Redmine - Defect #11678

MySQL locks and Redmine collapsing

2012-08-22 10:20 - Petr Pospisil

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Database	**Estimated time:**	0.00 hour
Target version:			
Resolution:	Duplicate	**Affected version:**	2.0.3

Description

Hello

Our heavily used Redmine often dies during update or create a new issue:

ActiveRecord::StatementInvalid: Mysql::Error: Lock wait timeout exceeded; try restarting transaction: SELECT `issues`.* FROM `issues` WHERE `issues`.`root_id` IS NULL ORDER BY `rgt` desc LIMIT 1 FOR UPDATE

Ruby 1.8.7, MySql 5.5.4, Redmine 2.0.3, OS FreeBSD, WebServer is Passenger. Every 5 minutes is running rake redmine:email:receive_pop3.

No errors in production.log.

It is imposible to simulate this behaviour, but it occurs if more than 5-10 peoples are working. We had no problems on older Redmine < 2.0.

It's frustrating...

Related issues:

Is duplicate of Redmine - Defect #6579: Tree hierachy being currupted on mult...	**Closed**	**2010-10-05**

History

#1 - 2012-08-22 10:52 - Etienne Massip

- Category set to Database

Is there a way you could give more information about the existing blocking transaction when the problem occurs?

I'm not a MySQL expert, maybe http://dev.mysql.com/doc/refman/5.6/en/innodb-monitors.html#innodb-standard-monitor could be of some help (for InnoDB).

#2 - 2012-08-22 12:06 - Petr Pospisil

I should you provide more information, but i dont know which... I think the hint is in upgrade Redmine from 1.9.x to 2.0.3. Others systems are same.

#3 - 2012-08-22 12:24 - Petr Pospisil

deleting issue: ActiveRecord::StatementInvalid: Mysql::Error: Lock wait timeout exceeded; try restarting transaction: SELECT id FROM `issues` WHERE (`lft` >= 1) FOR UPDATE

#4 - 2012-08-22 14:08 - Etienne Massip

Petr Pospisil wrote:

> I should you provide more information, but i dont know which...

Told you, blocking transactions. Because the SQL you've been writing down here is the blocked one, not the blocking one.

Try to execute a SHOW ENGINE INNODB STATUS\G as soon as the issue occurs again?

#5 - 2012-08-22 14:28 - Toshi MARUYAMA

Try increase max pool size.
http://stackoverflow.com/questions/3966215/how-to-increase-max-pool-size-in-activerecord

#6 - 2012-08-22 17:24 - Petr Pospisil

- File innodb.txt added

SHOW ENGINE INNODB STATUS attached

Added to database.yml:
pool: 20
wait_timeout: 30

#7 - 2012-08-22 17:51 - Etienne Massip

Petr Pospisil wrote:

> SHOW ENGINE INNODB STATUS attached

Not sure but you might have to create a innodb_lock_monitor table teporarily to get lock information by issuing the SHOW ENGINE INNODB STATUS command.

As I said, I'm not a MySQL expert.

#8 - 2012-08-28 15:33 - Etienne Massip

You can identify long running requests:

- enable request logging and use request-log-analyzer
- set a value to config.active_record.auto_explain_threshold_in_seconds parameter in your config/environment/production.rb so that they automatically get an explain plan in log

BTW, by default there should not be such "FOR UPDATE" part at the end of statement unless specified in code and there's no such thing.
Could that be a tuned parameter or a plugin behavior?

#9 - 2012-09-18 12:37 - Petr Pospisil

Nobody uses FOR UPDATE statement. But the problem seems to be solved. We bought better server and we changed ruby to 1.9.3... :o)

#10 - 2012-09-18 16:48 - Etienne Massip

Petr Pospisil wrote:

> Nobody uses FOR UPDATE statement. But the problem seems to be solved. We bought better server and we changed ruby to 1.9.3... :o)

Did you change adapter from mysql to mysql2 too?

#11 - 2012-09-19 10:18 - Petr Pospisil

Of course. The original adapter mysql is not compatible with ruby 1.9.3.
There are other benefits - Rails 3.x.y is much faster with new Ruby. E.g. /issues - up to 4 times!
(And FreeBSD were replaced by Debian)

#12 - 2012-09-19 10:30 - Etienne Massip

Petr Pospisil wrote:

> Of course. The original adapter mysql is not compatible with ruby 1.9.3.

And do you still have FOR UPDATE clause appended to the end of the SELECT statement?

#13 - 2012-09-19 14:12 - Petr Pospisil

Nop. As I wrote earlier. After upgrading HW and migrating to mysql2 and ruby 193 the locks not occured yet.

#14 - 2015-05-21 19:43 - @ go2null

related to Defect #8143 *Problem to update subtask*.

I am also having the same issue - it is caused with large parent-child hierarchies and the consequent cascading updates of status, priority and other checks.

```
Environment:
  Redmine version                2.6.0.stable
  Ruby version                   1.9.3-p484 (2013-11-22) [x86_64-linux]
```

```
Rails version                   3.2.19
Environment                     production
Database adapter                Mysql2
```

#15 - 2015-06-20 10:22 - Toshi MARUYAMA

- Status changed from New to Closed

- Resolution set to Duplicate

This is fixed by #6579 on 3.0.0.
But there are new problems of MySQL 5.6 and 5.7 (#17460, #19344).

#16 - 2015-06-20 10:23 - Toshi MARUYAMA

- Is duplicate of Defect #6579: Tree hierachy being currupted on multiple submissions of an issue added

Files

innodb.txt	5.61 KB	2012-08-22	Petr Pospisil