Arvados - Feature #16571

Permission system supports seeing & sharing with a group without having access to group contents.

06/29/2020 02:33 PM - Peter Amstutz

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assigned To:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

Customer has curators who are not admins who need to share data they manage with other groups. They need to be able to see those groups to select them without being able to see the other group's contents.

It already works this way for Users (you can see a user without gaining access to the things the user owns) but there isn't a way to achieve this behavior for groups

Some ideas:

- Workaround: create a fake user, grant can_manage, people share with the fake user
- New group_class that has the desired behavior
- "view" permission with new semantics (can view group record but follow any of its links)
- permission links specify separate permission levels for record and traversal: can read/write/manage record, can gain read/write/manage by traversing record
- "can_use_permissions" and "can_list_members" with new semantics #15372

Related issues:

Related to Arvados - Story #15372: Revise group permissions to separate them ...	**New**		
Related to Arvados Epics - Story #16445: Expand permission system	**New**	**09/01/2021**	**11/30/2021**

History

#1 - 06/29/2020 02:36 PM - Peter Amstutz

- Description updated

#2 - 06/29/2020 03:15 PM - Peter Amstutz

- Description updated

#3 - 06/29/2020 03:52 PM - Peter Amstutz

- Related to Story #15372: Revise group permissions to separate them from permissions on managed objects added

#4 - 06/29/2020 05:03 PM - Peter Amstutz

- Related to Story #16445: Expand permission system added

#5 - 06/30/2020 02:05 PM - Tom Clegg

I think [a subset of] #15372#note-19 would address this. "can_use_permissions" would function the way "can_manage" does now, and "can_read" would do what this customer is looking for, i.e., just read the target group.

We could also add "can_use_read_permissions" (to accomplish what A -can_read-> B -can_write-> C does now) but I'm not sure whether it's an important case.

> permission links specify separate permission levels for record and traversal

#15372 has stuff like "can_use_permissions + can_list_members" (implying can_read but not can_manage) but doesn't say whether that should happen in a single permission link, or multiple links.

Encoding permissions as arbitrary bitmaps instead of predefined constants might make the implementation more efficient, and would enable people to use weird combinations that we haven't named, like "can use target's permissions, but can't see the target itself".