　linux作为服务器系统，当运行高并发TCP程序时，通常会出现连接建立到一定个数后不能再建立连接的情况

　　本人在工作时，测试高并发tcp程序（GPS服务器端程序），多次测试，发现每次连接建立到3800左右时，再也不能建立tcp连接，最总上网搜索，参考：[http://blog.csdn.net/guowake/article/details/6615728](http://blog.csdn.net/guowake/article/details/6615728" \t "http://blog.csdn.net/liwateryi/article/details/_blank)解决了连接限制的问题，此处记录下来，方便日后工作继续使用，虽然参考博文中内容有点多，文中所描述的内容也不怎么明白，但总结下来，按如下几步操作即可（我映像中，按此方法设置后，测试建立连接8000多时也未出现什么错误） ：

* 第一步，修改/etc/security/limits.conf文件，在文件中添加如下行(\*指代系统用户名)，修改Linux系统对用户的关于打开文件数的软限制和硬限制：

　　　　\* soft nofile 20000

　　　　\* hard nofile 20000

* 第二步，修改/etc/pam.d/login文件，在文件中添加如下行：

　　　　session required /lib/security/pam\_limits.so

　　　　如果是64bit系统的话，应该为 :  
　　　　session required /lib64/security/pam\_limits.so

* 第三步，修改/etc/sysctl.conf文件，在文件中(清楚文件原始内容)添加如下行（修改网络内核对TCP连接的有关限制）：

IMG_256

net.ipv4.ip\_local\_port\_range = 1024 65535

net.core.rmem\_max=16777216

net.core.wmem\_max=16777216

net.ipv4.tcp\_rmem=4096 87380 16777216

net.ipv4.tcp\_wmem=4096 65536 16777216

net.ipv4.tcp\_fin\_timeout = 10

net.ipv4.tcp\_tw\_recycle = 1

net.ipv4.tcp\_timestamps = 0

net.ipv4.tcp\_window\_scaling = 0

net.ipv4.tcp\_sack = 0

net.core.netdev\_max\_backlog = 30000

net.ipv4.tcp\_no\_metrics\_save=1

net.core.somaxconn = 262144

net.ipv4.tcp\_syncookies = 0

net.ipv4.tcp\_max\_orphans = 262144

net.ipv4.tcp\_max\_syn\_backlog = 262144

net.ipv4.tcp\_synack\_retries = 2

net.ipv4.tcp\_syn\_retries = 2

IMG_257

* 第四步，执行如下命令（使上述设置生效）：

/sbin/sysctl -p /etc/sysctl.conf

/sbin/sysctl -w net.ipv4.route.flush=1

* 第五步，执行如下命令（linux系统优化完网络必须调高系统允许打开的文件数才能支持大的并发，默认1024是远远不够的）：

　echo ulimit -HSn 65536 >> /etc/rc.local

　echo ulimit -HSn 65536 >>/root/.bash\_profile

　ulimit -HSn 65536

* 第六步，重启机器。