instead of using the "Setup" button. This nearly works, except that the "all users" link isn't created, and sharing features in Workbench don't work as expected for those users.</p> <p>We should support this admin workflow by adding the appropriate link (and doing any other needed setup activities) if needed when a user's is_active flag changes to true.</p>			
Implementation			
Run auto_setup_new_user as an after_update hook when the is_active flag changes to true.			
Subtasks:			
Task # 11833: Review 10557-auto-setup			Resolved
Task # 11873: Review 10557-setup-cleanup			Resolved
Related issues:			
Related to Arvados - Bug #12178: Activating an account (setting is_active fro...		New	08/25/2017

Associated revisions

Revision 021d36f1 - 06/15/2017 04:11 PM - Tom Clegg

Merge branch '10557-auto-setup'

refs #10557

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tom@curoverse.com>

Revision b1f94004 - 06/20/2017 08:05 PM - Tom Clegg

Merge branch '10557-setup-cleanup'

refs #10557

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tom@curoverse.com>

History

#1 - 11/18/2016 01:19 PM - Nico César

- Project changed from Arvados Private to Arvados

#2 - 11/21/2016 04:16 PM - Tom Clegg

The link should be created when the user is activated (not created). I think we can make this (and any other user-activation tasks?) happen automatically whenever is_active changes to true, so it's not as easy for an admin to make a "half-setup" user.

#3 - 11/21/2016 08:46 PM - Tom Clegg

- Subject changed from Create can_read permission link to all_users when user is created. to Perform full user setup procedure when (incl. can_read permission link to all_users) when a user is activated by updating the is_active flag.

#4 - 01/03/2017 05:36 PM - Tom Clegg

- Description updated

#5 - 06/07/2017 04:43 PM - Tom Morris

- Target version set to 2017-06-21 sprint

#6 - 06/07/2017 07:16 PM - Tom Clegg

- Assigned To set to Tom Clegg

#7 - 06/13/2017 05:41 PM - Tom Clegg

- Status changed from New to In Progress

#8 - 06/14/2017 01:05 AM - Tom Clegg

10557-auto-setup @ [2b62223c9ba420208b9f293825e7f6ae3f50f95b](#)

#9 - 06/14/2017 03:25 PM - Peter Amstutz

Is the belt-and-suspenders check strictly necessary? It does this:

```
:if => Proc.new { |user|
  ![system_user_uuid, anonymous_user_uuid].include?(user.uuid)
}
```

and then does this:

```
return if [system_user_uuid, anonymous_user_uuid].include?(self.uuid)
```

Why does the setup method both set @object (initially nil, then set to a User object) and but also set the object_found flag? Seems like the only case that object_found is false is when the object is newly created? Which can be detected with new_record? I think?

#10 - 06/14/2017 03:37 PM - Tom Clegg

Peter Amstutz wrote:

Is the belt-and-suspenders check strictly necessary?

No, I just added one without removing the other. Fixed.

Why does the setup method both set @object (initially nil, then set to a User object) and but also set the object_found flag? Seems like the only case that object_found is false is when the object is newly created? Which can be detected with new_record? I think?

AFAICT "object_found" means "user already existed before setup() was called", yes. I don't think new_record? would be true any more after create!(). Anyway, I was mainly trying to reduce the two setup methods to one, so I thought I'd leave well enough alone here.

10557-auto-setup @ [eb0012d203974e54023dfcac6e04fd4c2c40270f](#)

#11 - 06/14/2017 03:45 PM - Peter Amstutz

Tom Clegg wrote:

AFAICT "object_found" means "user already existed before setup() was called", yes. I don't think new_record? would be true any more after create!(). Anyway, I was mainly trying to reduce the two setup methods to one, so I thought I'd leave well enough alone here.

Right, you removed one branch based on object_found. There's another branch based on object_found but I don't quite understand why it makes the distinction.

10557-auto-setup @ [eb0012d203974e54023dfcac6e04fd4c2c40270f](#)

#12 - 06/14/2017 07:31 PM - Tom Clegg

Peter Amstutz wrote:

Right, you removed one branch based on object_found. There's another branch based on object_found but I don't quite understand why it makes the distinction.

Looks like in the new user case, "username/" gets prepended to the given repo name even if it already starts with "username/". Kinda sounds like a bug. Seems like we could use more cleanup in this area?

#13 - 06/20/2017 02:33 PM - Tom Clegg

10557-setup-cleanup @ [74b3ad1f061185ca695e8bbead723b5212bbb06a](#)

#14 - 06/20/2017 05:51 PM - Peter Amstutz

Get out your whip and gun, its time for some code archeology.

A few things I found confusing:

- The setup method can be used to both create a new user and set up an existing user. That should probably be more clearly reflected in a code comment.
- full_repo_name can be nil, which means that no repo will be assigned or created, but this doesn't get checked until User.create_user_repo_link. A comment might make it a little more obvious what is going on.
- I could run the setup method with {user => "peter", repo_name => "tomclegg/secrets"} and it'll happily give me manage permission to that repository. I don't think it's a security hole because only admins can do setup, but this behavior intended/desirable? Seems like it mostly provides an opportunity for the wrong repository to be assigned by mistake.
- Could UserNotifier.account_is_setup go in the setup method of the model, instead the controller?

#15 - 06/20/2017 07:28 PM - Tom Clegg

Peter Amstutz wrote:

Get out your whip and gun, its time for some code archeology.

Is it, though? ;)

A few things I found confusing:

- The setup method can be used to both create a new user and set up an existing user. That should probably be more clearly reflected in a code comment.
- full_repo_name can be nil, which means that no repo will be assigned or created, but this doesn't get checked until User.create_user_repo_link. A comment might make it a little more obvious what is going on.

OK, I'm now un-volunteering to do further code cleanup in this area as part of this bugfix.

- I could run the setup method with {user => "peter", repo_name => "tomclegg/secrets"} and it'll happily give me manage permission to that repository. I don't think it's a security hole because only admins can do setup, but this behavior intended/desirable? Seems like it mostly provides an opportunity for the wrong repository to be assigned by mistake.

To me, the admin's intent seems pretty clear, and handling such a request by adding a "peter/tomclegg/secrets" repo would seem surprising/wrong.

Normally the repo name would be given as just "secrets", of course.

- Could UserNotifier.account_is_setup go in the setup method of the model, instead the controller?

It could, but I don't think there's any particular benefit: there's no way for the "update" call (or the UI feature we're fixing here) to set the "want notification" flag.

#16 - 06/20/2017 07:47 PM - Peter Amstutz

ok, merge it!

#17 - 06/20/2017 08:32 PM - Tom Clegg

- Status changed from In Progress to Resolved