Pulp - Story #4188

As a Pulp3 user, I have containers

11/26/2018 10:16 PM - bmbouter

Status:	CLOSED - CURRENTRELEASE		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	29%
Category:			**Estimated time:**	0:00 hour
Sprint/Milestone:	3.0.0			
Platform Release:			**Tags:**	
Groomed:	No		**Sprint:**	
Sprint Candidate:	No		**Quarter:**	

Description

Motivation

Some Pulp users likely want to run Pulp processes inside containers. We want these containers pre-built and available in the significant registries so that users can easily run Pulp from there.

There are a few flavors of containers that people want. Specifically 3 groups:

Core Containers. These containers (1 per Pulp process) only have pulpcore and pulpcore-plugin installed on them. This is useful for anyone who wants to build a derivative container, or have a vanilla core container without any plugins installed yet.

End User Single-Plugin Containers. These containers contain 1 plugin, e.g. pulp_rpm or pulp_docker. It's a container offering from a plugin developer to their end user as an easy way to run that plugin.

Specialty Plugin-Mix Containers. There will likely be several of these mixture containers made and maintained over time. For example we may want a single container with all known pulp plugins installed in it for compatability testing purposes. Also we'll want a plugin specifically for large stakeholders like Katello who uses a specific set of plugins.

What OS?

The images will be CentOS7. CentOS 7 images that comes pre-bundled with python 3.6 are available and pre-built from the CentOS group.

How Many Processes per Container?

Container best-practices suggest we only run 1 process per container

Is each Pulp process, i.e. pulp_resource_manager, pulp_worker, pulp_webserver its own container?

Yes. This makes it easy for end users who only have to specify a name instead of a name and environment variables to give that container its personality.

Which registries?

The primary registry Pulp publishes containers to is quay.io. It would also be nice to be on dockerhub additionally. More registries would also be OK beyond these.

What about Redis and Postgresql?

We need to find where those are maintained on the registries also.

Who is building these images?

Travis will be doing the container building, testing, and publishing. These will be additional steps in existing pipelines and new pipelines in Travis. For example, the same Travis pipeline that publishes a Pulp release to PyPI, will also build the container and pulp-smash test it before publishing the container to the registry.

Who is maintaining the Travis automation?

The core team maintains the images that do not include any plugins.
Each plugin team can choose to add Travis build/test/publish workflows for their plugins and can maintain that.
For new plugins, the plugin_template should come with as much boilerplate container build/test/publish code as possible

What about specialty plugin mixtures of containers?

For specialty build mixtures, Pulp can produce build files, e.g. a Dockerfile that would allow a public build service like dockerhub to build an image for you to your needs. This could be a check-the-plugins-you-want webform in the docs or on pulpproject.org for example.

Subtasks:	
Task # 4189: Extend Travis build+publish pipeline to build, test, and publish vanilla c...	**CLOSED - WONTFIX**
Story # 4198: As a plugin writer, the plugin_template has a container-build-test pipeli...	**CLOSED - WONTFIX**
Story # 4199: As an Ansible user, I have containers	**NEW**
Story # 4200: As a Python user, I have containers	**NEW**
Story # 4201: As a File user, I have containers	**NEW**
Story # 4202: As a Docker user, I have containers	**NEW**
Story # 4203: As an RPM user, I have containers	**NEW**

Associated revisions

Revision 764dfc36 - 04/01/2019 07:36 PM - Eric D. Helms

fixes #4188: Add Pulp 3 container image built from PyPi

Revision 7012b62a - 04/17/2019 11:57 PM - bmbouter

Merge pull request #13 from ehelms/add-dockerfile

Fixes #4188: Add Pulp 3 container image built from PyPi

History

#1 - 11/27/2018 05:15 PM - amacdona@redhat.com

For arbitrary plugin mixtures, one alternative to generating Dockerfiles could be to use an init container[0].

We could build containers with pulpcore (without plugins), and use the init container to install plugins. We would have to mount the site-packages and modify the python path so that the plugin container and the pulpcore container know about each other and install dependencies correctly.

[0]: https://kubernetes.io/docs/concepts/workloads/pods/init-containers/

#2 - 11/27/2018 10:57 PM - bmbouter

- Tracker changed from Issue to Story

- % Done set to 0

#3 - 04/17/2019 11:57 PM - Anonymous

- Status changed from NEW to MODIFIED

- % Done changed from 0 to 100

Applied in changeset pulpcore|764dfc365128a9fcb3d2d143a7d33a3a378f7e92.

#4 - 04/25/2019 06:44 PM - daviddavis

- Sprint/Milestone set to 3.0.0

#5 - 04/26/2019 10:33 PM - bmbouter

- Tags deleted (Pulp 3)

#6 - 12/13/2019 06:26 PM - bmbouter

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE