Pulp - Story #6737

Story # 6134 (CLOSED - CURRENTRELEASE): [EPIC] Pulp import/export

As a user, I can import a split export

05/14/2020 04:53 PM - daviddavis

Status:	CLOSED - CURRENTRELEASE	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	ggainey	**% Done:**	100%
Category:		**Estimated time:**	0:00 hour
Sprint/Milestone:	3.6.0		
Platform Release:		**Tags:**	
Groomed:	No	**Sprint:**	Sprint 77
Sprint Candidate:	No	**Quarter:**	

Description

Export files can be split. This needs to handle this case.

This is basically the import version of https://pulp.plan.io/issues/6736

Related issues:

Related to Pulp - Story #6736: As a user, I can export into a series of files...	**CLOSED - CURRENTRELEASE**

Associated revisions

Revision 40f7cc3b - 07/24/2020 10:10 PM - ggainey

Taught export to produce, and import to understand, a table-of-contents (toc) file.

Emitted 'next to' the export file or files, named -toc.json.

Consists of keys "meta" and "files". "files" is a dictionary of export-file/checksums. "meta" contains the "file", "chunk_size", and "global_hash" of the export.

Added toc= to import. Import will find and validate the checksums of any chunk_files, reassemble them into a single .tar.gz, and pass that along to the rest of the import process. Deletes chunks as it goes, to minimize disk-space.

Updated import-export docs to describe TOC file and its use.

closes #6737

History

#1 - 05/14/2020 04:53 PM - daviddavis

- Related to Story #6736: As a user, I can export into a series of files of a particular size added

#2 - 05/24/2020 02:42 PM - ggainey

- Status changed from NEW to ASSIGNED

- Assignee set to ggainey

- Sprint set to Sprint 73

#3 - 05/26/2020 10:37 PM - ggainey

We won't be able to do the subprocess-streaming trick we did for PulpExport here with a chunked-export - import needs random-access 'into' the export-tarfile, and you can't have that *and* stream from a subprocess. We could:

1. take the import-filename and look for chunks of the form .dddd. If found, recreate the tarfile, and and process as normal
2. require the import-caller to recreate the tarfile before calling import
3. add a param to import "chunk_list", which would be the output of the export.output_file_info field. This would let us do a), above, while also checking checksums of each chunk for integrity

There is (iirc) a 'clever' trick to use 'dd' to recreate the .tar.gz that would never use more disk than <tar.gz full size> + 1 'chunk' size. Regardless of whether pulp or its caller is responsible for recreating the tar.gz should investigate using this approach to minimize disk requirements.

#4 - 05/29/2020 02:32 PM - rchan

- Sprint changed from Sprint 73 to Sprint 74

#5 - 06/11/2020 10:27 PM - rchan

- Sprint changed from Sprint 74 to Sprint 75

#6 - 06/26/2020 06:04 PM - rchan

- Sprint changed from Sprint 75 to Sprint 76

#7 - 07/04/2020 08:58 PM - ggainey

ggainey wrote:

> We won't be able to do the subprocess-streaming trick we did for PulpExport here with a chunked-export - import needs random-access 'into' the export-tarfile, and you can't have that *and* stream from a subprocess. We could:
>
> 1. take the import-filename and look for chunks of the form .dddd. If found, recreate the tarfile, and and process as normal

Requires way too much implied-magic, and doesn't let us vet the 'chunks' we're trying to recombine.

> 1. require the import-caller to recreate the tarfile before calling import

This is what we get if we do nothing and is available right now. Implementing the functionality described in 3. will not break this option.

> 1. add a param to import "chunk_list", which would be the output of the export.output_file_info field. This would let us do a), above, while also checking checksums of each chunk for integrity

This is the best option, as it requires an implicit request from the user, allows us to find the chunks and check their validity (via checksum), and lets us do the 'minimal extra filespace used' trick without requiring the pulp-user to know how to do Magic With DD.

Implementing a chunks= on import is mutually exclusive with filename= - either you have the whole file, or you have a file with the output of export.output_file_info, whic is expected to be in the 'same place' as the chunks it describes.

#8 - 07/10/2020 08:32 PM - rchan

- Sprint changed from Sprint 76 to Sprint 77

#9 - 07/20/2020 02:39 AM - pulpbot

- Status changed from ASSIGNED to POST

PR: https://github.com/pulp/pulpcore/pull/794

#10 - 07/24/2020 10:11 PM - ggainey

- Status changed from POST to MODIFIED

- % Done changed from 0 to 100

Applied in changeset pulpcore|40f7cc3bb28830d9944a9908f971ea7002702b16.

#11 - 08/13/2020 10:20 PM - dkliban@redhat.com

- Sprint/Milestone set to 3.6.0

#12 - 08/13/2020 11:38 PM - pulpbot

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE