

Foreman - Bug #687

rake reports:expire abuses memory and network bandwidth

02/23/2011 11:04 AM - Kal McFate

Status:	Closed	Fixed in Releases: Found in Releases: Red Hat JIRA:
Priority:	Normal	
Assignee:	Ohad Levy	
Category:	Database	
Target version:	0.4	
Difficulty:		
Triaged:		
Bugzilla link:		
Pull request:		
Description		
Noticed this specifically from the reports:expire process:		
SELECT * FROM `reports` WHERE (created_at < '2011-02-22 15:52:29' and status = 0)		
There is no need at all to send the entire row. With thousands of hosts with 50 reports a day, this job sends GB of data over the network from the sql server (and into the rake processes memory) every time it is run just to delete reports. This should be optimized as it can cause some serious memory and network usage problems if processing a large number of reports.		

Associated revisions

Revision 0de3b547 - 07/18/2011 09:39 AM - Ohad Levy

fixes #687 - this dramatically improves expired reports deletion speed and system usage.

History

#1 - 02/23/2011 03:12 PM - Kal McFate

Just for example:

I ran:
rake reports:expire days=12

followed by:
rake reports:expire days=10

results in:
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
5830 root 19 0 1300m 1.2g 3484 R 95.7 30.7 28:56.57 /usr/bin/ruby /usr/bin/rake reports:expire days=10

on 2 days of data.

30 minutes, and still going.

#2 - 02/23/2011 05:14 PM - Ohad Levy

yes you are correct.

the main reason we need to pull all reports is to know if we need to delete the related log, message and source records.

maybe we could improve the finder sql to make it faster.

one thing we could do quickly, is simply to read 1000 records or so each time instead of loading them all into memory.

#3 - 07/18/2011 09:13 AM - Ohad Levy

- Category set to Database
- Status changed from New to Assigned
- Assignee set to Ohad Levy
- Target version set to 0.4

#4 - 07/18/2011 09:40 AM - Ohad Levy

- *Status changed from Assigned to Closed*

- *% Done changed from 0 to 100*

Applied in changeset [0de3b547b52b2e2046d1953ffb286afcf2e46413](#).