

## Pulp - Issue #5982

### Race condition in RepositoryVersion.new\_version()

01/14/2020 02:01 PM - osapryki

<b>Status:</b>	CLOSED - WORKSFORME	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>Estimated time:</b>	0:00 hour
<b>Category:</b>			
<b>Sprint/Milestone:</b>			
<b>Severity:</b>	2. Medium	<b>Groomed:</b>	No
<b>Version:</b>		<b>Sprint Candidate:</b>	No
<b>Platform Release:</b>		<b>Tags:</b>	
<b>OS:</b>		<b>Sprint:</b>	
<b>Triaged:</b>	Yes	<b>Quarter:</b>	
<b>Description</b>			
If `RepositoryVersion.new_version` is executed at the same time in two parallel transactions, one of transactions may fail with unique constraint error:			
duplicate key value violates unique constraint "pulp_app_repositoryversion_repository_id_number_07cb5788_uniq" DETAIL: Key (repository_id, number)=(1, 735) already exists.			
This is caused because of race condition possible in `RepositoryVersion.new_version()` call:			
<pre>SELECT last_version FROM pulp_app_repository WHERE id = 1 last_version -----                 21</pre>			
<pre>TRANSACTION 1      TRANSACTION 2 ===== BEGIN; -----                 BEGIN; ----- INSERT INTO pulp_app_repositoryversion (repository_id, number) VALUES (1, 22) # NOTE: number = int(self.last_version) + 1; ----- UPDATE pulp_app_repository SET last_version = 22; -----                 INSERT INTO pulp_app_repositoryversion                 (repository_id, number) VALUES (1, 22)                 # NOTE: number = int(self.last_version) + 1; -----                 UPDATE pulp_app_repository                 SET last_version = 22; ----- COMMIT; # NOTE: OK -----                 COMMIT;                 # NOTE: uplicate key value violates unique                 # constraint -----</pre>			
<b>Solution:</b>			
You should lock repository record to perform `last_version` increment and update (e.g. do SELECT FOR UPDATE before			

incrementing last\_version counter and creating a new version).

---

## History

---

### #1 - 01/14/2020 04:17 PM - bmbouter

I want to share some info on Pulp's tasking system. Pulp's tasking system provides a serialization of tasks based on locks. In the case of new\_version, e.g. it's typically called from add\_and\_remove [here](#). That is dispatched with the lock on repository itself [second param here](#) which causes add\_and\_remove operations to be serialized.

Improving the code to have fewer race conditions sounds good though, especially if plugin writers want to create new versions in the viewset. Do plugins want to create new\_version in the viewsets?

### #2 - 01/14/2020 04:35 PM - fao89

- *Triaged changed from No to Yes*

### #3 - 10/09/2020 04:03 PM - dkliban@redhat.com

- *Status changed from NEW to CLOSED - WORKSFORME*

Repository Versions should be created inside of tasks so that the work is serialized.