```
# Global settings
#---------------------------------------------------------------------
global
# to have these messages end up in /var/log/haproxy.log you will
# need to:
#
# 1) configure syslog to accept network log events. This is done
# by adding the '-r' option to the SYSLOGD_OPTIONS in
# /etc/sysconfig/syslog
#
# 2) configure local2 events to go to the /var/log/haproxy.log
# file. A line like the following can be added to
# /etc/sysconfig/syslog
#
# local2.* /var/log/haproxy.log
#
log 127.0.0.1 local3
chroot  /usr/local/haproxy
pidfile /var/run/haproxy.pid
maxconn 5000
user haproxy
group haproxy
daemon

# turn on stats unix socket
stats socket /var/lib/haproxy/stats
#---------------------------------------------------------------------
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#---------------------------------------------------------------------
defaults
#mode http
mode tcp
log global
#option httplog
```

```
#采用tcp日志格式
option tcplog
option dontlognull
#当服务器负载很高的时候，自动结束掉当前队列处理比较久的链接
#option abortonclose
#每次请求完毕后主动关闭http通道,haproxy不支持keep-alive,只能模拟这种模式的实现
#option http-server-close
#如果后端服务器需要获得客户端真实ip需要配置的参数，可以从Http Header中获得客户端ip
#option forwardfor except 127.0.0.0/8
option redispatch
retries 3
#默认http请求超时时间
timeout http-request 10s
#默认队列超时时间，后端服务器在高负载时，会将haproxy发来的请求放进一个队列中.
timeout queue 1m
#haproxy与后端服务器连接超时时间.
timeout connect 30m
#客户端与haproxy连接后，数据传输完毕，不再有数据传输，即非活动连接的超时时间
timeout client 30m
#haproxy与后端服务器非活动连接的超时时间.
timeout server 30m
#默认新的http请求连接建立的超时时间，时间较短时可以尽快释放出资源，节约资源.
timeout http-keep-alive 10s
#心跳检测超时时间
timeout check 10s
maxconn 5000

#---------------------------------------
# main frontend which proxys to the backends
#---------------------------------------

frontend Qd-front
bind *:3306
mode tcp
```

```
default_backend Qd-Cluster

frontend stats-front
bind *:8080
mode http
default_backend stats-back

frontend Qd-onenode-front
bind *:3307
mode tcp
default_backend Qd-onenode-Cluster
#--------------------------------------
# static backend for serving up images, stylesheets and such
#--------------------------------------
backend Qd-Cluster
mode tcp
balance leastconn
option httpchk
#rise 2是2次正确认为服务器可用，fall 3是3次失败认为服务器不可用，weight代表权重
#cookie 1表示serverid为1，check inter 1500 是检测心跳频率
server M1 192.168.1.76:3306 check port 9200 inter 12000 rise 3 fall 3
server S01 192.168.1.74:3306 check port 9200 inter 12000 rise 3 fall 3
server S02 192.168.1.73:3306 check port 9200 inter 12000 rise 3 fall 3
server S03 192.168.1.64:3306 check port 9200 inter 12000 rise 3 fall 3
server S04 192.168.1.65:3306 check port 9200 inter 12000 rise 3 fall 3
server S05 192.168.1.66:3306 check port 9200 inter 12000 rise 3 fall 3
server S06 192.168.1.68:3306 check port 9200 inter 12000 rise 3 fall 3

backend stats-back
mode http
balance roundrobin
stats refresh 15s
stats uri /haproxy/stats
stats realm RongLian\ RongLian
stats auth admin:123456
```

```
#--------------------------------------
# round robin balancing between the various backends
#--------------------------------------
backend Qd-onenode-Cluster
mode tcp
balance leastconn
option httpchk
server M1 192.168.1.76:3306 check port 9200 inter 12000 rise 3 fall 3
server S01 192.168.1.74:3306 check port 9200 inter 12000 rise 3 fall 3 backup
server S02 192.168.1.73:3306 check port 9200 inter 12000 rise 3 fall 3 backup
server S03 192.168.1.64:3306 check port 9200 inter 12000 rise 3 fall 3 backup
server S04 192.168.1.65:3306 check port 9200 inter 12000 rise 3 fall 3 backup
server S05 192.168.1.66:3306 check port 9200 inter 12000 rise 3 fall 3 backup
server S06 192.168.1.68:3306 check port 9200 inter 12000 rise 3 fall 3 backup
```