查看：<http://www.linuxidc.com/Linux/2016-04/130546.htm>

一、问题描述

    今天运维同事找到我问，有两台数据库MYSQL的连接数超过监控的阈值（700），当然他们的监控（zabbix）也是刚刚搭建起来的。表示该值已经从很低的数值一点一点调上来了，但是仍然报警。于是就找到我，做为[Oracle](http://www.linuxidc.com/topicnews.aspx?tid=12)’s DBA的我，准备变为双料DBA，这正是学习的好时机。马上来学习一下，如何在MYSQL中查询连接数。

二、实验

1.show status  查看所有状态参数，其中Threads\_connected 当前的连接数，Connections 试图连接到（不管是否成功）MYSQL服务器的连接总数， Max\_used\_connections 服务器启动后已经同时使用过的连接最大数量（并发）。

mysql> show status like '%connect%';

+----------------------+---------+

| Variable\_name        | Value  |

+----------------------+---------+

| Aborted\_connects    | 163    |

| Connections          | 1116123 |

| Max\_used\_connections | 266    |

| Threads\_connected    | 208    |

+----------------------+---------+

4 rows in set (0.00 sec)

mysql>

2.show processlist  显示当前正在执行的mysql连接

mysql> show processlist;

+---------+----------+--------------------+-----------+---------+-------+-------+------------------+

| Id      | User    | Host              | db        | Command | Time  | State | Info            |

+---------+----------+--------------------+-----------+---------+-------+-------+------------------+

| 1105357 | tjuser  | 10.10.100.30:36210 | testdb  | Sleep  |  1377 |      | NULL            |

| 1112435 | tjuser  | 10.10.100.30:54112 | testdb  | Sleep  |  1616 |      | NULL            |

|…… 略

| 1116128 | tjuser | 10.10.100.21:47484 | testdb | Sleep  |    1 |      | NULL            |

| 1116129 | tjuser | 10.10.100.21:47485 | testdb | Sleep  |    64 |      | NULL            |

+---------+----------+--------------------+-----------+---------+-------+-------+------------------+

207 rows in set (0.00 sec)

3.mysqladmin -u -p -h status    显示当前mysql状态

[root@db ~]# mysqladmin -uroot -p -hlocalhost status

Enter password:

Uptime: 14604445  Threads: 208  Questions: 34034734  Slow queries: 179  Opens: 12553  Flush tables: 3  Open tables: 977  Queries per second avg: 2.330

[root@db ~]#

4.mysqladmin -u -p -h extended-status    显示mysql的其他状态

[root@db ~]# mysqladmin -uroot -p -hlocalhost extended-status

Enter password:

+------------------------------------------+--------------+

| Variable\_name                            | Value        |

+------------------------------------------+--------------+

| Aborted\_clients                          | 53041        |

| Aborted\_connects                        | 163          |

| Connections                              | 1116157      |

…… 略

| Threads\_connected                        | 206          |

| Threads\_created                          | 633          |

| Threads\_running                          | 1            |

| Uptime                                  | 14604661    |

| Uptime\_since\_flush\_status                | 14604661    |

+------------------------------------------+--------------+

[root@db ~]#

三、总结

    此次查询只需要掌握mysql中show status，show process list命令，以及命令mysqladmin。 通过上述命令可以快速得到MYSQL数据库连接参数与状态值。查询到数据库设置connect数为1000，告诉同事后修改了阈值后正常。

    It’s never too late to be what you might have been.

**本文永久更新链接地址**：<http://www.linuxidc.com/Linux/2016-04/130546.htm>

参考：<http://www.cnblogs.com/zhengah/p/4775565.html>

[**MYSQL 查看最大连接数和修改最大连接数**](http://www.cnblogs.com/zhengah/p/4775565.html)

MySQL查看最大连接数和修改最大连接数

1、查看最大连接数  
show variables like '%max\_connections%';  
2、修改最大连接数  
set GLOBAL max\_connections = 200;

  以下的文章主要是向大家介绍的是MySQL最大连接数的修改，我们大家都知道MySQL最大连接数的默认值是100, 这个数值对于并发连接很多的数据库的应用是远不够用的，当连接请求大于默认连接数后，就会出现无法连接数据库的错误，因此我们需要把它适当调大一些。在使 用MySQL数据库的时候，经常会遇到这么一个问题，就是“Can not connect to MySQL server. Too many connections”-mysql 1040错误，这是因为访问MySQL且还未释放的连接数目已经达到MySQL的上限。通常，mysql的最大连接数默认是100, 最大可以达到16384。

    常用的修改最大连接数的最常用的两种方式如下:

    第一种：命令行查看和修改最大连接数(max\_connections)。

  >mysql -uuser -ppassword(命令行登录MySQL)

    mysql>show variables like 'max\_connections';(查可以看当前的最大连接数)

    msyql>set global max\_connections=1000;(设置最大连接数为1000，可以再次查看是否设置成功)

    mysql>exit

    这种方式有个问题，就是设置的最大连接数只在mysql当前服务进程有效，一旦mysql重启，又会恢复到初始状态。因为mysql启动后的初始化工作是从其配置文件中读取数据的，而这种方式没有对其配置文件做更改。

    第二种：通过修改配置文件来修改mysql最大连接数(max\_connections)。

    这种方式说来很简单，只要修改MySQL配置文件my.ini 或 my.cnf的参数max\_connections，将其改为max\_connections=1000，然后重启MySQL即可。但是有一点最难的就是 my.ini这个文件在哪找。通常有两种可能，一个是在安装目录下，另一种是在数据文件的目录下，安装的时候如果没有人为改变目录的话，一般就在 C:/ProgramData/MySQL往下的目录下,linux系统中一般在/etc目录下。

**其他需注意的：**

   在编程时，由于用MySQL语句调用数据库时，在每次之执行语句前，会做一个临时的变量用来打开数据库，所以你在使用MySQL语句的时候，记得在每次调用完MySQL之后就关闭MySQL临时变量。

    另外对于访问量大的，可以考虑直接写到文本中，根据预测的访问量，先定义假若是100个文本文件名,需要的时候，再对所有文本文件中的数据进行分析，再导入数据库。

如果是root帐号，你能看到所有用户的当前连接。如果是其它普通帐号，只能看到自己占用的连接

怎么进入mysql命令行呢？ mysql的安装目录下面有个bin目录，先用命令行进入该目录，然后用 mysql -uroot -p123456 来登录（注意：用户名和密码不用包含“”）

命令： show processlist;   
如果是root帐号，你能看到所有用户的当前连接。如果是其它普通帐号，只能看到自己占用的连接。

show processlist;只列出前100条，如果想全列出请使用show full processlist;   
mysql> show processlist;

命令： show status;  
命令：show status like '%下面变量%';   
Aborted\_clients                由于客户没有正确关闭连接已经死掉，已经放弃的连接数量。   
Aborted\_connects            尝试已经失败的MySQL服务器的连接的次数。   
Connections                     试图连接MySQL服务器的次数。   
Created\_tmp\_tables          当执行语句时，已经被创造了的隐含临时表的数量。   
Delayed\_insert\_threads     正在使用的延迟插入处理器线程的数量。   
Delayed\_writes                用INSERT DELAYED写入的行数。   
Delayed\_errors                用INSERT DELAYED写入的发生某些错误(可能重复键值)的行数。   
Flush\_commands            执行FLUSH命令的次数。   
Handler\_delete                 请求从一张表中删除行的次数。   
Handler\_read\_first          请求读入表中第一行的次数。   
Handler\_read\_key           请求数字基于键读行。   
Handler\_read\_next           请求读入基于一个键的一行的次数。   
Handler\_read\_rnd           请求读入基于一个固定位置的一行的次数。   
Handler\_update              请求更新表中一行的次数。   
Handler\_write                 请求向表中插入一行的次数。   
Key\_blocks\_used            用于关键字缓存的块的数量。   
Key\_read\_requests         请求从缓存读入一个键值的次数。   
Key\_reads                      从磁盘物理读入一个键值的次数。   
Key\_write\_requests         请求将一个关键字块写入缓存次数。   
Key\_writes                      将一个键值块物理写入磁盘的次数。   
Max\_used\_connections    同时使用的连接的最大数目。   
Not\_flushed\_key\_blocks   在键缓存中已经改变但是还没被清空到磁盘上的键块。   
Not\_flushed\_delayed\_rows      在INSERT DELAY队列中等待写入的行的数量。   
Open\_tables                  打开表的数量。   
Open\_files                     打开文件的数量。   
Open\_streams               打开流的数量(主要用于日志记载）   
Opened\_tables              已经打开的表的数量。   
Questions                     发往服务器的查询的数量。   
Slow\_queries                要花超过long\_query\_time时间的查询数量。   
Threads\_connected       当前打开的连接的数量。   
Threads\_running          不在睡眠的线程数量。   
Uptime                        服务器工作了多少秒。