# 问题描述

|  |  |
| --- | --- |
| 事务1 | 事务2 |
| begin; | begin; |
| update tb set pk=2 where pk=1; |  |
|  | update tb set pk=2 where pk=1; |
|  | 锁超时 |
|  |  |

# 处理步骤

在出现锁等待的时候，需要调查清楚锁等待涉及的事务，查到对应的SQL操作。

**监控锁状态：**

1. 查看是否有锁等待

SHOW STATUS LIKE ‘innodb\_row\_lock%’;

**关注点：**

Innodb\_row\_lock\_current\_waits：当前有多少锁等待

Innodb\_row\_lock\_waits：一共发生过多少锁等待

1. 查看哪个事务在等待（被阻塞了）

SELECT \* FROM information\_schema.INNODB\_TRX

WHERE trx\_state=’LOCK WAIT’;

**关注点：**

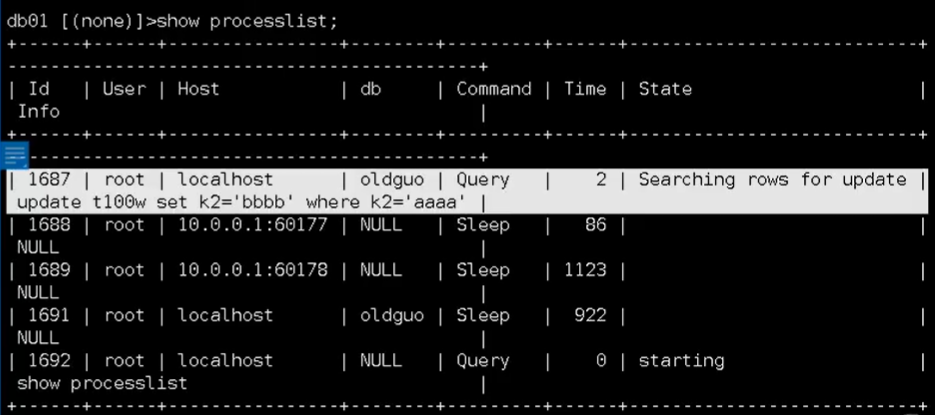
trx\_id：事务ID号

trx\_state：当前事务的状态

trx\_mysql\_thread\_id：连接层，连接线程ID（SHOW PROCESSLIST->Id或trx\_id）

trx\_query：当前被阻塞的操作（一般是要丢给开发的）

通过步骤2中查询的trx\_id可以查看show processlist中输出的事务ID对应的操作情况（可以看到具体操作，但是不知道是不是阻塞）。



1. 查看锁源（谁锁定的）

SELECT \* FROM sys.innodb\_lock\_waits; //被锁的和锁定它的之间关系

**关注点：**

locked\_table：哪张表出现的等待

waiting\_trx\_id：等待的事务（与上述trx\_id对应）

waiting\_pid：等待的线程号（与上述trx\_mysql\_thread\_id对应）

blocking\_trx\_id：锁源的事务ID

blocking\_pid：锁源的线程号

1. 找到锁源的thread\_id

SELECT \* FROM performance\_schema.threads WHERE processlist\_id=204;

1. 找到锁源的SQL语句

查看在执行的语句：

SELECT \* FROM performance\_schema.`event\_statements\_current` WHERE thread\_id=41;

执行语句的历史：

SELECT \* FROM performance\_schema.`events\_statement\_history` WHERE thread\_id=41;