

## Foreman - Bug #1546

### report::expire slow with many reports

03/22/2012 02:36 PM - Jacob McCann

<b>Status:</b> Duplicate	
<b>Priority:</b> Normal	
<b>Assignee:</b> Ori Rabin	
<b>Category:</b> Reporting	
<b>Target version:</b> 1.7.5	
<b>Difficulty:</b>	<b>Fixed in Releases:</b>
<b>Triaged:</b>	<b>Found in Releases:</b>
<b>Bugzilla link:</b>	<b>Red Hat JIRA:</b>
<b>Pull request:</b>	
<b>Description</b>	
<p>report::expire is causing me issues, one of which is its very slow to run (~20 minutes, and thats before it ultimately errors).</p> <p>Ohad believes this is due to table scans:</p> <p>Ok, one simple thing that I found out, was that we were using the following query:</p> <pre>Report Load (16588.0ms) SELECT id FROM `reports` WHERE (reports.id = 0) AND (created_at &lt; '2011-06-16 14:13:02') ORDER BY reports.id ASC</pre> <p>LIMIT 1000</p> <p>if you run the following query in mysql console using explain:</p> <pre>explain SELECT id FROM `reports` WHERE (reports.id &gt;= 0) AND (created_at &lt; '2012-06-16 14:13:02') ORDER BY reports.id ASC LIMIT 1000;</pre> <p>You would see that the query was forced to scan the entire reports table and was not using the indexes.</p> <p>it should be a "bit" faster to change from created_at to reported_at (as created_at has no index at all). I assume that the reason why using the older code was better for you (as it was using reported_at which has an index) and using the in batches was just a side effect.</p> <p>we need to figure out the correct index for making this process a bit faster, in my limited tests, creating index on multiple columns (id and reported_at) were not very helpful.</p> <p>in your case, creating an index is not trivial (as you have a large set of data) as the indexes usually locks the db.</p> <p><a href="https://groups.google.com/d/msg/foreman-users/_Mn4oxXmP7E/T3ByUZ9YTclJ">https://groups.google.com/d/msg/foreman-users/_Mn4oxXmP7E/T3ByUZ9YTclJ</a></p>	
<b>Related issues:</b>	
Is duplicate of Foreman - Bug #1592: report::expire errors with millions of r...	<b>Closed</b> <b>04/30/2012</b>

### History

#1 - 04/30/2012 01:43 PM - Jacob McCann

To me the slowness of the expire seems more to do with the giant arrays being generated from the loops:

**#2 - 11/13/2012 12:04 PM - Greg Sutcliffe**

- Target version set to Bug scrub
- Start date deleted (03/22/2012)

**#3 - 10/16/2013 04:03 PM - Greg Sutcliffe**

- Target version deleted (Bug scrub)

**#4 - 07/30/2014 09:13 AM - Ohad Levy**

- Target version set to 1.7.5

**#5 - 08/04/2014 07:32 AM - Ori Rabin**

- Description updated
- Status changed from New to Assigned
- Assignee set to Ori Rabin

**#6 - 08/06/2014 06:55 AM - Ori Rabin**

- Is duplicate of Bug #1592: report::expire errors with millions of records added

**#7 - 08/06/2014 06:55 AM - Ori Rabin**

- Status changed from Assigned to Duplicate