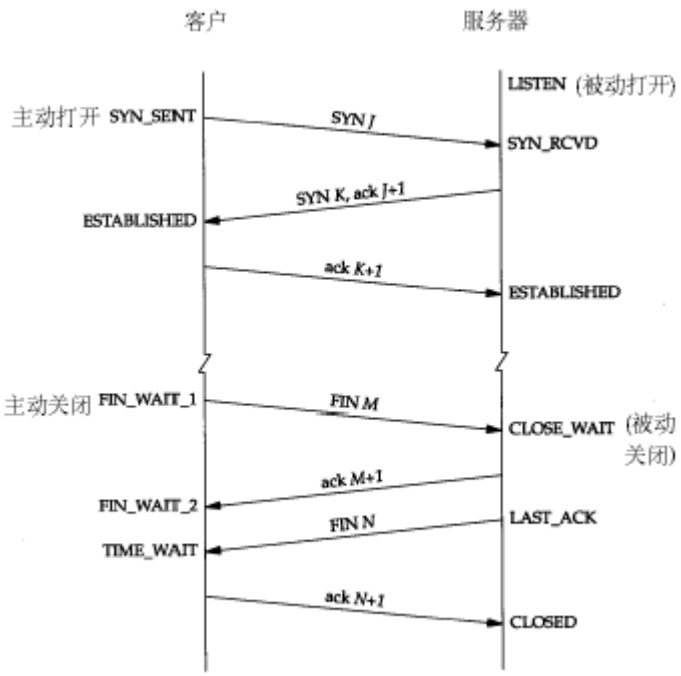
服务器网络存在大量TIME_WAIT和CLOSE_WAIT详解和解决办法



在服务器的日常维护过程中，会经常用到下面的命令：

```
netstat -n | awk '/^tcp/ {++S[$NF]} END {for(a in S) print a, S[a]}'
```

它会显示例如下面的信息：

```
TIME_WAIT 814

CLOSE_WAIT 1

FIN_WAIT1 1

ESTABLISHED 634

SYN_RECV 2

LAST_ACK 1
```

常用的三个状态是：ESTABLISHED 表示正在通信，TIME_WAIT 表示主动关闭，CLOSE_WAIT 表示被动关闭。

这么多状态不用都记住，只要了解到我上面提到的最常见的三种状态的意义就可以了。一般不到万不得已的情况也不会去查看网络状态，如果服务器出了异常，百分之八九十都是下面两种情况：

1.服务器保持了大量TIME_WAIT状态

2.服务器保持了大量CLOSE_WAIT状态

因为linux分配给一个用户的文件句柄是有限的，而TIME_WAIT和CLOSE_WAIT两种状态如果一直被保持，那么意味着对应数目的通道就一直被占着，而且是“占着茅坑不使劲”，一旦达到句柄数上限，新的请求就无法被处理了，接着就是大量Too Many Open Files异常，tomcat崩溃。。

1.服务器保持了大量TIME_WAIT状态

这种情况比较常见，一些爬虫服务器或者WEB服务器（如果网管在安装的时候没有做内核参数优化的话）上经常会遇到这个问题，这个问题是怎么产生的呢？

爬虫服务器完成一个爬去服务后会主动关闭连接进入time_wait状态。

解决思路很简单，就是让服务器能够快速回收和重用那些TIME_WAIT的资源。

下面来看一下我们网管对/etc/sysctl.conf文件的修改：

```
#对于一个新建连接，内核要发送多少个 SYN 连接请求才决定放弃,不应该大于255，默认值是5，对应于180秒左右时间

net.ipv4.tcp_syn_retries=2
#net.ipv4.tcp_synack_retries=2
#表示当keepalive起用的时候，TCP发送keepalive消息的频度。缺省是2小时，改为300秒
net.ipv4.tcp_keepalive_time=1200
net.ipv4.tcp_orphan_retries=3
#表示如果套接字由本端要求关闭，这个参数决定了它保持在FIN-WAIT-2状态的时间
net.ipv4.tcp_fin_timeout=30
#表示SYN队列的长度，默认为1024，加大队列长度为8192，可以容纳更多等待连接的网络连接数。
net.ipv4.tcp_max_syn_backlog = 4096
#表示开启SYN Cookies。当出现SYN等待队列溢出时，启用cookies来处理，可防范少量SYN攻击，默认为0，表示关闭

net.ipv4.tcp_syncookies = 1

#表示开启重用。允许将TIME-WAIT sockets重新用于新的TCP连接，默认为0，表示关闭
net.ipv4.tcp_tw_reuse = 1
#表示开启TCP连接中TIME-WAIT sockets的快速回收，默认为0，表示关闭
net.ipv4.tcp_tw_recycle = 1

##减少超时前的探测次数
net.ipv4.tcp_keepalive_probes=5
##优化网络设备接收队列
net.core.netdev_max_backlog=3000
```

修改完之后执行/sbin/sysctl -p让参数生效。

2.服务器保持了大量CLOSE_WAIT状态

换句话说，就是在对方连接关闭之后，程序里没有检测到，或者程序压根就忘记了这个时候需要关闭连接，于是这个资源就一直被程序占着。个人觉得这种情况，通过服务器内核参数也没办法解决，服务器对于程序抢占的资源没有主动回收的权利，除非终止程序运行。

所以如果将大量**CLOSE_WAIT**的解决办法总结为一句话那就是：查代码。因为问题出在服务器程序里头啊。