Pulp - Task #2323

Support skipping the standard storage path migration.

10/05/2016 05:02 PM - jortel@redhat.com

Status:	CLOSED - WONTFIX		**Start date:**	
Priority:	High		**Due date:**	
Assignee:			**% Done:**	0%
Category:			**Estimated time:**	0:00 hour
Sprint/Milestone:				
Platform Release:			**Tags:**	Pulp 2
Groomed:	Yes		**Sprint:**	Sprint 10
Sprint Candidate:	Yes		**Quarter:**	

Description

In 2.8, pulp standardized how the platform determines where a content unit is stored under /var/lib/pulp/content. In the past, each plugin determined this on its own so it varied. To achieve consistency, a migration was added in 2.8 that recalculated the storage path for all content units created before the upgrade. Then, moved the file(s) and recreated the published symlinks. The migration was tuned for maximum performance against an installation containing 500k units and 1.5 million symlinks. However, as it turns out, there are larger customer with an estimated 5-10 million symlinks. To make matters worse, both the content and links are on an NFS share. The result is customer reports of the upgrades (migration) taking days and weeks.

It's important to note that pulp does not need this migration for correctness. The content unit storage path is stored in the DB and having a combination of old and new style storage paths does not cause problems. However, there is a use case that needs to be supported. Mainly for developers, sales and support users, not customers. That is: "As a user, I want to restore /var/lib/pulp/content from an archive so I don't need to re-download content ."

The proposal is to provide a way to disable (skip) this migration in Satellite installations. It's not needed for correctness and Satellite customers don't care about the restore use case.

But what about upstream pulp?

We have a few options here:

Option [#1](#):

Disable (skip) the migration based on installation environment. The default would be to perform the migration. Not sure how to detect this. Could use an environment variable but not sure how it would get set in all cases. Sometimes admins run pulp-manage-db directly. Perhaps something else? Something that already exists? Something persistent?

Pros:

- Least code changed and migration would happen as part of upgrade when desired.

Cons:

- How to detect when the migration is to be skipped.
- Can't migrate later (after upgrade) if you change your mind.
- This may require downstream work.

Option [#2](#):

Remove the migration code all together and provide a script to perform the migration after the upgrade as needed to support the use case.

Pros:

- Cleaner and less gimmicky.
- Admin can skip the migration to expedite upgrade and then do the migration later when system is idle. Though, ensuring the system is idle would be hard.
- Separate script provides better opportunity for good progress reporting and users could run the migration is small batches.

Cons:

- More development needed.
- Admins will likely not run the migration voluntarily unless they need/want to support the restore use case. This does not advance the pulp team's desire for broad /var/lib/pulp/content layout consistency. And there is value in this consistency.

Other options?

Thoughts? Comments?

History

#1 - 10/05/2016 05:04 PM - jortel@redhat.com

- Description updated

- Status changed from NEW to ASSIGNED

- Assignee set to jortel@redhat.com

#2 - 10/05/2016 06:15 PM - bmbouter

For option 1, we could have pulp-manage-db take a special script that causes the migration to be a no-op.

We should not do option 2 because we want all pulp users to use the new filesystem layout. A separate script should be published but we should not remove the migration. So in other words we should do option 1 and 2 with the adjustment that option 2 not remove the migration code.

This story needs more discussion before it leaves the NEW state. We need a plan before work begins.

#4 - 10/05/2016 07:46 PM - jortel@redhat.com

- Status changed from ASSIGNED to NEW

- Assignee deleted (jortel@redhat.com)

#5 - 10/05/2016 07:52 PM - jortel@redhat.com

- Sprint/Milestone deleted (27)

#6 - 10/05/2016 08:01 PM - jortel@redhat.com

I like the possibility this suggest brings. Can you elaborate on what this option might look like for pulp-manage-db? Each plugin has this migration so they would all (4-5) likely need to be skipped for most satellite customers. The penalty for not including the skip option could be very high. When run with satellite tooling, it would automated but worried about cases where sat admins run pulp-manage-db directly.

#7 - 10/05/2016 08:05 PM - bmbouter

It could be called like this:

```
pulp-manage-db --skip-filesystem-migration
```

It would skip any of the associated filesystem layout migrations depending on which ones are installed.

#9 - 10/05/2016 08:08 PM - mhrivnak

One option is to have pulp-manage-db read a file at startup that would contain a list of migrations to skip. That's generic and would let us guarantee that users won't forget to use a specific option if they run it manually.

#10 - 10/05/2016 08:12 PM - bmbouter

mhrivnak wrote:

> One option is to have pulp-manage-db read a file at startup that would contain a list of migrations to skip. That's generic and would let us guarantee that users won't forget to use a specific option if they run it manually.

That sounds better than what I proposed. Since the 2.y migration system only records the most recent migration level running it once would cause all future runs to also skip. Right?

#11 - 10/05/2016 08:32 PM - jortel@redhat.com

mhrivnak wrote:

> One option is to have pulp-manage-db read a file at startup that would contain a list of migrations to skip. That's generic and would let us guarantee that users won't forget to use a specific option if they run it manually.

I like this option. It's generic from the upstream perspective and solves the problem. We'd need to document appropriately to ensure it's **never** used by end users. The potential for abuse is high. Imagining something like /etc/pulp/migration/blacklist.txt that contains a list of migrations to no-op.

#14 - 10/06/2016 12:08 AM - mmccune@redhat.com

- *Version set to Master*

#18 - 10/07/2016 05:01 PM - amacdona@redhat.com

- *Tracker changed from Issue to Task*

- *Sprint/Milestone set to 27*

- *% Done set to 0*

- *Groomed changed from No to Yes*

#20 - 10/12/2016 06:01 PM - jortel@redhat.com

Considering the following minimal patch for these reasons:

- The most minimum code changes needed to accomplish the goal. This minimizes possibility of merge conflicts applying the patch.
- Targeting only the migrations causing the pain.
- Some of the migrations include fixes to regressions when converting to mongoengine.

```
[jortel@f23d pulp_rpm]$ git diff -p
diff --git a/plugins/pulp_rpm/plugins/migrations/0028_standard_storage_path.py b/plugins/pulp_rpm/plugins/migr
ations/0028_standard_storage_path.py
index 08049ec..327428c 100644
```

```
--- a/plugins/pulp_rpm/plugins/migrations/0028_standard_storage_path.py
+++ b/plugins/pulp_rpm/plugins/migrations/0028_standard_storage_path.py
@@ -23,11 +23,11 @@ def migrate(*args, **kwargs):
     _logger.info(stars)

     migration = Migration()
-    migration.add(rpm_plan())
-    migration.add(srpm_plan())
-    migration.add(drpm_plan())
-    migration.add(YumMetadataFile())
-    migration.add(DistributionPlan())
+    # migration.add(rpm_plan())
+    # migration.add(srpm_plan())
+    # migration.add(drpm_plan())
+    # migration.add(YumMetadataFile())
+    # migration.add(DistributionPlan())
     migration.add(ISO())
     migration()
```

#21 - 10/12/2016 06:35 PM - jortel@redhat.com

jortel@redhat.com wrote:

> Considering the following minimal patch for these reasons:
>
> - The most minimum code changes needed to accomplish the goal. This minimizes possibility of merge conflicts applying the patch.
> - Targeting only the migrations causing the pain.
> - Some of the migrations include fixes to regressions when converting to mongoengine.
>
> [...]

After further consideration, this really could be patched as a noop in the platform instead. Any of the corrections made in the plugin migrations would only fix issues with early 2.8.0 betas. Satellite 6.2 upgrades would only involve later matured pulp 2.8 that did not create units with these issues. Including another patch shortly.

#22 - 10/12/2016 09:19 PM - jortel@redhat.com

Considering this for a patch:

```
diff --git a/pulp/plugins/migration/standard_storage_path.py b/pulp/plugins/migration/standard_storage_path.py
index f651e71..90f814e 100644
--- a/pulp/plugins/migration/standard_storage_path.py
+++ b/pulp/plugins/migration/standard_storage_path.py
@@ -392,6 +392,11 @@ class Migration(object):
            1. Batch the migration.
            2. Delete empty directories created by the migration.
         """
+
+        # Re: Bug:1381702
+        _log.info('Path migration skipped.')
+        return
+
         batch = Batch()
         for unit in chain(*self.plans):
             batch.add(unit)
```

#23 - 10/14/2016 10:30 PM - jortel@redhat.com

- File storage-migration.patch added

The spec file change was tested with the attached patch.

```
[jortel@localhost 2323]$ diff -ru ~/git/pulp/pulp.spec rpmbuild/SPECS/pulp.spec
--- /home/jortel/git/pulp/pulp.spec     2016-10-14 15:27:39.228841171 -0500
+++ rpmbuild/SPECS/pulp.spec     2016-09-23 13:30:25.000000000 -0500
@@ -35,7 +35,7 @@

 Name: pulp
 Version: 2.8.7
-Release: 1%{?dist}
+Release: 2%{?dist}
 Summary: An application for managing software content
 Group: Development/Languages
 License: GPLv2
@@ -54,12 +54,17 @@
 %endif
 BuildRequires: rpm-python

+Patch0: storage-migration.patch
+
 %description
 Pulp provides replication, access, and accounting for software repositories.

 %prep
+
 %setup -q

+%patch0 -p1
+
 %build
 for directory in agent bindings client_consumer client_lib common devel
 do
```

#25 - 10/24/2016 04:29 PM - jortel@redhat.com

- Sprint/Milestone changed from 27 to 28

#26 - 10/28/2016 04:06 PM - jortel@redhat.com

- Status changed from NEW to CLOSED - WONTFIX

After much discussion, it was determined that this is both not an upstream concern and too late to benefit upstream users.

#27 - 03/08/2018 08:12 PM - bmbouter

- Sprint set to Sprint 10

#28 - 03/08/2018 08:13 PM - bmbouter

- Sprint/Milestone deleted (28)

#29 - 04/15/2019 10:24 PM - bmbouter

- Tags Pulp 2 added

Files

storage-migration.patch	1019 Bytes	10/14/2016	jortel@redhat.com