

Краткий отчет по выявлению и устранению проблемы регулярного «зависания» сервиса MySQL на сервере СУБД веб-проекта <http://example.com>

Исходные данные:

- выделенный сервер СУБД MySQL (ip 7.8.9.10)
- наблюдаемое периодическое «зависание» сервиса MySQL с отказом обслуживания новых запросов и потреблением ресурсов CPU и оперативной памяти в 100% объеме.
- текущий способ решения проблемы - периодический перезапуск сервиса MySQL с использованием планировщика заданий crond каждые 20 минут.

Анализ сервера:

- Установка на сервер агента системы мониторинга Zabbix для сбора следующих параметров в реальном времени:
 1. загрузка ЦПУ
 2. использование оперативной памяти
 3. использование swar пространства
 4. загрузка сетевого интерфейса (входящее направление)
 5. загрузка сетевого интерфейса (исходящее направление)
 6. количество свободного дискового пространства на корневом разделе ФС
 7. доступность сервера по icmp
 8. время непрерывной работы (uptime) сервера
 9. uptime сервиса MySQL
 10. количество «медленных запросов» (slow queries) MySQL в текущий момент момент
 11. количество MySQL-запросов в секунду (qps)
 12. количество MySQL-потоков в секунду (threads)
- Анализ файла конфигурации MySQL (/etc/my.cnf)

Явных ошибок в конфигурационном файле найдено не было. Единственный момент, на который обратили внимание – активировано ведение бинарных лог-файлов;
опция server-id=1
Бинарные логи должны быть выключены, если не применяется механизм MySQL репликации между серверами или если не применяется процедура восстановления после сбоя на основе бинарных логов. Запущенный механизм ведения бинарных лог-файлов значительно замедляет работу сервера (особенно, если эти логи физически пишутся на тот же самый дисковый раздел, где находятся файлы БД).
- Анализ системных настроек (которые потенциально могут являться причиной подобного поведения сервиса).
 1. Подозрение на ситуацию с нехваткой дискового пространства во временном каталоге MySQL (по умолчанию /tmp) было отброшено, т.к. вся система целиком установлена на один дисковый раздел:

```
[root@localhost ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup00-LogVol100
                145G  85G   53G   62% /
/dev/sda1       99M   40M   55M   42% /boot
none            1005M    0 1005M    0% /dev/shm
```

это некорректно с системной точки зрения (спонтанный рост объема данных базы приложения или любого лог-файла со 100% заполнением корневой фс приведет к останову всей системы), и ухудшает производительность СУБД (файлы данных, лог-файлы, временный каталог СУБД должны находиться на различных дисковых разделах, а в идеале на различных физических устройствах); однако, в данном случае это не может приводить к остановке сервиса MySQL.

2. Анализ лог-файла сервиса MySQL также не дал результатов, т.к. не обнаружены какие-либо системные ошибки а только сообщения о регулярных перезапусках сервиса и, в некоторых случаях, об отмененных операциях сортировки в связи с перезапуском.

- Мониторинг параметров сервера

В системе мониторинга был создан комплексный отчет с одновременным отображением графиков по ряду системных параметров и параметров сервиса MySQL (см. рисунок 1):

1. Параметр `cpu_idle` – количество свободных ЦПУ ресурсов [%]; шкала справа; синий цвет.
2. Параметр `memory_pcnt_freesize` – количество свободного ОЗУ[%]; шкала справа; голубой цвет.
3. Параметр `MySQL queries per second` – количество запросов в секунду; шкала справа; красный цвет;
4. Параметр `MySQL number of slow queries` – количество медленных запросов (по умолчанию MySQL считает медленным запрос, который выполняется более 10 секунд); шкала справа; черный цвет.
5. Параметр `MySQL number of threads` – количество потоков; шкала справа; желтый цвет.
6. параметр `MySQL uptime` – время непрерывной работы сервиса MySQL; шкала слева; зеленый цвет.

- Предварительный анализ результатов:

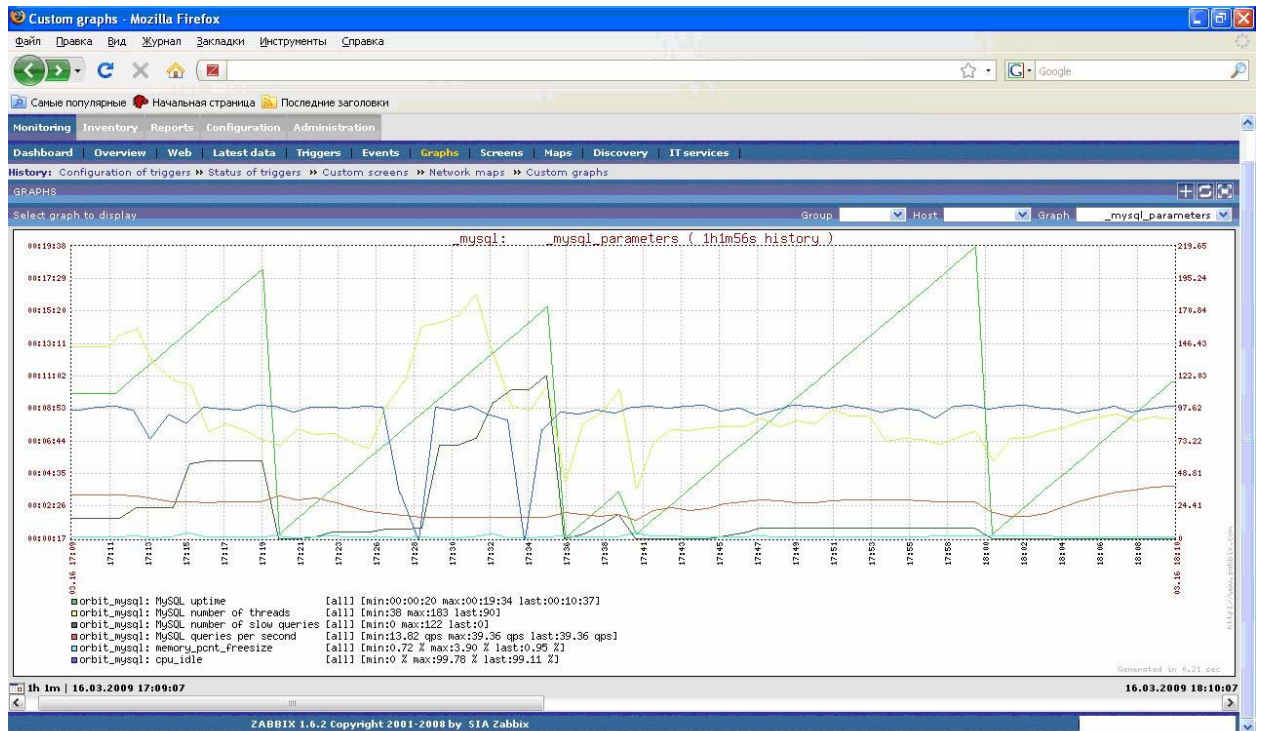


Рис. 1. Комплексный отчет с графиками значений основных параметров системы.

На графике наблюдаются моменты перезапуска сервиса MySQL (на пилообразном графике uptime зеленого цвета), а также поведение *всех* заявленных параметров. Предварительный вывод – существенное повышение загрузки ЦПУ при росте количества slow-queries.

- Поиск момента останова сервиса

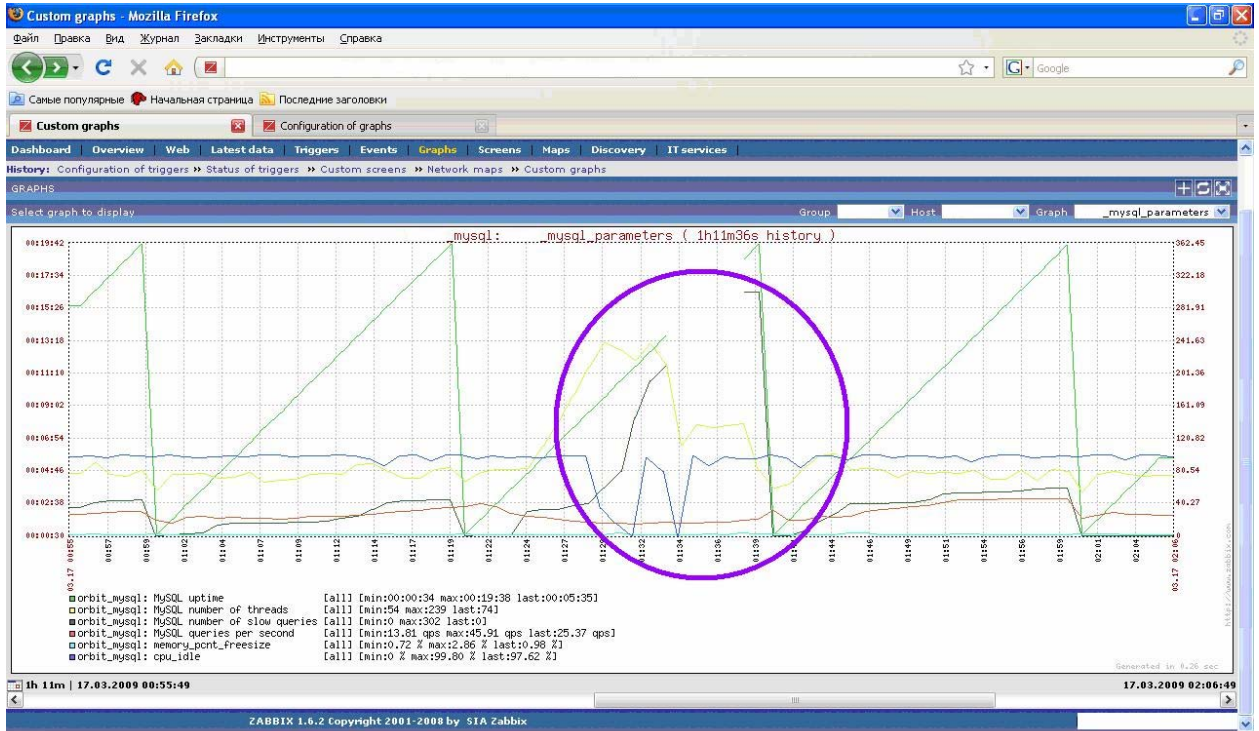


Рис. 2. Определение момента возникновения отказа в обслуживании.

Примерно в 01-28 начался рост количества slow-queries, одновременно с этим начала расти загрузка ЦПУ (см. черный и синий графики на рисунке 2). Далее значительный рост количества медленных запросов привел к 100% использованию ресурсов ЦПУ сервера, а также к тому, что агент системы мониторинга был не в состоянии отдать данные (разрывы в графиках на рисунке 2). При этом в течение рассматриваемого на рис.2 интервала времени (1 ч 11 мин) система мониторинга зафиксировала предельное количество медленных запросов, равное 302 (комментарий под графиками на рисунке 2).

- Анализ лог-файла медленных запросов.

«Медленные запросы» сервис MySQL фиксирует в файле /var/log/slow-queries.log

```
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 36 Lock_time: 0 Rows_sent: 11 Rows_examined: 41329
SELECT * FROM storage_comments WHERE file_id = 1011359 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:55:11
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 16 Lock_time: 0 Rows_sent: 11 Rows_examined: 130011
SELECT * FROM storage_comments WHERE file_id = 1599737 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:55:12
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 59 Lock_time: 0 Rows_sent: 11 Rows_examined: 85718
SELECT * FROM storage_comments WHERE file_id = 1612497 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:57:14
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 33 Lock_time: 0 Rows_sent: 11 Rows_examined: 104283
SELECT * FROM storage_comments WHERE file_id = 893360 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:57:34
# User@Host: example[example] @ [7.8.9.10]
```

```

# Query_time: 16 Lock_time: 0 Rows_sent: 11 Rows_examined: 85719
SELECT * FROM storage_comments WHERE file_id = 1612497 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:57:49
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 33 Lock_time: 0 Rows_sent: 11 Rows_examined: 104284
SELECT * FROM storage_comments WHERE file_id = 893360 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 11:58:02
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 12 Lock_time: 0 Rows_sent: 11 Rows_examined: 104285
SELECT * FROM storage_comments WHERE file_id = 893360 ORDER BY date DESC LIMIT 0, 11;
/usr/libexec/mysqld, Version: 4.1.20-log. started with:
Tcp port: 3306 Unix socket: /var/lib/mysql/mysql.sock
Time          Id Command  Argument
# Time: 090316 12:01:25
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 27 Lock_time: 0 Rows_sent: 11 Rows_examined: 39634
use example_users;
SELECT * FROM storage_comments WHERE file_id = 1646392 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 12:01:27
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 22 Lock_time: 0 Rows_sent: 5 Rows_examined: 748939
SELECT YX1.id, YX1.link, YX1.user_id, YX1.shared FROM storage_files as YX1 WHERE 1 AND
mimeclass='image' and shared > 0 and shared < 4 AND status != 'P' AND violence = 'No' AND password
= '' ORDER BY lastdate DESC LIMIT 0,5;
# Time: 090316 12:06:43
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 27 Lock_time: 0 Rows_sent: 11 Rows_examined: 104288
SELECT * FROM storage_comments WHERE file_id = 893360 ORDER BY date DESC LIMIT 0, 11;
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 21 Lock_time: 0 Rows_sent: 11 Rows_examined: 62408
SELECT * FROM storage_comments WHERE file_id = 827328 ORDER BY date DESC LIMIT 0, 11;
# Time: 090316 12:10:11
# User@Host: example[example] @ [7.8.9.10]
# Query_time: 16 Lock_time: 0 Rows_sent: 11 Rows_examined: 130011
SELECT * FROM storage_comments WHERE file_id = 1599737 ORDER BY date DESC LIMIT 0, 11;

```

Видно, что все «медленные запросы» – это чтение из таблицы storage_comments. Запрос формируется без индекса, что приводит к значительному времени выполнения каждого запроса в отдельности (59, 33, 36 секунд). Как результат, при наличии большого числа таких запросов (например, 302 запроса, как в рассмотренном выше случае) возможен отказ в обслуживании.

Действия по устранению проблемы (временные)

- По нашей рекомендации разработчики приложения временно приостановили операции чтения и записи из/в таблицу storage_comments.
- Был отключен 20-минутный регулярный перезапуск сервиса MySQL

Анализ результатов:

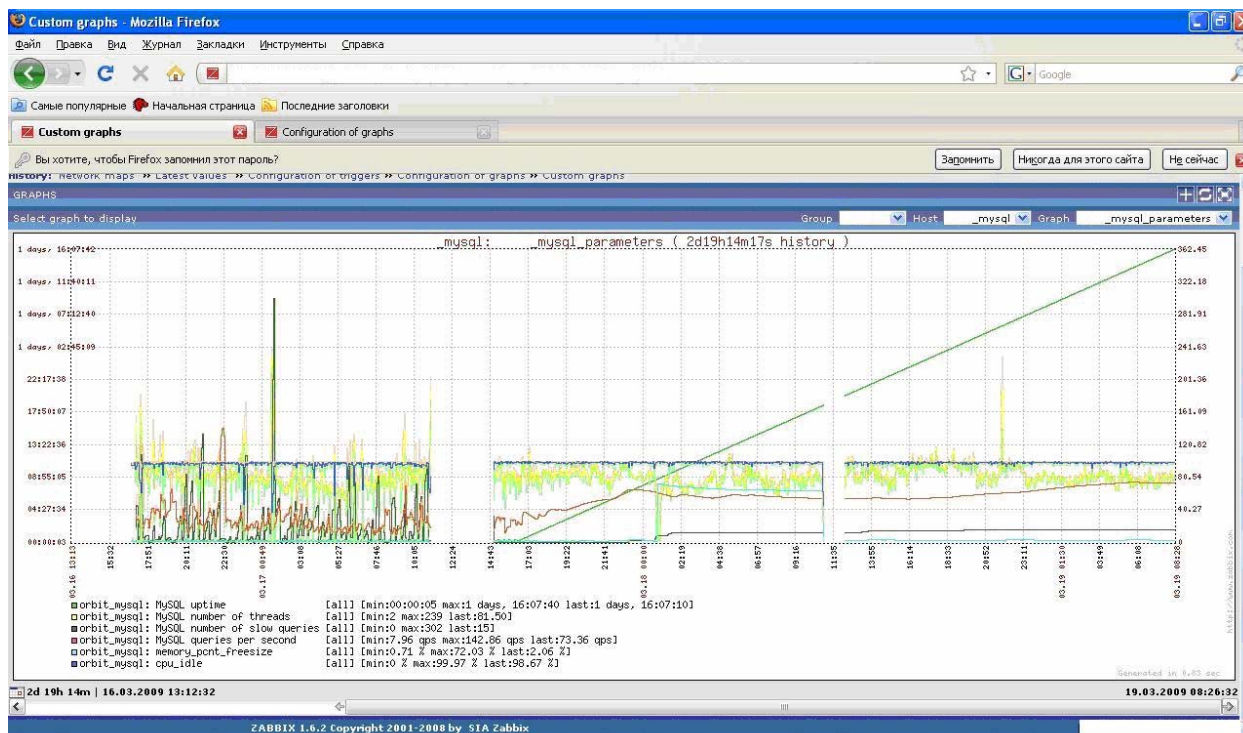


Рис. 3. Анализ результатов.

Согласно графику на рисунке 3 сервис MySQL функционирует без перезапусков и отказов обслуживания 1 день, 16 часов, 7 минут, что свидетельствует о корректности установления причины возникновения проблемы (два разрыва на графике связаны с автоматическим перезапуском фаервола на сервере СУБД и отсутствием связи между сервером системы мониторинга и его агентом на анализируемом сервере вследствие сетевых проблем).

Предпринятые действия по запрету операций чтения/записи из таблицы storage_comments являются временными. Для полного устранения подобных проблем необходимо проводить комплексный анализ и оптимизацию структуры БД, логики работы приложения и структуры MySQL запросов.

По возникшим по данному отчету вопросам просим обращаться по электронной почте:

support@futureservice.ru

Тел: 8 (48439) 90986

Отчет составлен 18.03.2009