

## Pulp - Issue #1580

### Cannot Download Published RPMs

01/25/2016 08:42 PM - Ichimonji10

<b>Status:</b>	CLOSED - CURRENTRELEASE	<b>Start date:</b>	
<b>Priority:</b>	High	<b>Due date:</b>	
<b>Assignee:</b>	semyers	<b>Estimated time:</b>	0:00 hour
<b>Category:</b>			
<b>Sprint/Milestone:</b>			
<b>Severity:</b>	3. High	<b>Groomed:</b>	No
<b>Version:</b>	Master	<b>Sprint Candidate:</b>	No
<b>Platform Release:</b>	2.8.0	<b>Tags:</b>	Pulp 2
<b>OS:</b>		<b>Sprint:</b>	
<b>Triaged:</b>	Yes	<b>Quarter:</b>	

#### Description

A valuable use case for the RPM plugin is the following:

1. Create an RPM repository with a feed URL such as [https://repos.fedorapeople.org/repos/pulp/pulp/demo\\_repos/zoo/](https://repos.fedorapeople.org/repos/pulp/pulp/demo_repos/zoo/).
2. Sync the repository, give it a distributor, and publish it.
3. Download an RPM from the published repository. If bear-4.1-1.noarch.rpm is available in the original repository, we should also be able to download it from Pulp's repository.

Unfortunately, this use case is currently broken. When a client attempts to download the RPM from Pulp (by issuing an HTTP GET request to e.g. <https://192.168.121.173/pulp/repos/efda1f99-315b-4cf3-802f-66211f041603/bear-4.1-1.noarch.rpm>), Pulp should serve the file up. Instead, Pulp responds with an HTTP 500 message. Here's an example of what the response headers look like:

```
{'Connection': 'close',
'Content-Length': '527',
'Content-Type': 'text/html; charset=iso-8859-1',
'Date': 'Mon, 25 Jan 2016 19:16:03 GMT',
'Server': 'Apache/2.4.18 (Fedora) OpenSSL/1.0.2e-fips mod_wsgi/4.4.8 Python/2.7.10'}
```

And here's an example of what the response body looks like:

```
u'<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">\n<html><head>\n<title>500 Internal Server Error</title>\n</head><body>\n<h1>Internal Server Error</h1>\n<p>The server encountered an internal error or \nmisconfiguration and was unable to complete\nyour request.</p>\n<p>Please contact the server administrator at \n root@localhost to inform them of the time this error occurred,\n and the actions you performed just before this error.</p>\n<p>More information about this error may be available\nin the server error log.</p>\n</body></html>\n'
```

Checking the system's log with journalctl doesn't offer many hints as to why the HTTP 500 is being generated:

```
Jan 25 19:31:06 dev pulp[9896]: kombu.transport.qpid:INFO: Connected to qpid with SASL mechanism ANONYMOUS
Jan 25 19:31:06 dev pulp[9880]: celery.worker.strategy:INFO: Received task: pulp.server.async.tasks._queue_reserved_task[10bc05b8-88e7-44cf-8152-fd6821897895]
Jan 25 19:31:06 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.managers.content.upload.import_uploaded_unit[ba49d616-b252-4b05-8fcb-271faalda325]
Jan 25 19:31:06 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.async.tasks._release_resource[58f2187f-12fd-4caf-988f-f7a41bc55b74]
Jan 25 19:31:06 dev pulp[9880]: celery.worker.job:INFO: Task pulp.server.async.tasks._queue_reserved_task[10bc05b8-88e7-44cf-8152-fd6821897895] succeeded in 0.0375700519999s: None
Jan 25 19:31:06 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.managers.content.upload.import_uploaded_unit[ba49d616-b252-4b05-8fcb-271faalda325] succeeded in 0.03955116s: {'details': {}, 'success_flag': True, 'summary': ''}
Jan 25 19:31:06 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.async.tasks._release_resource[58f2187f-12fd-4caf-988f-f7a41bc55b74] succeeded in 0.00918509700023s: None
Jan 25 19:31:06 dev pulp[9880]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
```

```
s._queue_reserved_task[dd09e443-9986-4f61-8ca1-74ff4c1d3cf3]
Jan 25 19:31:06 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.managers.r
epo.unit_association.associate_from_repo[a9164795-e67c-4e2c-9791-clc773a070b6]
Jan 25 19:31:06 dev pulp[9880]: celery.worker.job:INFO: Task pulp.server.async.tasks._queue_reserv
ed_task[dd09e443-9986-4f61-8ca1-74ff4c1d3cf3] succeeded in 0.0313184919996s: None
Jan 25 19:31:06 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
s._release_resource[ae7e6ae2-5997-4fa2-a7a2-15fb0da14f38]
Jan 25 19:31:06 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.managers.repo.unit Associ
ation.associate_from_repo[a9164795-e67c-4e2c-9791-clc773a070b6] succeeded in 0.052356604001s: {'un
its_successful': [{'unit_key': {'name': u'bear', 'checksum':...
Jan 25 19:31:06 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.async.tasks._release_reso
urce[ae7e6ae2-5997-4fa2-a7a2-15fb0da14f38] succeeded in 0.013986725s: None
Jan 25 19:31:07 dev pulp[9880]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
s._queue_reserved_task[a4a325e8-80c3-4cfb-8126-9a9d15c24acc]
Jan 25 19:31:07 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.managers.r
epo.publish.publish[e6381f22-ed0b-4cfd-ba42-1bd66613543d]
Jan 25 19:31:07 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
s._release_resource[426f1b56-d1ce-4a45-96c0-48c31c2cf51e]
Jan 25 19:31:07 dev pulp[9880]: celery.worker.job:INFO: Task pulp.server.async.tasks._queue_reserv
ed_task[a4a325e8-80c3-4cfb-8126-9a9d15c24acc] succeeded in 0.0564984590001s: None
Jan 25 19:31:07 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.managers.repo.publish.pub
lish[e6381f22-ed0b-4cfd-ba42-1bd66613543d] succeeded in 0.268445558s: {'exception': None, 'repo_id
': u'eb1e9ede-2180-4072-a534-b8ede98da398', 'traceback': None, 'started': '2016-01-25T19:31:07Z',.
..
Jan 25 19:31:07 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.async.tasks._release_reso
urce[426f1b56-d1ce-4a45-96c0-48c31c2cf51e] succeeded in 0.0107085719992s: None
Jan 25 19:31:12 dev pulp[9880]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
s._queue_reserved_task[a6bf6d3c-e015-4dc5-b485-1a29ee803a1f]
Jan 25 19:31:12 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.managers.r
epo.publish.publish[1a0d4e8f-68d9-4d53-979c-04eb44dc2455]
Jan 25 19:31:12 dev pulp[9884]: celery.worker.strategy:INFO: Received task: pulp.server.async.task
s._release_resource[e35c95b6-5519-4892-a132-5b94f38da940]
Jan 25 19:31:12 dev pulp[9880]: celery.worker.job:INFO: Task pulp.server.async.tasks._queue_reserv
ed_task[a6bf6d3c-e015-4dc5-b485-1a29ee803a1f] succeeded in 0.0306932370004s: None
Jan 25 19:31:12 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.managers.repo.publish.pub
lish[1a0d4e8f-68d9-4d53-979c-04eb44dc2455] succeeded in 0.250417443s: {'exception': None, 'repo_id
': u'8c58f8e8-c6c6-4490-9b8d-06456bacc653', 'traceback': None, 'started': '2016-01-25T19:31:12Z',.
..
Jan 25 19:31:13 dev pulp[9884]: celery.worker.job:INFO: Task pulp.server.async.tasks._release_reso
urce[e35c95b6-5519-4892-a132-5b94f38da940] succeeded in 0.00903029599976s: None
```

This bug can be reproduced with Pulp Smash. Try executing one of the following:

- python -m unittest2 pulp\_smash.tests.rpm.api\_v2.test\_broker.BrokerTestCase
- python -m unittest2 pulp\_smash.tests.rpm.api\_v2.test\_sync\_publish.PublishTestCase

You may need to remove some test skipping logic from the relevant tests for those commands to complete.

**Related issues:**

Related to Puppet Support - Issue #1599: Forge API app is failing to respond ... **CLOSED - CURRENT RELEASE**  
Related to RPM Support - Issue #1655: pulp task workers connect to mongo befo... **CLOSED - WONTFIX**

**Associated revisions**

**Revision 94517217 - 01/26/2016 05:28 PM - Jeremy Cline**

Remove database initialization from content.wsgi.

Firstly, the content application does not use the database, so it shouldn't be initializing a connection to the database. Secondly, this implementation breaks with pymongo-3.x because it was creating a connection in a parent process (pre-WSGI forking, I believe) which is not safe (see <https://jira.mongodb.org/browse/PYTHON-961>).

**History**

**#1 - 01/26/2016 07:17 PM - jcline@redhat.com**

- Status changed from NEW to MODIFIED
- Assignee set to jcline@redhat.com
- Triaged changed from No to Yes

<https://github.com/pulp/pulp/pull/2365>

## #2 - 01/26/2016 11:57 PM - Ichimonji10

I spun up a pair of fresh Pulp 2.7 and 2.8 VMs this afternoon, and can confirm that this bug is fixed.

## #3 - 01/27/2016 12:07 AM - Ichimonji10

See: <https://github.com/PulpQE/pulp-smash/commit/47535f12351d6dc5049b9c7551cbeaa5aa640af4>

## #4 - 01/27/2016 11:36 PM - dkliban@redhat.com

- Subject changed from *Cannot Download Published RPMs to Mongo connection is opened before process is forked*

- Status changed from *MODIFIED* to *NEW*

- Assignee deleted (*jcline@redhat.com*)

I noticed the following message from a worker that received it's first task:

```
Jan 27 22:16:08 dev pulp[19827]: celery.worker.strategy:INFO: Received task: pulp.server.managers.consumer.applicability.batch_regenerate_applicability[878da468-d460-426b-9094-295c73f9ce5e]
Jan 27 22:16:08 dev pulp[20123]: py.warnings:WARNING: (20123-13184) /usr/lib64/python2.7/site-packages/pymongo/topology.py:74: UserWarning: MongoClient opened before fork. Create MongoClient with connect=False, or create client after forking. See PyMongo's documentation for details: http://api.mongodb.org/python/current/faq.html#using-pymongo-with-multiprocessing>
Jan 27 22:16:08 dev pulp[20123]: py.warnings:WARNING: (20123-13184) "MongoClient opened before fork. Create MongoClient "
Jan 27 22:16:08 dev pulp[20123]: py.warnings:WARNING: (20123-13184)
```

We need to fix this for our celery workers also.

## #5 - 01/28/2016 03:24 PM - rbarlow

- Project changed from *RPM Support* to *Pulp*

- Priority changed from *Normal* to *High*

- Severity changed from *2. Medium* to *3. High*

- Platform Release set to *2.8.0*

## #6 - 01/28/2016 10:15 PM - rbarlow

- Related to Issue #1599: *Forge API app is failing to respond to requests from httpd added*

## #7 - 02/01/2016 08:34 PM - semyers

- Status changed from *NEW* to *ASSIGNED*

- Assignee set to *semyers*

## #8 - 02/05/2016 03:03 AM - semyers

The warning message we're seeing about opening a connection before forking shouldn't be breaking anything; it only potentially causes problems in BSD-derived environments[0]. I'm going to go back to the original issue as-reported to try to reproduce and fix it (if needed). We can probably also squash the connect before fork warning. Randy asked about having pymongo itself not even issue the warning[1], but upstream disagreed. We've eliminated potentially harmful pre-fork connections in other places, but those that remain are proving difficult to get rid of. They appear to be happening as a result of minimal database activity during worker startup, at which point the forking process never uses the database connection, effectively granting the forked process exclusive control of it.

[0]: <https://emptysqua.re/blog/getaddrinfo-deadlock/>

[1]: <https://jira.mongodb.org/browse/PYTHON-961>

## #9 - 02/05/2016 05:25 PM - semyers

The original issue as reported is fixed, so the acceptance criteria for this issue has been reduced to "pulp suppresses a scary warning from pymongo about connecting before forking". That'll be a fun test to write. :)

Suppressing the warning is certainly not a solution to this issue, but I think preventing the scary warning from showing up in the logs is a good call for 2.8. Once that's done I'll open up a new task for finding where we're connecting to mongo pre-fork.

## #10 - 02/10/2016 07:00 PM - semyers

I started a remote debugger in pymongo right after the offending warning gets raised, and narrowed it down to the following calls:

```

> /home/vagrant/devel/pulp/server/pulp/server/async/tasks.py(105)__call__()
-> return super(PulpTask, self).__call__(*args, **kwargs)
(Pdb) l
100         Deserializes args and kwargs using _type_transform()
101         """
102         args = self._type_transform(args)
103         kwargs = self._type_transform(kwargs)
104     ->         return super(PulpTask, self).__call__(*args, **kwargs)

-> task_status = TaskStatus.objects.get(task_id=self.request.id)
(Pdb) l
452         This overrides PulpTask's __call__() method. We use this method
453         for task state tracking of Pulp tasks.
454         """
455         # Check task status and skip running the task if task state is 'canceled'.
456         try:
457     ->             task_status = TaskStatus.objects.get(task_id=self.request.id)
458         except DoesNotExist:
459             task_status = None
460         if task_status and task_status['state'] == constants.CALL_CANCELED_STATE:
461             _logger.debug("Task cancel received for task-id : [%s]" % self.request.id)
462             return

> /home/vagrant/devel/pulp/server/pulp/server/async/tasks.py(472)__call__()
-> set__state=constants.CALL_RUNNING_STATE, set__start_time=start_time, upsert=True)
(Pdb) l
467         now = datetime.now(dateutils.utcnow())
468         start_time = dateutils.format_iso8601_datetime(now)
469         # Using 'upsert' to avoid a possible race condition described in the apply_async method
470         # above.
471         TaskStatus.objects(task_id=self.request.id).update_one(
472     ->             set__state=constants.CALL_RUNNING_STATE, set__start_time=start_time, upsert=True)
473         # Run the actual task
474         _logger.debug("Running task : [%s]" % self.request.id)
475         return super(Task, self).__call__(*args, **kwargs)
476
477         def on_success(self, retval, task_id, args, kwargs):

> /home/vagrant/devel/pulp/server/pulp/server/async/tasks.py(498)on_success()
-> task_status = TaskStatus.objects.get(task_id=task_id)
(Pdb) l
493                 % {'id': kwargs['scheduled_call_id']})
494                 utils.reset_failure_count(kwargs['scheduled_call_id'])
495         if not self.request.called_directly:
496             now = datetime.now(dateutils.utcnow())
497             finish_time = dateutils.format_iso8601_datetime(now)
498     ->             task_status = TaskStatus.objects.get(task_id=task_id)
499             task_status['finish_time'] = finish_time
500             task_status['result'] = retval
501
502             # Only set the state to finished if it's not already in a complete state. This is
503             # important for when the task has been canceled, so we don't move the task from canceled

> /home/vagrant/devel/pulp/server/pulp/server/async/tasks.py(523)on_success()
-> task_status.save()
(Pdb) l
518                 task_status['spawned_tasks'] = task_list
519                 if isinstance(retval, AsyncResult):
520                     task_status['spawned_tasks'] = [retval.task_id, ]
521                     task_status['result'] = None
522
523     ->                 task_status.save()
524                 common_utils.delete_working_directory()
525
526         def on_failure(self, exc, task_id, args, kwargs, einfo):
527             """
528             This overrides the error handler run by the worker when the task fails.

```

These have all be wrapped in a context manager that suppresses the specific pymongo warning while letting all other warnings be logged as usual.

Additionally, [#1655](#) has been created to track the task of finding and fixing the pre-fork connection that causes these warnings to occur in the first place.

#11 - 02/10/2016 07:01 PM - semyers

- Related to Issue #1655: pulp task workers connect to mongo before forking added

**#12 - 02/10/2016 07:14 PM - semyers**

- Status changed from ASSIGNED to POST

<https://github.com/pulp/pulp/pull/2409>

**#13 - 02/10/2016 09:13 PM - semyers**

- File 1580.patch added

- Status changed from POST to MODIFIED

After a good discussion with bmbouter, I've closed the related PR. Reliably suppressing the pymongo warnings has proven difficult, if not impossible. After seeing what it takes to suppress the warnings, we've concluded that having the warnings in the log is not a release blocker for 2.8.0 if this is what the fix looks like. That fix is now attached to this issue as a patch in case we want to bring it back in the future (or downstream). I'm setting this back to MODIFIED, which was its state based on jcline's earlier commit, [94517217228352f48b3bbcaecfbac5cc726f02ec](https://github.com/pulp/pulp/commit/94517217228352f48b3bbcaecfbac5cc726f02ec), before the pymongo warnings took focus.

**#14 - 02/23/2016 09:56 PM - dkliban@redhat.com**

- Status changed from MODIFIED to 5

**#15 - 03/02/2016 05:12 PM - pthomas@redhat.com**

- Status changed from 5 to 6

```
[root@mgmt3 ~]# rpm -qa pulp-server
pulp-server-2.8.0-0.6.beta.git.40.28675fb.el7.noarch
```

```
$ python -m unittest2 pulp_smash.tests.rpm.api_v2.test_broker.BrokerTestCase
```

```
..
-----
```

```
Ran 2 tests in 244.155s
```

```
OK
```

```
$ python -m unittest2 pulp_smash.tests.rpm.api_v2.test_sync_publish.PublishTestCase
```

```
.....
-----
```

```
Ran 10 tests in 60.996s
```

```
OK
```

**#16 - 03/23/2016 07:48 PM - dkliban@redhat.com**

- Status changed from 6 to CLOSED - CURRENTRELEASE

**#18 - 02/07/2017 04:51 PM - mhrivnak**

- Subject changed from *Mongo connection is opened before process is forked* to *Cannot Download Published RPMs through streamer*

I'm changing the title back to address the original problem that was reported and fixed.

The mongo warnings in the log are a separate issue that was not resolved here.

**#19 - 02/07/2017 04:51 PM - mhrivnak**

- Subject changed from *Cannot Download Published RPMs through streamer* to *Cannot Download Published RPMs*

**#20 - 04/15/2019 10:37 PM - bmbouter**

- Tags Pulp 2 added

**Files**

---

1580.patch	16.7 KB	02/10/2016	semyers
------------	---------	------------	---------