**Galera Cluster is a synchronous multi-master database cluster, based on**

**synchronous replication and Oracle’s MySQL/InnoDB. When Galera Cluster**

**is in use, you can direct reads and writes to any node, and you can lose**

**any individual node without interruption in operations and without the**

**need to handle complex failover procedures.**

**HAProxy is a very fast and reliable solution for high**

**availability, load balancing, It supports TCP and HTTP-based applications.**

**Nowadays maximizing websites up-time is very crucial for heavy traffic websites.**

**This is not possible with a single server setup. Then we need some high availability**

**environment which can easily manage with a single server failure.**

**Percona XtraDB Cluster with Haproxy**

\*percona1 192.168.72.91 Cluster nodes

\*percona2 192.168.72.92 Cluster nodes

\*haproxy 192.168.72.95 Load Balancer node

**configure Network**

#vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

#vi /etc/hosts

#vi /etc/resolv.conf

#service network restart

#ping 192.168.72.1

\*\*\*\*diabled selinux

#vi /etc/sysconfig/selinux

**SELINUX=disabled**

#init 6

**Add ports**

#firewall-cmd --permanent --add-port={3306/tcp,4444/tcp,4567/tcp,4568/tcp}

#firewall-cmd --reload

**install percona repository**

#yum install https://repo.percona.com/yum/percona-release-latest.noarch.rpm

#vi /etc/yum.repos.d/percona-original-release.repo

gpgcheck=0

gpgcheck=0

#yum install Percona-XtraDB-Cluster-57 -y

#systemctl start mysqld

**Run the below command for get the temprory password**

#sudo grep 'temporary password' /var/log/mysqld.log

**u will get temp pass**

^VFU(:>">^JH

**like that**

**then enter in mysql**

#mysql -u root -p

password: {past above temprory password}

mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'temp';

mysql> exit

#systemctl stop mysqld

#vi /etc/my.cnf

[mysqld]

wsrep\_provider=/usr/lib64/galera3/libgalera\_smm.so

wsrep\_cluster\_name=pxc-cluster

wsrep\_cluster\_address=gcomm://192.168.72.91,192.168.72.92

wsrep\_node\_name=pxc1

wsrep\_node\_address=192.168.72.91

wsrep\_sst\_method=xtrabackup-v2

wsrep\_sst\_auth=sstuser:temp

pxc\_strict\_mode=ENFORCING

binlog\_format=ROW

default\_storage\_engine=InnoDB

innodb\_autoinc\_lock\_mode=2

**follow this step on both nodes afte that we need to configure only first nodes**

#systemctl start mysql@bootstrap

#mysql -u root -p

mysql> CREATE USER 'sstuser'@'localhost' IDENTIFIED BY 'temp';

mysql>GRANT RELOAD, LOCK TABLES, PROCESS, REPLICATION CLIENT ON \*.\* TO 'sstuser'@'localhost';

mysql>FLUSH PRIVILEGES;

**after that we need to start mysql on second node**

#systemctl start mysqld

#mysql -u root -p

mysql> show status like '%wsrep%';

**u can also check by creating database on first node and show database on second node**

**if its work then ur pxc is completed....**

**After completd percona we have to configure some more steps for HAProxy**

**On any cluster node, execute mysql queries to create a user for cluster check.**

# mysql -u root -p

mysql>GRANT PROCESS ON \*.\* TO 'clustercheckuser'@'localhost' IDENTIFIED BY 'clustercheckuserpass';

mysql> FLUSH PRIVILEGES;

**On all cluster nodes(percona1, percona2), run the following commands.**

# apt-get install xinetd

#service xinetd restart

**Open the file /etc/services**

# vi /etc/services

**Append the following lines to the end of /etc/services.**

mysqlchk 9200/tcp # mysqlchk

**for check the cluster is sync or not for haproxy we need to run command**

#clustercheckuser

**On the load balancer node (haproxy), run the following commands to install haproxy**.

# apt-get install haproxy

Configure /etc/haproxy/haproxy.cfg with the content below.

# vi /etc/haproxy/haproxy

global

log 127.0.0.1 local0

log 127.0.0.1 local1 notice

maxconn 4096

chroot /usr/share/haproxy

user haproxy

group haproxy

daemon

defaults

log global

mode http

option tcplog

option dontlognull

retries 3

option redispatch

maxconn 2000

contimeout 5000

clitimeout 50000

srvtimeout 50000

frontend pxc-front

bind \*:3306

mode tcp

default\_backend pxc-back

frontend stats-front

bind \*:8080

mode http

default\_backend stats-back

backend pxc-back

mode tcp

balance leastconn

option httpchk

server percona1 192.168.72.91:3306 check port 9200 inter 12000 rise 3 fall 3

server percona2 192.168.72.92:3306 check port 9200 inter 12000 rise 3 fall 3

backend stats-back

mode http

balance roundrobin

stats enable

stats uri /haproxy?stats

stats auth abhi:temp

# haproxy -f /etc/haproxy/haproxy.cfg

**[for check the syntax]**

#service haproxy restart

**we need to stop firewall**

#systemctl stop firewalld

**also we need to check ports are listen or not**

#netstat -ntl

**if there is all ok then**

**Verify if haproxy works.**

**open the browser and type**

'http://192.168.72.95:8080/haproxy?stats'

**in the address bar. u'll see the response content containing tables**.

**\_\_\_\_\_\_\_\_Finished\_\_\_\_\_\_\_\_**