

RPM Support - Issue #2540

Syncing repo with 200,000+ RPMs causes a BSONObj size limit exception

01/24/2017 12:18 AM - semyers

Status:	CLOSED - WONTFIX	Start date:	
Priority:	High	Due date:	
Assignee:	jortel@redhat.com	Estimated time:	0:00 hour
Category:			
Sprint/Milestone:			
Severity:	3. High	Groomed:	No
Version:		Sprint Candidate:	No
Platform Release:		Tags:	Pulp 2
OS:		Sprint:	Sprint 16
Triaged:	Yes	Quarter:	

Description

Description as copied from Bugzilla:

Running on rhel 6 with mongo 2.4 on a very large satellite install (with ~200,000 rpms), it is possible to get an error on sync:

```
Traceback (most recent call last):
```

```
File "/usr/lib/python2.6/site-packages/celery/app/trace.py", line 240, in trace_task
  R = retval = fun(*args, **kwargs)
```

```
File "/usr/lib/python2.6/site-packages/pulp/server/async/tasks.py", line 473, in __call__
  return super(Task, self).__call__(*args, **kwargs)
```

```
File "/usr/lib/python2.6/site-packages/pulp/server/async/tasks.py", line 103, in __call__
  return super(PulpTask, self).__call__(*args, **kwargs)
```

```
File "/usr/lib/python2.6/site-packages/celery/app/trace.py", line 437, in __protected_call__
  return self.run(*args, **kwargs)
```

```
File "/usr/lib/python2.6/site-packages/pulp/server/controllers/repository.py", line 810, in
```

sync

```
    raise pulp_exceptions.PulpExecutionException(_('Importer indicated a failed response'))
```

```
PulpExecutionException: Importer indicated a failed response
```

```
'command SON([('mapreduce', u'units_rpm'), ('map', Code("
function() {
  var key_fields = [this.name, this.epoch, this.version, this.release, this.arch]
  emit(key_fields.join('-'), {ids: [this._id]});
}
", {})), ('reduce', Code("
function (key, values) {
  // collect mapped values into the first value to build the list of ids for this key/nevra
  var
collector = values[0]
  // since collector is values[0] start this loop at index 1
  // reduce isn't called if map only emits one result for key,
  // so there is at least one value to collect
  for(var i = 1; i < values.length; i++) {
    collector.ids = collector.ids.concat(values[i].ids)
  }
  returncollector
}
", {})), ('out', {'inline': 1}), ('query', {}), ('finalize', Code("
function (key, reduced) {
  if (reduced.ids.length > 1) {
return reduced;
  }
  // if there's only one value after reduction, this key is useless
  // undefined is implicitly returned here, which saves space
}

```

```
"", {}))]] on namespace pulp_database.$cmd failed: exception: BSONObj size: 18210078 (0x1EDD1501) is invalid. Size must be between 0 and 16793600(16MB) First element: 0:
{ _id: "0ad-0-0.0.20-4.e17-x86_64", value: null }'
```

The mapreduce code seen in that traceback appears here:

https://github.com/pulp/pulp_rpm/blob/2.12-dev/plugins/pulp_rpm/plugins/importers/yum/purge.py#L466-L479

It should only be run when "annotate" is unavailable in mongodb, indicating a version of mongo 2.4 or lower. which happens on el6. This problem shouldn't occur when "annotate" is available, since the most notable difference between the methods is that when using "annotate" mongo is able to return a cursor that can be used to gather results iteratively, where the "mapreduce" method returns all of the mapped/reduced data in a single document.

History

#1 - 01/24/2017 12:18 AM - semyers

I thought this issue already existed in redmine, but while I did find a few similar issues, I was unable to find one tracking this particular problem. Apologies if this is a dupe.

#2 - 01/24/2017 06:57 PM - bizhang

- Priority changed from Normal to High

- Sprint/Milestone set to 32

- Severity changed from 2. Medium to 3. High

- Triaged changed from No to Yes

#3 - 02/06/2017 02:53 PM - mhrivnak

- Sprint/Milestone changed from 32 to 33

#4 - 02/07/2017 10:06 PM - semyers

While we're working to get this fixed, a workaround is to use mongo 2.6 or better, available via software collections:

<https://www.softwarecollections.org/en/scls/?search=mongo>

#5 - 02/07/2017 10:28 PM - fdobrovo

I have found a possible solution. This errors arises from the fact that we fetch from the mongoDB all rpm units that are stored there in form

```
{ _id: "0ad-0-0.0.20-4.e17-x86_64", value: null }
```

which is already stripped version instead of null on the mongo originally there would be id of the unit. This need arises from Mongodb that has limit of the size of the response document on this particular version it's 16 MB, but in other versions it's half and even smaller. But when we have large pulp database it's not enough. Ideally we would need to filter out those which are not duplicated which is practically 99% of all them. It's long wanted feature which never made it into mongo <https://jira.mongodb.org/browse/SERVER-2340>

The only possible solution I found is not to send the result to pulp, but write it into another collection and after that fetch from the new collection only the ones which we need.

But it has one downsides. I can't predict how will it behave when multiple syncs are issued at once it might need to incorporate the name of the worker to the name of the collection.

Also there is possibility to maybe do it faster it's possible to store it once for the collection and do it incrementally, but that requires last change date on the units which I don't know if they have.

The changes would be minimal, except handling the issue mentioned above it's change on two lines.

#6 - 02/17/2017 05:42 PM - fdobrovo

Possible solutions to concurrent sync:

- Name of the worker would be used as the name of the temporary collection
- There would be just one collection per unit type and every time sync would be issued new lookup for duplicates would be made. After that the tmp. collection would be in state after sync which might not resemble actual state as some duplicates may be already solved, by previously running workers. Solution to sub problem:
 - Checking if all unit duplicated units exists
 - Deleting the record of duplication before de-duplicating, but this would involve pulling list of duplicated NEVRAs at the beginning and than asking again for the specific NEVRA if it wasn't resolved yet.

#7 - 02/23/2017 10:15 PM - jortel@redhat.com

- Status changed from NEW to ASSIGNED

- Assignee set to jortel@redhat.com

#8 - 02/27/2017 03:55 PM - mhrivnak

- Sprint/Milestone changed from 33 to 34

#12 - 03/03/2017 05:07 PM - jortel@redhat.com

Recommend users upgrade to mongo 2.6 using SCL (as needed).

Next steps:

- Do a trial upgrade
- Document the process.

Need to validate that mongod upgraded via SCL will still use /var/lib/mongo and /etc/mongod.conf and functions. Basically upgrades normally.

#13 - 03/03/2017 05:23 PM - bizhang

- Assignee changed from jortel@redhat.com to bizhang

#14 - 03/06/2017 08:58 PM - bizhang

Steps to install mongodb26 via scl:

1. Stop mongod24

```
mongod --dbpath=/var/lib/mongod --shutdown
```

2. Enable the scl repo:

```
yum-config-manager --enable rhel-server-rhsccl-6-rpms
```

3. Install mongodb26:

```
yum install rh-mongodb26 rh-mongodb26-mongod
```

4. mongod should now be installed in

```
/opt/rh/rh-mongodb26/root/usr/bin/
```

Add this to your path if it has not been automatically added

5. Enable the mongod scl:

```
scl enable rh-mongodb26 bash
```

6. Copy the configuration from /etc/mongod.conf to /etc/opt/rh/rh-mongodb26/mongod.conf

Make sure to update the pidfilepath and logpath to point to /var/opt/rh/rh-mongodb26/run/mongod/ instead of /var/lib/mongod since /var/lib/mongod is removed when mongod24 is uninstalled

7. Start mongod26

```
service rh-mongodb26-mongod start
```

8. Make sure mongod is 2.6:

```
mongod --version
```

9. Confirm that all the pulp data is accessible

10. Remove mongo24

```
yum remove mongod mongodb-server
```

#15 - 03/06/2017 09:33 PM - bizhang

- Assignee changed from bizhang to jortel@redhat.com

#16 - 03/17/2017 05:02 PM - jortel@redhat.com

- Status changed from ASSIGNED to CLOSED - WONTFIX

#17 - 03/08/2018 11:42 PM - bmbouter

- Sprint set to Sprint 16

#18 - 03/09/2018 12:05 AM - bmbouter

- Sprint/Milestone deleted (34)

#19 - 04/15/2019 10:21 PM - bmbouter

- Tags Pulp 2 added