Foreman - Bug #20932

rake process dying with memory errors

09/14/2017 10:10 AM - Bhanu Prasad Ganguru

Status:	New		
Priority:	Normal		
Assignee:			
Category:	Rake tasks		
Target version:			
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	1.13.0
Bugzilla link:	1487050	**Red Hat JIRA:**	
Pull request:			

Description

Hi,
we're using foreman 1.13.0

Foreman host is provisioned with 8G Memory initially
It worked fine for a few months and then OOM started killing rake process

so we increased RAM from 8 to 16G
After a few months rake again started taking up all memory

now we increased RAM to 32G
Now the issue is I see 2 rake processes running all the time
Even if I kill both of them, After some time I see both processes running again and one of them is getting killed by OOM

Is this a known issue??
Is there an resolution for this???

Thanks in advance,
Bhanu

History

#1 - 09/14/2017 10:36 AM - Ohad Levy

which rake task are you actually running? I assume its started from cron?

also, 1.13 is really old at this stage, please consider upgrading.

#2 - 09/14/2017 11:07 AM - Bhanu Prasad Ganguru

Hi Ohad,
Yes it's a cron for `foreman-rake`

And

I know 1.13 is old, but I'm worried to upgrade since we're in production
What is the impact of upgrading to 1.14.3 from 1.13.0 and do we have to update puppet as well ??
we're using puppet 4.8.2

What are the other dependencies that might break

Bhanu

#3 - 09/14/2017 11:45 AM - Ivan Necas

There can be a lot of subcommands in foreman-rake, please provide the full command that is consuming the memory.

#4 - 09/14/2017 12:05 PM - Bhanu Prasad Ganguru

the two commands that are running

```
PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
18307 foreman   20   0 12.346g 0.012t   1780 R  63.5 38.2  24:42.35 /opt/rh/rh-ruby22/root/usr/bin/ruby /opt/r
h/rh-ruby22/root/usr/bin/rake trends:counter
15431 foreman   20   0 13.278g 0.013t   1228 R  62.1 41.2  48:09.28 /opt/rh/rh-ruby22/root/usr/bin/ruby /opt/r
h/rh-ruby22/root/usr/bin/rake trends:counter
```

#5 - 09/15/2017 07:22 AM - Ivan Necas

- Bugzilla link set to 1487050

#6 - 09/15/2017 07:26 AM - Ivan Necas

Branu: do you think it would be possible to share the data from trends and trend_counters tables from your setup, in case it's doesn't contain sensitive data, for further analysis?

#7 - 09/15/2017 10:13 AM - Bhanu Prasad Ganguru

We don't have any sensitive data
Here you go

```
foreman=> SELECT count(*) FROM trends; count
count
---------
4656994
(1 row)

foreman=> select count(*) from trend_counters;  count
count
---------
4182107
(1 row)

foreman=> SELECT * FROM trend_counters;
id    | trend_id | count |         created_at         |         updated_at         |       interval_start       |        interval_end
---------+----------+-------+----------------------------+----------------------------+----------------------------+----------------------------
1216781 |   609217 |     0 | 2017-04-07 16:30:23.460929 | 2017-05-01 00:52:34.951262 | 2017-04-07 16:30:23.460929 | 2017-04-30 23:30:29.987925
1584036 |   795547 |     1 | 2017-04-23 18:30:25.152193 | 2017-04-23 19:42:01.619967 | 2017-04-23 18:30:25.152193 | 2017-04-23 19:00:25.00961
391505 |   195174 |     0 | 2017-03-24 10:00:11.799516 | 2017-05-05 10:17:52.432887 | 2017-03-24 10:00:11.799516 | 2017-05-05 09:00:33.869969
1843682 |   923705 |     0 | 2017-04-28 06:00:28.884791 | 2017-05-11 02:04:03.060378 | 2017-04-28 06:00:28.884791 | 2017-05-11 01:00:36.391698
3482888 |  3209176 |     1 | 2017-07-16 09:31:26.047687 | 2017-07-16 10:10:44.469751 | 2017-07-16 09:31:26.047687 | 2017-07-16 10:01:26.102196
256204 |   128217 |     1 | 2017-03-22 02:00:11.894811 | 2017-03-22 02:38:46.610683 | 2017-03-22 02:00:11.894811 | 2017-03-22 02:30:12.895528
3256428 |  2510624 |     1 | 2017-06-22 06:31:08.186384 | 2017-06-22 07:37:58.758556 | 2017-06-22 06:31:08.186384 | 2017-06-22 07:01:08.487333
617004 |   308905 |     1 | 2017-03-28 07:30:12.558068 | 2017-03-28 08:04:20.130362 | 2017-03-28 07:30:12.558068 | 2017-03-28 08:00:12.775755
1484306 |   746300 |     1 | 2017-04-22 02:00:24.346014 | 2017-04-22 02:41:03.12227  | 2017-04-22 02:00:24.346014 | 2017-04-22 02:30:24.141439
1074555 |   537074 |     1 | 2017-04-05 04:30:19.695444 | 2017-04-05 05:22:43.2218   | 2017-04-05 04:30:19.695444 | 2017-04-05 05:00:20.671482

foreman=> SELECT * FROM trends;
id    | trendable_type | trendable_id |                    name                    |   type    |                  fact_value                  |  fact_name   |         created_at
|
updated_at
---------+----------------+--------------+--------------------------------------------+-----------+----------------------------------------------+--------------+----------------------------
+----
------------------------
1 | FactName       |          115 | host uptime                                | FactTrend |                                              | system_uptime | 2017-03-17
15:52:57.564875 | 2017-03-17 15:52:57.564875
2 | FactName       |          115 | uptime18 dayshours448days18seconds1612821  | FactTrend | uptime18 dayshours448days18seconds1612821     |
system_uptime | 2017-03-17 15:52:57.602467 | 201
7-03-17 15:52:57.602467
3 | FactName       |          115 | hours452days18seconds1628830uptime18 days      | FactTrend | hours452days18seconds1628830uptime18 days     |
system_uptime | 2017-03-17 15:52:57.606234 | 201
7-03-17 15:52:57.606234
4 | FactName       |          115 | days121uptime121 dayysseconds10539341hours2927  | FactTrend | days121uptime121
dayysseconds10539341hours2927  | system_uptime | 2017-03-17 15:52:57.609622 | 201
7-03-17 15:52:57.609622
5 | FactName       |          115 | uptime170 dayysseconds14760749hours4100days170  | FactTrend | uptime170
dayysseconds14760749hours4100days170  | system_uptime | 2017-03-17 15:52:57.613055 | 201
7-03-17 15:52:57.613055
6 | FactName       |          115 | uptime150 daysdays150seconds13017020hours3615  | FactTrend | uptime150
daysdays150seconds13017020hours3615  | system_uptime | 2017-03-17 15:52:57.616343 | 201
7-03-17 15:52:57.616343
7 | FactName       |          115 | hours2106uptime87 dayysseconds7582934days87     | FactTrend | hours2106uptime87 dayysseconds7582934days87
| system_uptime | 2017-03-17 15:52:57.619632 | 201
7-03-17 15:52:57.619632
8 | FactName       |          115 | hours452seconds1629759days18uptime18 days      | FactTrend | hours452seconds1629759days18uptime18 days     |
system_uptime | 2017-03-17 15:52:57.622916 | 201
```

7-03-17 15:52:57.622916
9 | FactName | 115 | days17seconds1541917hours428uptime17 days | FactTrend | days17seconds1541917hours428uptime17 days | system_uptime | 2017-03-17 15:52:57.626159 | 201
7-03-17 15:52:57.626159
10 | FactName | 115 | days191seconds16504647hours4584uptime191 days | FactTrend | days191seconds16504647hours4584uptime191 days | system_uptime | 2017-03-17
15:52:57.629555 | 201

#8 - 09/17/2017 04:39 AM - Shimon Shtein

Could you please export your trends and tren_counters tables data to a zip file, so I would be able to reproduce the memory consumption?

For psql you can use:

```
psql -c "COPY trends TO stdout DELIMITER ',' CSV HEADER" | gzip > trends.csv.gz
psql -c "COPY trend_counters TO stdout DELIMITER ',' CSV HEADER" | gzip > trend_counters.csv.gz
```

Sorry, don't know how to do it on mysql.

#9 - 09/19/2017 10:56 AM - Bhanu Prasad Ganguru

Hi Shimon,
I am unable to export tables due to the upload size limit
I can email those directly if you can give me your email

Bhanu

#10 - 10/12/2017 06:12 PM - Bhanu Prasad Ganguru

Hi Ivan,

we've upgraded to foreman 1.14.3

And I found the foreman-rake trends:counter is what taking all the memory

My question is I can't even load trends from foreman api
It's taking almost around 50G, but still sits at loading
we only have one trend named host uptime
I stopped trends:counter cron job

Is there a way to purge some of the trends
By looking at postgres, all the trends that are in db are not older than 6 months

Any help would be appreciated

Bhanu