[MYSQL] 服务器出现大量的TIME_WAIT, 每天凌晨就清零了

原创 大大刺猬 大大刺猬 2025年04月01日 18:10 上海

背景

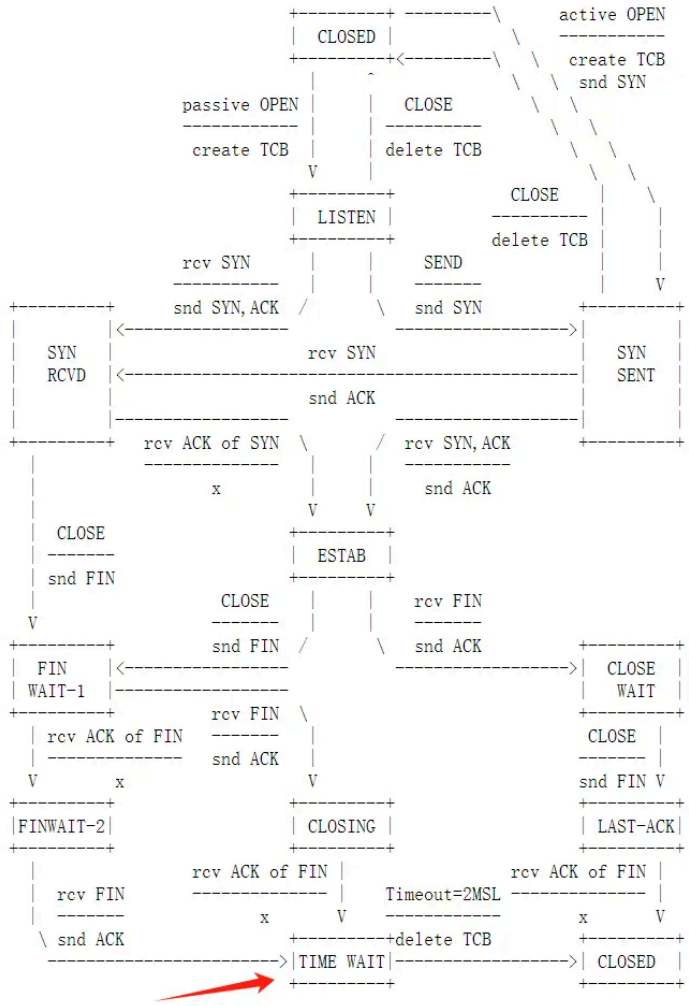
某数据库服务器发现存在**大量处于TIME_WAIT状态的tcp连接**, 但是mysql数据库里面的连接不到100, 应用服务器处于TIME_WAIT的tcp连接更是达到了几万, 连接的端口都是mysql服务器的3306, 也就是这些连接活着的时候都是连接的数据库. 而**每天凌晨的时候这些TIME WAIT的连接就都没了**.

分析

首先我们使用 `man netstat` 查看下TIME_WAIT是个啥状态. 这里稍汇总了下:

column1	column2
ESTABLISHED	The socket has an established connection
SYN_SENT	The socket is actively attempting to establish a connection
SYN_RECV	A connection request has been received from the network
FIN_WAIT1	The socket is closed, and the connection is shutting down
FIN_WAIT2	Connection is closed, and the socket is waiting for a shutdown from the remote end
TIME_WAIT	The socket is waiting after close to handle packets still in the network
CLOSE	The socket is not being used
CLOSE_WAIT	The remote end has shut down, waiting for the socket to close
LAST_ACK	The remote end has shut down, and the socket is closed. Waiting for acknowledgement
LISTEN	The socket is listening for incoming connections. Such sockets are not included in the output unless you specify the --listening (-l) or --all (-a) option
CLOSING	Both sockets are shut down but we still don’t have all our data sent
UNKNOWN	The state of the socket is unknown.

也就是说TIME_WAIT状态是在CLOSED之前的一个状态,比如是刚发完ACK之后的状态. 完整的状态变化过程我们可以查看相关的rfc文档, 其示意图如下:

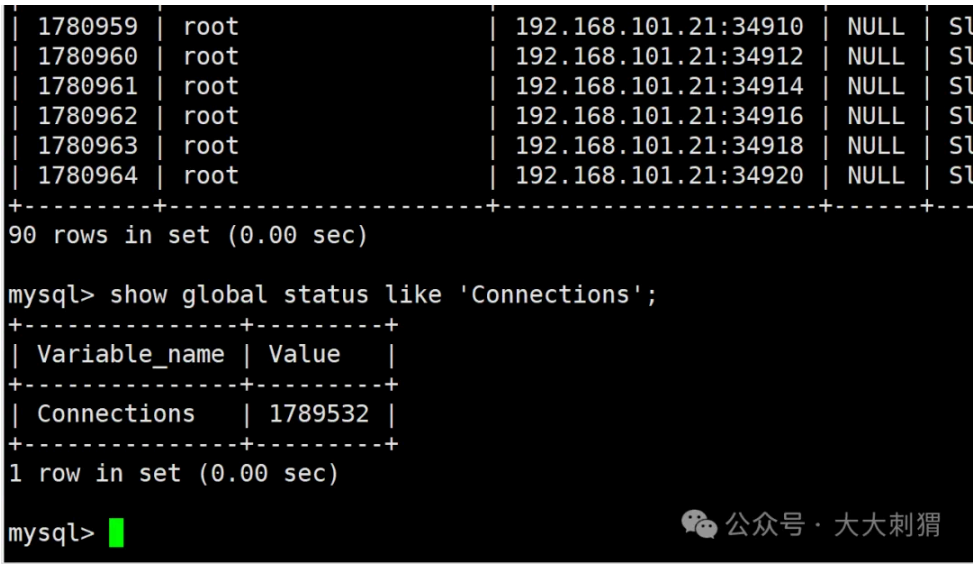


TCP Connection State Diagram
Figure 6.

也就是说在关闭tcp连接了, 但未关闭完成, 而这么大的量, 说明在频繁的断开连接, 也就是还存在频繁的建立连接. 也就是说应用使用的是**短连接**! 我们可以登录数据库,执行如下sql确认

```
-- 查看一共的连接次数
show global status like 'Connections';

-- 查看当前的连接的id 绝大部分的id应该都是接近Connections值的. 表明都是新连接
show processlist;
```

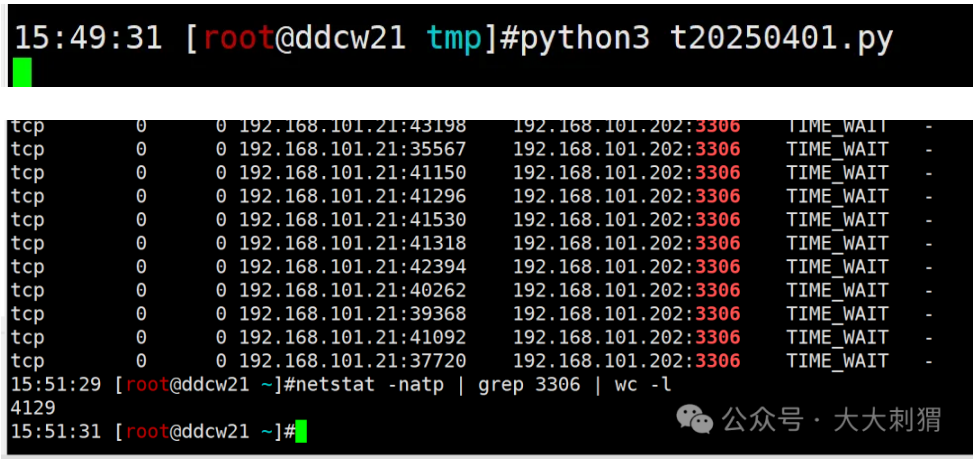


我们还可以查看下mysql的error日志,
应该能在日志里面发现大量的 [Note] Got an error reading communication packets 信息,
而且应该很少有 [Note] Aborted connection 2599805 to db 之类的信息.(异常断开连接太多的话, 是很难有TIME WAIT状态的连接的, 而我们本次环境有大量的TIME WAIT连接, 说明是很多短连接正常断开的.)

每天凌晨的时候TIME WAIT的连接清零应该就是应用重启了一波. 我们可以使用 ps -ef 查看进程的启动时间确定.

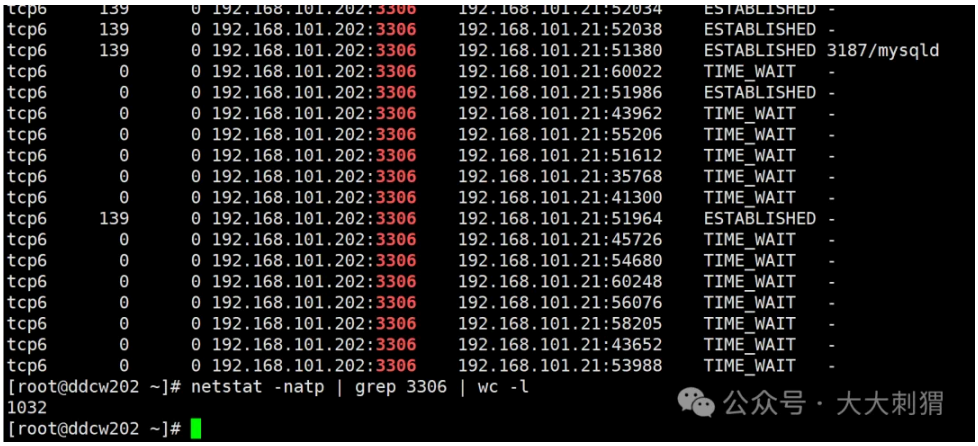
复现

既然原因知道了, 那我们就复现验证下吧. 在应用服务器上执行测试脚本模拟大量的短连接(见文末),然后查看连接情况

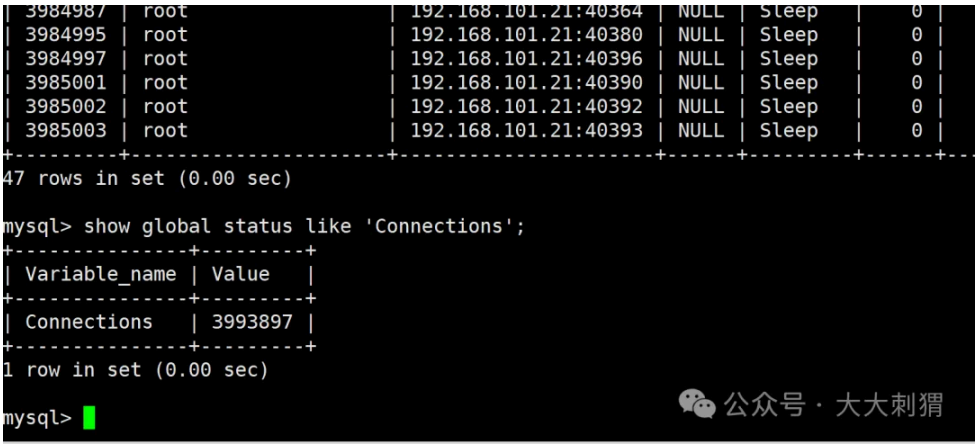


发现确实存在大量的TIME_WAIT的连接

然后我们在数据库服务器查看tcp连接



发现数据库也有不少处于TIME WAIT的连接. 我们再查看下数据库里面的连接情况:



最后我们停止测试脚本, 再观察下, TIME WAIT的连接是否会"清零"

发现连接数都降下来了, 毕竟连接都没了, 连接相关的socket资源之类的肯定也是回收了的

> 如果复现的时候未出现大量TIME WAIT, 则需要加大并发, 或者调整下相关内核参数(net.ipv4.tcp_tw_reuse和net.ipv4.tcp_tw_reuse)

总结

关于"服务器出现大量的TIME_WAIT, 每天凌晨就清零了"的结论就是:

1. 应用使用大量的短连接.

2. 每天凌晨重启了应用.

参考:
https://www.rfc-editor.org/rfc/rfc793

附脚本:

```
import pymysql
import time
from multiprocessing import Process
def testconn():
        conn = pymysql.connect(
                host='192.168.101.202',
                port=3306,
                user='root',
                password='123456',
                )
        cursor = conn.cursor()
        cursor.execute('select 1+1')
        conn.close()

def testrun():
        while True:
                testconn()
        #time.sleep(0.1)

maxconn = 200
p = {}
for i in range(maxconn):
        p[i] = Process(target=testrun,)
for i in range(maxconn):
        p[i].start()
for i in range(maxconn):
        p[i].join()

```