

Pulp - Issue #4740

Pulpcore doesn't provide a way to guarantee uniqueness in repo versions

04/25/2019 10:32 PM - daviddavis

Status: CLOSED - DUPLICATE	Start date:
Priority: Normal	Due date:
Assignee:	Estimated time: 0:00 hour
Category:	
Sprint/Milestone: 3.0.0	
Severity: 3. High	Groomed: No
Version:	Sprint Candidate: No
Platform Release:	Tags:
OS:	Sprint:
Triaged: No	Quarter:
Description	
Suppose you have two units of content that aren't unique. This could be two packages with the same NEVRA (one signed, one unsigned or two different checksums), two errata with the same id, two files with the same filename, two docker tags with the same name, etc. Let's say that their ids are \$CU1 and \$CU2. Now you call:	
<pre>http POST ':24817'\$REPO_HREF'versions/' add_content_units:["\"\$CU1\", \"\$CU2\"]"</pre>	
This causes conflicting content units to be added to a repo version. The problem here is that pulpcore doesn't provide plugin writers with a way to prevent this and ensure uniqueness for repo version content.	
Related issues:	
Is duplicate of Pulp - Issue #3541: Core should not add/remove content to a...	CLOSED - CURRENTRELEASE

History

#1 - 04/25/2019 10:33 PM - daviddavis

- Description updated

#2 - 04/25/2019 10:33 PM - daviddavis

- Blocks Task #4028: Prevent duplicate files in repositories added

#3 - 04/25/2019 10:36 PM - daviddavis

- Sprint/Milestone set to 3.0.0

#4 - 04/25/2019 10:56 PM - amacdona@redhat.com

This was part of the intended use of the RemoveDuplicates stage. Can you specify why that doesn't work in this case?

<https://pulp.plan.io/issues/3934>

#5 - 04/25/2019 11:05 PM - daviddavis

This is just adding units to a repo (or repo version) not syncing. Here's an example from pulp_file:

```
http POST http://localhost:24817/pulp/api/v3/repositories/ name=foo
export REPO_HREF=$(http :24817/pulp/api/v3/repositories/ | jq -r '.results[] | select(.name == "foo") | ._href
')
echo "blah1" > 1.iso
echo "blah2" > 2.iso
export ART1=$(http --form POST http://localhost:24817/pulp/api/v3/artifacts/ file@./1.iso | jq -r '._href')
export ART2=$(http --form POST http://localhost:24817/pulp/api/v3/artifacts/ file@./2.iso | jq -r '._href')
export CU1=$(http POST http://localhost:24817/pulp/api/v3/content/file/files/ relative_path=1.iso _artifact="$
ART1" | jq -r '._href')
export CU2=$(http POST http://localhost:24817/pulp/api/v3/content/file/files/ relative_path=1.iso _artifact="$
ART2" | jq -r '._href')
http POST ':24817'$REPO_HREF'versions/' add_content_units:["\"$CU1\", \"$CU2\"]"
sleep 5
http ":24817${REPO_HREF}versions/1/"
```

This creates a repository with two conflicting files (ie two files with the same name).

#6 - 04/26/2019 12:39 AM - bmbouter

This seems similar to the story from <https://pulp.plan.io/issues/3541>

We want core to provide generic functionality yet only plugins can know and 'validate' the repo. I was pitching the hooking option about a year ago which has the nice default of having plugin writers take no action if they don't require extra validation and a clear mechanism they can provide if they do. <https://pulp.plan.io/issues/3541?pn=1#note-3>

#7 - 04/26/2019 12:42 AM - daviddavis

- Blocks deleted (Task #4028: Prevent duplicate files in repositories)

#8 - 04/26/2019 12:43 AM - daviddavis

- Status changed from NEW to CLOSED - DUPLICATE

#9 - 04/26/2019 12:43 AM - daviddavis

- Is duplicate of Issue #3541: Core should not add/remove content to a repository or create a repository_version without plugin input added