**MySQL Server Load-Balancing with HAProxy**

Reference Links:

1. <http://www.fromdual.com/making-haproxy-high-available-for-mysql-galera-cluster>
2. <https://www.youtube.com/watch?v=525iJTWxFWc>

HAProxy (High Availability Proxy) is an open-source load-balancer which can load balance any TCP service. HAProxy is a free, very fast and reliable solution that offers load-balancing, high-availability, and proxying for TCP and HTTP-based applications. It is particularly well suited for very high traffic web sites and powers many of the world's most visited ones.

Ubuntu 14.04 and install it by:

apt-get install haproxy

You can check the version by:

haproxy -v

We need to enable HAProxy to be started by the init script /etc/default/haproxy. Set ENABLED option to 1 as:

ENABLED=1

To verify if this change is done properly, execute the init script of HAProxy without any parameters. You should see the following:

$ service haproxy <press\_tab\_key>

reload restart start status stop

cp /etc/haproxy/haproxy.cfg /etc/haproxy/haproxy.cfg.bak

nano /etc/haproxy/haproxy.cfg

#

# /etc/haproxy/haproxy.cfg

#

#---------------------------------------------------------------------

# 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 local2

chroot /var/lib/haproxy

pidfile /var/run/haproxy.pid

maxconn 1020 # See also: ulimit -n

user haproxy

group haproxy

daemon

# turn on stats unix socket

stats socket /var/lib/haproxy/stats.sock mode 600 level admin

stats timeout 2m

#---------------------------------------------------------------------

# common defaults that all the 'frontend' and 'backend' sections will

# use if not designated in their block

#---------------------------------------------------------------------

defaults

mode tcp

log global

option dontlognull

option redispatch

retries 3

timeout queue 45s

timeout connect 5s

timeout client 1m

timeout server 1m

timeout check 10s

maxconn 1020

#---------------------------------------------------------------------

# HAProxy statistics backend

#---------------------------------------------------------------------

listen haproxy-monitoring \*:80

mode http

stats enable

stats show-legends

stats refresh 5s

stats uri /

stats realm Haproxy\ Statistics

stats auth monitor:AdMiN123

stats admin if TRUE

frontend haproxy1 # change on 2nd HAProxy

bind \*:3306

default\_backend galera-cluster

backend galera-cluster

balance roundrobin

server nodeA 192.168.2.161:3306 maxconn 151 check

server nodeB 192.168.2.103:3306 maxconn 151 check

server nodeC 192.168.1.63:3306 maxconn 151 check

------------------------------------

service haproxy start

Go to MySQL Server 1 and 2:

Permission for access root user externally:

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY 'mysql';

FLUSH PRIVILEGES;

Test HaProxy:

Mysql -h 127.0.0.1 -u root -p

Check connected SRV details: SHOW GLOBAL VARIABLES LIKE '%host%' ;