

Pulp - Issue #8352

Possible race condition in reserved resources

03/05/2021 03:16 PM - wibbit

Status:	CLOSED - CURRENTRELEASE	Start date:	
Priority:	Normal	Due date:	
Assignee:		Estimated time:	0:00 hour
Category:		Groomed:	No
Sprint/Milestone:	3.12.0	Sprint Candidate:	No
Severity:	2. Medium	Tags:	
Version:		Sprint:	
Platform Release:		Quarter:	
OS:			
Triaged:	Yes		
Description			
<pre>('task_href: %s, has failed with error %f', '/pulp/api/v3/tasks/111ef25e-5798-4bdf-b619-ad120e a31ebb/', {'traceback': ' File "/opt/bats/lib/python3.8/site-packages/rq/worker.py", line 982, in perfo rm_job self.handle_job_success(job=job, File "/opt/bats/lib/python3.8/site-packages/pulpcore/tasking /worker.py", line 143, in handle_job_success task.release_resources() File "/opt/bats/lib/python3.8/site-packages/pulpcore/app/models/task.py", line 429, in relea se_resources reservation.delete() File "/opt/bats/lib/python3.8/site-packages/django_lifecycle/mixins.py", line 141, in delete super().delete(*args, **kwargs) File "/opt/bats/lib/python3.8/site-packages/django/db/models/base.py", line 922, in delete return collector.delete() File "/opt/bats/lib/python3.8/site-packages/django/db/models/deletion.py", line 317, in dele te signals.post_delete.send(File "/opt/bats/lib/python3.8/site-packages/django/db/transaction.py", line 240, in __exit__ connection.commit() File "/opt/bats/lib/python3.8/site-packages/django/db/backends/base/base.py", line 262, in c ommit self._commit() File "/opt/bats/lib/python3.8/site-packages/django/db/backends/base/base.py", line 240, in _ commit return self.connection.commit() File "/opt/bats/lib/python3.8/site-packages/django/db/utils.py", line 89, in __exit__ raise dj_exc_value.with_traceback(traceback) from exc_value File "/opt/bats/lib/python3.8/site-packages/django/db/backends/base/base.py", line 240 , in _commit return self.connection.commit()', 'description': 'update or delete on table "core_reservedresource" violates foreign key constraint "core_taskreservedres_resource_id_ee0b7c62_fk_core_rese" on table "core_taskreservedre source" DETAIL: Key (pulp_id)=(f680438a-966e-46bb-9a40-5e761237ad1e) is still referenced from table " core_taskreservedresource". '})</pre>			

Associated revisions

Revision 289c1cde - 04/08/2021 04:53 PM - dalley

Fix race condition in handling of reserved resources

closes: #8352 <https://pulp.plan.io/issues/8352>

History

#1 - 03/05/2021 04:17 PM - mdellweg

It looks like the task cleanup code looks for any task that is still attached to a reserved_resource before deleting it. But that leaves a small window where another task can attach to that reserved_resource letting the delete fail with a foreign_key_constraint.

The safe way (i think) is to try to delete it and ignore the ConstraintException.

<https://github.com/pulp/pulpcore/blob/master/pulpcore/app/models/task.py#L422>

#2 - 03/05/2021 04:43 PM - fao89

- Triaged changed from No to Yes

#3 - 04/01/2021 12:55 PM - osapryki

The root cause of the problem is that pulp worker process is not atomic. Worker cleanup is executed by workers in a concurrent environment. When a worker queries for worker record [1] and related resources [2] [3] and before worker is marked as cleaned up [4], the resource manager process may insert new records that will be neither queried nor deleted after worker cleanup process finishes.

Proposed solution to this problem is using database locks on a worker record to prevent resource manager assigning new resources on worker when the cleanup process is running.

Implementation details:

1. Worker cleanup process must run in a transaction.
2. Querying a worker for cleanup must use FOR UPDATE row level lock.
3. Resource manager resource reservation must run in a transaction.
4. Resource manager must also use FOR UPDATE lock on a worker.

When resource manager queries a worker with FOR UPDATE lock, if the worker cleanup process has started, resource manager will wait until the worker cleanup process finishes.

Resource manager should lock only the worker it assigns task and resources to.

References:

1. https://github.com/pulp/pulpcore/blob/3d3a7849fc7e7b4489e664f4a93694044aae8404/pulpcore/tasking/worker_watcher.py#L179
2. https://github.com/pulp/pulpcore/blob/3d3a7849fc7e7b4489e664f4a93694044aae8404/pulpcore/tasking/worker_watcher.py#L184
3. https://github.com/pulp/pulpcore/blob/3d3a7849fc7e7b4489e664f4a93694044aae8404/pulpcore/tasking/worker_watcher.py#L184
4. https://github.com/pulp/pulpcore/blob/3d3a7849fc7e7b4489e664f4a93694044aae8404/pulpcore/tasking/worker_watcher.py#L195

Alternative approaches: Alternative solution is to use table level locks on tasks and resource tables.

#4 - 04/02/2021 07:32 PM - pulpbot

- Status changed from NEW to POST

PR: <https://github.com/pulp/pulpcore/pull/1228>

#5 - 04/08/2021 04:53 PM - dalley

- Status changed from POST to MODIFIED

Applied in changeset [pulpcore|289c1cde5ccb6c4d5208de0e74b29ee5eef445](https://github.com/pulp/pulpcore/pull/1228).

#6 - 04/08/2021 04:54 PM - mdellweg

- Sprint/Milestone set to 3.12.0

#7 - 04/08/2021 10:56 PM - pulpbot

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE