

## MapR Bigdata Cluster Configuration

Node	CLDB	ZOOKEEPER	NFS	WEBSERVER/APIserver	Fileserver
Master01	✓		✓	✓	✓
S1		✓	✓		✓
S2		✓	✓		✓
S3		✓	✓		✓
S4			✓		✓
S5			✓		✓

```
cat /etc/hosts
```

```
echo " 10.142.0.8 master01.c.voltaic-racer-208109.internal master01" >> /etc/hosts
```

```
echo " 10.142.0.9 s1.c.voltaic-racer-208109.internal s1" >> /etc/hosts
```

```
echo " 10.142.0.10 s2.c.voltaic-racer-208109.internal s2" >> /etc/hosts
```

```
echo " 10.142.0.11 s3.c.voltaic-racer-208109.internal s3" >> /etc/hosts
```

```
echo " 10.142.0.12 s4.c.voltaic-racer-208109.internal s4" >> /etc/hosts
```

```
echo " 10.142.0.13 s5.c.voltaic-racer-208109.internal s5" >> /etc/hosts
```

```
echo " 10.142.0.14 nhost6.c.voltaic-racer-208109.internal nhost6" >> /etc/hosts
```

```
echo " 10.142.0.3 s6.c.voltaic-racer-208109.internal s6" >> /etc/hosts
```

```
yum update -y
```

```
yum install -y curl device-mapper iputils libsysfs lvm2 nc nfs-utils ntp nss openssl python-devel sdparm syslinux sysstat wget yum-utils
```

```
wget --no-cookies --no-check-certificate --header "Cookie: oraclelicense=accept-securebackup-cookie" https://download.oracle.com/otn-pub/java/jdk/11.0.1+13/90cf5d8f270a4347a95050320eef3fb7/jdk-11.0.1_linux-x64_bin.rpm
```

```
rpm -ivh jdk-8u181-linux-x64.rpm
```

```
env_override.sh
```

```
export JAVA_HOME=/usr/java/jdk1.8.0_171-amd64
```

```
export MAPR_HOME=/opt/mapr
```

```
export MAPR_USER=mapr
```

```
groupadd -g 5000 mapr
```

```
useradd -g 5000 -u 5000 mapr
```

```
useradd -m -u 5000 -g 5000 -G $(stat -c '%G' /etc/shadow) mapr
```

```
passwd mapr
```

```
gpasswd -a mapr mapr
```

```
| vi /etc/sudoers
```

```
mapr ALL=(ALL)    ALL
```

```
vi /etc/ssh/sshd_config
```

```
vi /etc/ssh/ssh_config
```

```
systemctl restart sshd
```

```
ssh-copy-id -i /root/.ssh/id_rsa.pub 10.142.0.3:/root/.ssh/authorized_keys
```

```
vi /etc/yum.repos.d/maprtech.repo
```

```
[maprtch] name=MapR  
Technologies  
baseurl=http://package.m  
apr.com/releases/v6.0.1/r  
edhat/ enabled=1  
gpgcheck=1
```

```
[maprecosystem] name=MapR  
Technologies  
baseurl=http://package.mapr.com/releases/MEP/MEP-5.0.0/redhat  
enabled=1 gpgcheck=1  
protect=1
```

```
rpm --import http://package.mapr.com/releases/pub/maprgpg.key
```

Master01

```
yum install -y mapr-fileserver mapr-nfs mapr-cldb mapr-resourcemanager mapr-webserver mapr-apiserver mapr-collectd mapr-nodemanager
```

s1

```
yum install -y mapr-fileserver mapr-nfs mapr-zookeeper mapr-nodemanager mapr-resourcemanager mapr-collectd
```

S2

```
yum install -y mapr-fileserver mapr-nfs mapr-zookeeper mapr-nodemanager mapr-collectd
```

s3

```
yum install -y mapr-fileserver mapr-nfs mapr-zookeeper mapr-nodemanager mapr-collectd
```

s4 and s5

```
yum install -y mapr-fileserver mapr-nfs mapr-nodemanager mapr-collectd mapr-grafana
```

```
yum install -y mapr-fileserver mapr-nfs mapr-nodemanager mapr-collectd mapr-opentsdb
```

From Master01

Run in master01:

```
/opt/mapr/server/configure.sh -secure -genkeys -Z s1.c.voltaic-racer-208109.internal:5181,s2.c.voltaic-racer-208109.internal:5181,s3.c.voltaic-racer-208109.internal:5181 -C master01.c.voltaic-racer-208109.internal:7222 -RM master01.c.voltaic-racer-208109.internal:8090,s1.c.voltaic-racer-208109.internal:8090 -N maprcluster -u mapr -g mapr
```

```
cd /opt/mapr/conf
```

```
scp -r cldb.key ssl_truststore ssl_keystore maprserverticket mapr@10.142.0.(9-14):/tmp/
```

All S(1-5) nodes

```
mv ssl_truststore ssl_keystore maprserverticket cldb.key /opt/mapr/conf/  
sudo chmod 600 ssl_truststore ssl_keystore maprserverticket cldb.key
```

(for Grafana 644 permission For a **secured cluster**, copy the `/opt/mapr/conf/ssl_truststore.pem` file from the CLDB master node to `/opt/mapr/conf` on all nodes(or, at a minimum, to the Grafana nodes) )

Run in all S(1-5) Nodes

```
/opt/mapr/server/configure.sh -secure -Z s1.c.voltaic-racer-208109.internal:5181,s2.c.voltaic-racer-208109.internal:5181,  
s3.c.voltaic-racer-208109.internal:5181 -C master01.c.voltaic-racer-208109.internal:7222 -N maprcluster -u mapr -g mapr
```

```
vi disks /dev/sdb
```

```
/dev/sdc
```

```
/dev/sdd
```

```
/opt/mapr/server/disksetup -FW 1 disks
```

First Start zookeeper Service

```
systemctl start mapr-zookeeper  
systemctl status mapr-zookeeper
```

2<sup>nd</sup> Warden Service need to be start

```
systemctl start mapr-warden  
systemctl status mapr-warden
```

```
systemctl restart mapr-zookeeper  
systemctl restart mapr-warden
```

## **mapr-posix-client-basic**

```
yum update -y  
yum install -y curl device-mapper iputils libsysfs lvm2 nc nfs-utils ntp nss openssl python-devel sdparm syslinux sysstat  
wget yum-utils
```

```
wget --no-cookies --no-check-certificate --header "Cookie: oraclelicense=accept-securebackup-cookie"  
http://download.oracle.com/otn-pub/java/jdk/8u181-b13/96a7b8442fe84_8ef90c96a2fad6ed6d1/jdk-8u181-linux-  
x64.rpm
```

```
rpm -ivh jdk-8u181-linux-x64.rpm
```

```
vi /etc/yum.repos.d/maprtech.repo
```

```
[maprtech] name=MapR  
Technologies
```

```
baseurl=http://package.mapr.com/releases/v6.0.1/redhat/  
enabled=1 gpgcheck=1
```

```
[maprecosystem] name=MapR  
Technologies  
baseurl=http://package.mapr.com/releases/MEP/MEP-5.0.0/redhat  
enabled=1 gpgcheck=1 protect=1
```

```
rpm --import http://package.mapr.com/releases/pub/maprgpg.key
```

**yum install mapr-posix-client-basic**

```
vi /etc/hosts
```

```
echo " 10.142.0.8 master01.c.voltaic-racer-208109.internal master01" >>  
/etc/hosts echo " 10.142.0.9 s1.c.voltaic-racer-208109.internal s1" >> /etc/hosts  
echo " 10.142.0.10 s2.c.voltaic-racer-208109.internal s2" >> /etc/hosts echo "  
10.142.0.11 s3.c.voltaic-racer-208109.internal s3" >> /etc/hosts echo "  
10.142.0.12 s4.c.voltaic-racer-208109.internal s4" >> /etc/hosts echo "  
10.142.0.13 s5.c.voltaic-racer-208109.internal s5" >> /etc/hosts
```

```
groupadd -g 5000 mapr  
useradd -g 5000 -u 5000  
mapr  
useradd -m -u 5000 -g 5000 -G $(stat -c '%G' /etc/shadow) mapr  
passwd mapr
```

```
vi /etc/ssh/sshd_config  
vi /etc/ssh/ssh_config
```

```
vi /opt/mapr/conf/fuse.conf
```

**#set path to mapr login ticket fuse.ticketfile.location=/opt/mapr/conf/longlived\_ticket**

#Set path to the mount point

`fuse.mount.point=/mapr`

### **Create Directory**

```
mkdir  
/mapr
```

### **Any cluster node run the command**

```
maprlogin generateticket -type service -out /tmp/longlived_ticket -user mapr
```

```
scp -p longlived_ticket mapr@10.142.0.14:/tmp/ (copy to posix client node)
```

### **From posix client node**

```
cd  
/tmp/  
ls  
mv longlived_ticket /opt/mapr/conf
```

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/mapr/lib  
/opt/mapr/server/configure.sh -secure -Z  
s1.c.voltaic-racer-208109.internal:5181,s2.c.voltaic-  
racer-208109.internal:5181,s3.c.voltaic-racer-  
208109.internal:5181 -C master01.c.voltaic-racer-  
208109.internal:7222 -N maprcluster -u mapr -g mapr -  
c
```

### **Run service**

```
service mapr-posix-client-basic status
```

```
service mapr-posix-client-basic start
```

service mapr-posix-client-basic status

Check the Directory and verify

cd /mapr

Latest change in Posix: fuse.conf

----

`fuse.num.libs = 1`

<code>fuse.cluster.conf.location</code>		The path to the configuration file to use.
	<code>/opt/mapr/conf/mapr-clusters.conf</code>	

Hive Configuration

Install

`yum -y install yum install mapr-hive mapr-hiveserver2 mapr-hivemetastore mapr-hivewebhcat`

Database

`yum -y install mariadb-server yum -y install mysql-connector-java`



## Edit the configuration file

```
cd /opt/mapr/hive/hive-2.1/conf
```

```
vi hive-site.xml
```

```
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:mysql://n3.us-east1-b.c.main-form-217005.internal:3306/metastore</value>
  <description>JDBC connect string for a JDBC metastore</description>
</property>

<property>
  <name>javax.jdo.option.ConnectionDriverName</name>
  <value>com.mysql.jdbc.Driver</value>
  <description>Driver class name for a JDBC metastore</description>
</property>

<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>hive</value>
  <description>username to use against metastore database</description>
</property>

<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>hivepwd</value>
  <description>password to use against metastore database</description>
</property>

<property>
  <name>hive.metastore.uris</name>
  <value>thrift://n3.us-east1-b.c.main-form-217005.internal:9083</value>
</property>
```

## Db service Start

```
Systemctl start mariadb
```

## Create Database,user and give Grant access

```
#mysql
mysql> create database metastore;
mysql> use metastore;
mysql> CREATE USER 'hive'@'localhost' IDENTIFIED BY
'hivepwd';
...
mysql> REVOKE ALL PRIVILEGES, GRANT OPTION FROM
'hive'@'localhost';
mysql> GRANT ALL PRIVILEGES ON metastore.* TO 'hive'@'localhost';
mysql> FLUSH PRIVILEGES;
mysql> quit;

and
mysql> CREATE USER 'hive'@'n2' IDENTIFIED BY 'hivepwd';
...
mysql> REVOKE ALL PRIVILEGES, GRANT OPTION FROM
'hive'@'n2';
mysql> GRANT ALL PRIVILEGES ON metastore.* TO 'hive'@'n2';
mysql> FLUSH PRIVILEGES;
mysql> quit;

and
mysql> CREATE USER 'hive'@'n3.us-east1-b.c.main-form-217005.internal' IDENTIFIED BY
'hivepwd';

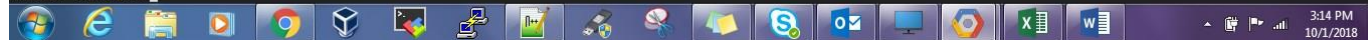
mysql> REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'hive'@'n3.us-east1-b.c.main-form-217005.internal';
mysql> GRANT ALL PRIVILEGES ON metastore.* TO 'hive'@'n3.us-east1-b.c.main-form-217005.internal';
mysql> FLUSH PRIVILEGES; mysql> quit;

SELECT host, user, password FROM mysql.user;
Drop databasemetastore;

Check for Login

#mysql -u hive -h s6.c.voltaic-racer-208109.internal -p
```

```
root@n3:~  
MariaDB [(none)]> SELECT host, user, password FROM mysql.user;  
+-----+-----+-----+  
| host | user | password |  
+-----+-----+-----+  
| localhost | root | |  
| n3 | root | |  
| 127.0.0.1 | root | |  
| ::1 | root | |  
| localhost | root | |  
| n3 | root | |  
| localhost | hive | *87404E49957F32FC20F6153952B919FFF3BAC878 |  
| n3 | hive | *87404E49957F32FC20F6153952B919FFF3BAC878 |  
| n3.us-east1-b.c.main-form-217005.internal | hive | *87404E49957F32FC20F6153952B919FFF3BAC878 |  
| n3.us-east1-b.c.main-form-217005.internal | hive | *87404E49957F32FC20F6153952B919FFF3BAC878 |  
+-----+-----+-----+  
10 rows in set (0.00 sec)  
  
MariaDB [(none)]> REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'hive'@'n3.us-east1-b.c.main-form-217005.internal';  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [(none)]> DROP USER 'hive'@'localhost';  
Query OK, 0 rows affected (0.01 sec)  
  
MariaDB [(none)]> DROP USER 'hive'@'n3';  
Query OK, 0 rows affected (0.01 sec)  
  
MariaDB [(none)]> DROP USER 'hive'@'n3.us-east1-b.c.main-form-217005.internal';  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [(none)]> DROP USER 'hive'@' n3.us-east1-b.c.main-form-217005.internal ';  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [(none)]> SELECT host, user, password FROM mysql.user;  
+-----+-----+-----+  
| host | user | password |  
+-----+-----+-----+  
| localhost | root | |  
| n3 | root | |  
| 127.0.0.1 | root | |  
| ::1 | root | |  
| localhost | root | |  
| n3 | root | |  
+-----+-----+-----+  
6 rows in set (0.00 sec)  
  
MariaDB [(none)]> |
```



```
mapr@n3:~  
[root@n3 conf]# mysql;  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 9  
Server version: 5.5.60-MariaDB MariaDB Server  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MariaDB [(none)]> use metastore;  
Database changed  
MariaDB [metastore]> CREATE USER 'hive'@'n3.us-east1-b.c.main-form-217005.internal' IDENTIFIED BY 'hivepwd';  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [metastore]> REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'hive'@'n3.us-east1-b.c.main-form-217005.internal';  
Query OK, 0 rows affected (0.01 sec)  
  
MariaDB [metastore]> GRANT ALL PRIVILEGES ON metastore.* TO 'hive'@'n3.us-east1-b.c.main-form-217005.internal';  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [metastore]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.00 sec)  
  
MariaDB [metastore]> quit  
Bye  
[root@n3 conf]# /opt/mapr/server/configure.sh -R  
Configuring Hadoop-2.7.0 at /opt/mapr/hadoop/hadoop-2.7.0  
Done configuring Hadoop  
Node setup configuration: collectd fileserver hive hivemetastore hiveserver2 hivewebhcat nfs nodemanager zookeep  
er  
Log can be found at: /opt/mapr/logs/configure.log  
Configuring collectd  
Configuring hive  
[root@n3 conf]# su - mapr  
Last login: Sun Sep 30 03:37:35 UTC 2018 on pts/0  
[mapr@n3 ~]$ maplogin password  
[Password for user 'mapr' at cluster 'maprcluster': ]  
MapR credentials of user 'mapr' for cluster 'maprcluster' are written to '/tmp/maprticket_5000'  
[mapr@n3 ~]$ /opt/mapr/hive/hive-2.1/bin/schematool -dbType mysql -initSchema  
Metastore connection URL: jdbc:mysql://n3.us-east1-b.c.main-form-217005.internal:3306/metastore  
Metastore Connection Driver : com.mysql.jdbc.Driver  
Metastore connection User: hive  
Starting metastore schema initialization to 2.1.0  
Initialization script hive-schema-2.1.0.mysql.sql  
Initialization script completed  
schemaTool completed  
[mapr@n3 ~]$
```

Configure

```
/opt/mapr/server/configure.sh -R
```

systemctl restart mapr-zookeeper

systemctl status mapr-zookeeper systemctl

restart mapr-warden

systemctl status mapr-warden

```
/opt/mapr/hive/hive-2.1/bin/schematool -dbType mysql -initSchema  
All HIVE service need to start
```

```
systemctl start mapr-hiveserver2 Systemctl
start mapr-hivemetastore
Systemctl start mapr-hivewebhcat

# hive
```

## **Disk adding and data encryption**

```
cryptsetup --batch-mode --use-random luksFormat /dev/sdg /etc/crypto/lukskey.bin
cryptsetup luksOpen /dev/sdg luks-sdg < /etc/crypto/lukskey.bin
--
/opt/mapr/server/mrconfig sp list -v
/opt/mapr/server/disksetup -F -W 4 /home/mapr/disklist
```

---

```
<property>
<name>hive.metastore.sasl.enabled</name>
<value>>false</value>
</property>
```

[https://mapr.com/docs/60/Spark/IntegrateSparkSQL\\_Hive.html](https://mapr.com/docs/60/Spark/IntegrateSparkSQL_Hive.html)

---

---

## Upgrade hive -2.1 to hive -2.3

```
[root@n2 tmp]# cat /etc/yum.repos.d/maprtech.repo
[maprtech] name=MapR
Technologies
baseurl=http://package.mapr.com/releases/v6.1.0/redhat/
enabled=1 gpgcheck=1
```

```
[maprecosystem] name=MapR Technologies
baseurl=http://package.mapr.com/releases/MEP/MEP-6.0.0/redhat
enabled=1 gpgcheck=1 protect=1
```

-----

**yum update mapr-hive\* -y**

-----

```
cp hive-site.xml ../hive-2.3/conf cp hive-
site.xml /opt/mapr/hive/hive2.3/conf/ pwd
cp hive-site.xml /opt/mapr/hive/hive-2.3/conf
pwd ls -lrt cd .. cd hive-2.3/ ls -lrt
cd /opt/mapr/hive/hive-2.1/lib
cd /opt/mapr/hive/hive-2.1/ ls
-lrt cd .. ls -lrt cd hive-
2.1.201804020853
ls -lrt
-----
cd /usr/share/java ls
-lrt
cp mysql-connector-java.jar /opt/mapr/hive/hive-2.3/lib

#/opt/mapr/server/configure.sh -R
----
su mapr
mysql
/opt/mapr/hive/hive-2.3/scripts/metastore/upgrade/mysql/
```

```
[root@n3 mysql]# mysql
```

```
MariaDB [(none)]> use metastore
```

```
Reading table information for completion of table and column names
```

```
You can turn off this feature to get a quicker startup with -A
```

```
MariaDB [metastore]> select * from VERSION;
```

VER_ID	SCHEMA_VERSION	VERSION_COMMENT
1	2.3.0	Hive release version 2.3.0

```
1 row in set (0.00 sec)
```

```
MariaDB [metastore]> source /opt/mapr/hive/hive-2.3/scripts/metastore/upgrade/mysql/upgrade-2.1.0-to-2.2.0.mysql.sql
```

```
MariaDB [metastore]> source /opt/mapr/hive/hive-2.3/scripts/metastore/upgrade/mysql/upgrade-2.2.0-to-2.3.0.mysql.sql
```

```
@@@@@@@@@@@@@@@@ Elastic Search Indexes Issue@@@@@@@@@@@@@@@@@@@@
```

```
[root@esekilxgp10 enaysib]# cd /opt/mapr/elasticsearch/elasticsearch-5.4.1/usr/share/curator/ [root@esekilxgp10 curator]#
```

```
[root@esekilxgp10 curator]# find . -type d -exec chmod +x {} \;
```

```
[root@esekilxgp10 curator]# ls -ltr total
```

```
2128
```

```
-rw-r--r-- 1 mapr maprg 1843136 Oct 28 2017 libpython2.7.so.1.0
```

```
-rw-r--r-- 1 mapr maprg 10488 Oct 28 2017 es_repo_mgr
```

```
-rw-r--r-- 1 mapr maprg 10488 Oct 28 2017 curator_cli
```

```
-rw-r--r-- 1 mapr maprg 10488 Oct 28 2017 curator -rw-r-
```

```
-r-- 1 mapr maprg 296399 Oct 28 2017 cacert.pem drwxr-xr-
```

```
x 3 mapr maprg 43 Jan 23 2018 lib drwxr-xr-x 3 mapr
```

```
maprg 43 Jan 23 2018 lib64 [root@esekilxgp10
```

```
curator]# chmod +x curator curator_cli
```

```
[root@esekilxgp10 curator]# cd /opt/mapr/elasticsearch/elasticsearch-5.4.1/bin
```

```

[root@esekilxgp10 bin]# ln -s /opt/mapr/elasticsearch/elasticsearch-5.4.1/usr/share/curator/curator curator
[root@esekilxgp10 bin]# vi /opt/mapr/elasticsearch/elasticsearch-5.4.1/etc/elasticsearch/curator.yml
[root@esekilxgp10 bin]# cat /opt/mapr/elasticsearch/elasticsearch-5.4.1/etc/elasticsearch/curator.yml ---
# Remember, leave a key empty if there is no value.  None will be a string,
# not a Python "NoneType"
client:  hosts:
    - esekilxgp10.rnd.ki.sw.ericsson.se
port: 9200  url_prefix:  use_ssl:
True  certificate:  client_cert:
client_key:  aws_key:
    aws_secret_key:
aws_region:
    ssl_no_validate: True
http_auth: admin:admin
timeout: 30  master_only:
False
logging:
    loglevel: INFO
logfile:
    logformat: default
    blacklist: ['elasticsearch', 'urllib3'] [root@esekilxgp10
bin]# crontab -u mapr -l
SHELL=/bin/bash
15 3 * * * /opt/mapr/elasticsearch/elasticsearch-5.4.1/bin/curator --config /opt/mapr/elasticsearch/elasticsearch-5.4.1
/opt/mapr/elasticsearch/elasticsearch-5.4.1/etc/elasticsearch/curator_actions/delete_indices.yml >>
/opt/mapr/elasticsearch/e
[root@esekilxgp10 bin]# su mapr
[mapr@esekilxgp10 bin]$ maprlogin password
[Password for user 'mapr' at cluster 'rdidev1': ]
MapR credentials of user 'mapr' for cluster 'rdidev1' are written to '/tmp/maprticket_123400016'
[mapr@esekilxgp10 bin]$ /opt/mapr/elasticsearch/elasticsearch-5.4.1/bin/curator --config
/opt/mapr/elasticsearch/elasticsearc
/opt/mapr/elasticsearch/elasticsearch-5.4.1/etc/elasticsearch/curator_actions/delete_indices.yml >> /opt/mapr/elasticsearch/
[mapr@esekilxgp10 bin]$ exit
[root@esekilxgp10 bin]# df -H /opt/
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/rootvg-rootvol 500G    193G   307G   39% /
[root@esekilxgp10 bin]#

```



-----R and sparkR and Spark and hive -----

```
[root@esekilxgp05 ~]# yum install R
```

```
[root@esekilxgp05 ~]# which R
```

```
Set $HOME in /usr/bin and its show in env
```

```
[root@esekilxgp05 ~]# cat .bash_profile
```

```
# .bash_profile
```

```
# Get the aliases and functions
```

```
if [ -f ~/.bashrc ]; then
```

```
. ~/.bashrc fi
```

```
# User specific environment and startup programs
```

```
PATH=$PATH:$HOME/bin
```

```
export PATH
```

```
[root@esekilxgp05 ~]# env
```

```
PATH=/usr/lib64/qt-
```

```
3.3/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin:/usr/java/jdk1.8.0_161//bin:/usr/java/
```

For sparkR and Spark and hive-----

```
[root@esekilxgp05 ~]# yum install -y mapr-spark mapr-hive
```

```
[root@esekilxgp05 ~]# cd /opt/mapr/spark/spark-2.1.0/conf/
```

```
[root@esekilxgp05 ~]#ls -ltar
```

```
-rwxr-xr-x  1 mapr maprg 7191 Oct 11 07:51 spark-env.sh
```

```
-rw-r--r--  1 mapr maprg 3727 Oct 11 07:51 hive-site.xml
```

```
-rw-r--r--  1 mapr maprg 1749 Oct 11 07:52 spark-defaults.conf
```

```
[root@esekilxgp05 ~]# vi spark-defaults.conf
```

Add below files:

```
spark.yarn.dist.files = /opt/mapr/spark/spark-2.3.1/conf/hive-site.xml
spark.sql.hive.metastore.version = 1.2.0
```

```
[root@esekilxgp05 conf]# cat hive-site.xml
```

```
<?xml version="1.0"?>
```

```
<!--
```

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the

"License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
-->
```

```
<configuration>
```

```
    <property>
```

```
        <name>javax.jdo.option.ConnectionURL</name>
```

```
        <value>jdbc:mysql://esekilxgp06.rnd.ki.sw.ericsson.se:3306/hive?createDatabaseIfNotExist=true</value>
```

```
<description>JDBC connect string for a JDBC metastore</description>
```

```
    </property>
```

```
    <property>
```

```
        <name>javax.jdo.option.ConnectionDriverName</name>
```

```
        <value>com.mysql.jdbc.Driver</value>
```

```
        <description>Driver class name for a JDBC metastore</description>
```

```
</property>
```

```
<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>hive</value>
  <description>username to use against metastore database</description>
</property>

<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>CyT4cPPfwDs</value>
  <description>password to use against metastore database</description>

</property>
  <property>
    <name>hive.metastore.schema.validation</name>
    <value>false</value>
  </property>

<property>
  <name>hive.metastore.uris</name>
  <value>thrift://esekilx5636.rnd.ki.sw.ericsson.se:9083,thrift://esekilx5637.rnd.ki.sw.ericsson.se:9083</value>
</property>
<!-- For hive server2 -->
<property>
  <name>hive.server2.enable.doAs</name>
  <value>true</value>
</property>
<!-- For hive server2 and meta store -->
<property>
  <name>hive.metastore.execute.setugi</name>
  <value>true</value>
</property>

<property>
  <name>hive.metastore.warehouse.dir</name>
  <value>/project/rdi/warehouse/hive</value>
  <description>location of default database for the warehouse</description>
</property>

<property>
  <name>hive.metastore.try.direct.sql</name>
  <value>true</value>
  <description>
```

Whether the Hive metastore should try to use direct SQL queries instead of the DataNucleus for certain read paths. This can improve metastore performance when fetching many partitions or column statistics by orders of magnitude; however, it is not guaranteed to work on all RDBMS-es and all versions. In case of SQL failures, the metastore will fall back to the DataNucleus, so it's safe even if SQL doesn't work for all queries on your datastore. If all SQL queries fail (for example, your metastore is backed by MongoDB), you might want to disable this to save the try-and-fall-back cost.

```
</description>
</property>
<property>
  <name>hive.metastore.client.socket.timeout</name>
  <value>1800s</value>
  <description>
    Expects a time value with unit (d/day, h/hour, m/min, s/sec, ms/msec, us/usec, ns/nsec), which is sec if not
    specified.
    MetaStore Client socket timeout in seconds
  </description>
</property>

<property>
<name>hive.metastore.sasl.enabled </name>
<value>false</value>
</property>

</configuration>
```

```
[root@esekilxgp05 conf]# cat spark-env.sh
```

```
#!/usr/bin/env bash
```

```
#
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with #
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at #
# http://www.apache.org/licenses/LICENSE-2.0
#
```

```

# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and #
# limitations under the License.
#

# This file is sourced when running various Spark programs.
# Copy it as spark-env.sh and edit that to configure Spark for your site.

# Options read when launching programs locally with
# ./bin/run-example or ./bin/spark-submit
# - HADOOP_CONF_DIR, to point Spark towards Hadoop configuration files
# - SPARK_LOCAL_IP, to set the IP address Spark binds to on this node
# - SPARK_PUBLIC_DNS, to set the public dns name of the driver program #
# - SPARK_CLASSPATH, default classpath entries to append

# Options read by executors and drivers running inside the cluster
# - SPARK_LOCAL_IP, to set the IP address Spark binds to on this node
# - SPARK_PUBLIC_DNS, to set the public DNS name of the driver program
# - SPARK_CLASSPATH, default classpath entries to append
# - SPARK_LOCAL_DIRS, storage directories to use on this node for shuffle and RDD data
# - MESOS_NATIVE_JAVA_LIBRARY, to point to your libmesos.so if you use Mesos
# Options read in YARN client mode
# - HADOOP_CONF_DIR, to point Spark towards Hadoop configuration files
# - SPARK_EXECUTOR_INSTANCES, Number of executors to start (Default: 2) #
# - SPARK_EXECUTOR_CORES, Number of cores for the executors (Default: 1).
# - SPARK_EXECUTOR_MEMORY, Memory per Executor (e.g. 1000M, 2G) (Default: 1G)
# - SPARK_DRIVER_MEMORY, Memory for Driver (e.g. 1000M, 2G) (Default: 1G)
# Options for the daemons used in the standalone deploy mode
# - SPARK_MASTER_HOST, to bind the master to a different IP address or hostname
# - SPARK_MASTER_PORT / SPARK_MASTER_WEBUI_PORT, to use non-default ports for the master
# - SPARK_MASTER_OPTS, to set config properties only for the master (e.g. "-Dx=y")
# - SPARK_WORKER_CORES, to set the number of cores to use on this machine
# - SPARK_WORKER_MEMORY, to set how much total memory workers have to give executors (e.g. 1000m, 2g)
# - SPARK_WORKER_PORT / SPARK_WORKER_WEBUI_PORT, to use non-default ports for the worker
# - SPARK_WORKER_INSTANCES, to set the number of worker processes per node
# - SPARK_WORKER_DIR, to set the working directory of worker processes
# - SPARK_WORKER_OPTS, to set config properties only for the worker (e.g. "-Dx=y")
# - SPARK_DAEMON_MEMORY, to allocate to the master, worker and history server themselves (default: 1g). #
# - SPARK_HISTORY_OPTS, to set config properties only for the history server (e.g. "-Dx=y")
# - SPARK_SHUFFLE_OPTS, to set config properties only for the external shuffle service (e.g. "-Dx=y")
# - SPARK_DAEMON_JAVA_OPTS, to set config properties for all daemons (e.g. "-Dx=y")

```

```

# - SPARK_PUBLIC_DNS, to set the public dns name of the master or workers
# Generic options for the daemons used in the standalone deploy mode
# - SPARK_CONF_DIR      Alternate conf dir. (Default: ${SPARK_HOME}/conf)
# - SPARK_LOG_DIR       Where log files are stored. (Default: ${SPARK_HOME}/logs)
# - SPARK_PID_DIR       Where the pid file is stored. (Default: /tmp)
# - SPARK_IDENT_STRING  A string representing this instance of spark. (Default: $USER)
# - SPARK_NICENESS       The scheduling priority for daemons. (Default: 0)
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground. It will not output a PID file.

##### #
Set MapR attributes and compute classpath
#####

# Set the spark attributes
if [ -d "/opt/mapr/spark/spark-2.1.0" ]; then
export SPARK_HOME=/opt/mapr/spark/spark-2.1.0 fi

# Load the hadoop version attributes
source /opt/mapr/spark/spark-2.1.0/mapr-util/hadoop-version-picker.sh
export HADOOP_HOME=$hadoop_home_dir export
HADOOP_CONF_DIR=$hadoop_conf_dir

# Enable mapr impersonation export
MAPR_IMPERSONATION_ENABLED=1

MAPR_HADOOP_CLASSPATH=`/opt/mapr/spark/spark-2.1.0/bin/mapr-classpath.sh`
MAPR_HADOOP_JNI_PATH=`hadoop jnipath`
MAPR_SPARK_CLASSPATH="$MAPR_HADOOP_CLASSPATH"

SPARK_MAPR_HOME=/opt/mapr

export SPARK_LIBRARY_PATH=$MAPR_HADOOP_JNI_PATH
export LD_LIBRARY_PATH="$MAPR_HADOOP_JNI_PATH:$LD_LIBRARY_PATH"

# Load the classpath generator script
source /opt/mapr/spark/spark-2.1.0/mapr-util/generate-classpath.sh

# Calculate hive jars to include in classpath generate_compatible_classpath
"spark" "2.1.0" "hive"
MAPR_HIVE_CLASSPATH=${generated_classpath} if
[ ! -z "$MAPR_HIVE_CLASSPATH" ]; then
    MAPR_SPARK_CLASSPATH="$MAPR_SPARK_CLASSPATH:$MAPR_HIVE_CLASSPATH"
fi

```

```

# Calculate hbase jars to include in classpath generate_compatible_classpath
"spark" "2.1.0" "hbase"
MAPR_HBASE_CLASSPATH=${generated_classpath} if
[ ! -z "$MAPR_HBASE_CLASSPATH" ]; then
    MAPR_SPARK_CLASSPATH="$MAPR_SPARK_CLASSPATH:$MAPR_HBASE_CLASSPATH"
    SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dspark.driver.extraClassPath=$MAPR_HBASE_CLASSPATH"
fi
# Set executor classpath for MESOS. Uncomment following string if you want deploy spark jobs on Mesos
#MAPR_MESOS_CLASSPATH=$MAPR_SPARK_CLASSPATH
SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dspark.executor.extraClassPath=$MAPR_HBASE_CLASSPATH:$MAPR_MESOS_CLASSPATH"

# Set SPARK_DIST_CLASSPATH
export SPARK_DIST_CLASSPATH=$MAPR_SPARK_CLASSPATH

# Security status source
/opt/mapr/conf/env.sh
if [ "$MAPR_SECURITY_STATUS" = "true" ]; then
    SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dhadoop.login=hybrid -Dmapr_sec_enabled=true" fi
#
scala
export SCALA_VERSION=2.11
export SPARK_SCALA_VERSION=$SCALA_VERSION export
SCALA_HOME=/opt/mapr/spark/spark-2.1.0/scala export
SCALA_LIBRARY_PATH=$SCALA_HOME/lib

# Use a fixed identifier for pid files
export SPARK_IDENT_STRING="mapr"
##### #
:::CAUTION::: DO NOT EDIT ANYTHING ON OR ABOVE THIS LINE
#####

#
# MASTER HA SETTINGS
#
#export SPARK_DAEMON_JAVA_OPTS="-Dspark.deploy.recoveryMode=ZOOKEEPER -Dspark.deploy.zookeeper.url=<zookeeper1:5181,zookeeper2:5181>
Djava.security.auth.login.config=/opt/mapr/conf/mapr.login.conf -Dzookeeper.sasl.client=false"

# MEMORY SETTINGS
export SPARK_DAEMON_MEMORY=1g
export SPARK_WORKER_MEMORY=16g
# Worker Directory

```

```
export SPARK_WORKER_DIR=$SPARK_HOME/tmp
```

```
# Environment variable for printing spark command everytime you run spark.Set to "1" to print. #
export SPARK_PRINT_LAUNCH_COMMAND=1
```

```
/opt/mapr/spark/spark-2.2.1/bin
```

[illegible]

Below are the instruction on how to install library for sparklyr incase asked in future.

- Go inside R as root.
- `install.packages("sparklyr")`
- choose 33 (for Sweden)
- it should work, incase dependencies not getting installed. We may need to do `"yum install <>"` for curl, xml, openssl, http packages.

For **"sparklyr.nested"** library

- Go inside R as root.
- `install.packages("devtools")`
- `devtools::install_github("mitre/sparklyr.nested")`
- `library(sparklyr.nested)`

[illegible]



Whenever we will get the any impala connection issue please execute the below command from **esekilxgp01**.

```
clush -b -g prod "impala-shell -i localhost:21000 -q 'select 1 from bpn.summary'"
```

If you get "**clush: esekilx[5646-5647] (2): exited with exit code: 1**", it means on these two node there is no running impalad daemon.

If you get "**clush: esekilx[5639-5640] (2): exited with exit code: 255**", it means on these two node there are running impalad daemon, but it can't fetch data from the impala. So in that case you need to debug the impalad daemon issue on those nodes.

To test JDBC port, i.e port no: 21050 (HAPROXY port: 21051) you need to have any jdbc-database viewer like **DBeaver**.

#### Imapal installation steps:

1. yum -y yum install mapr-impala mapr-impala-statestore mapr-impala-catalog mapr-impala-server
2. cd /opt/mapr/impala/impala-2.10.0/conf
3. cp /opt/mapr/hive/hive-2.3/conf/hive-site.xml .
4. chown mapr:mapr \*
5. vi env.sh

```
# This MUST point to the node running statestore
IMPALA_STATE_STORE_HOST=z2.c.voltaic-racer-208109.internal
IMPALA_STATE_STORE_PORT=24000
CATALOG_SERVICE_HOST=z2.c.voltaic-racer-208109.internal

#Set the Shared Memory to 128 MB
export MAPR_CLIENT_SHMEM=16384

# These impact the impala server and can be optionally changed
IMPALA_BACKEND_PORT=22000
IMPALA_LOG_DIR=${IMPALA_HOME}/logs
IMPALA_SERVER_ARGS=" \
    -log_dir=${IMPALA_LOG_DIR} \
    -state_store_port=${IMPALA_STATE_STORE_PORT} \
    -use_statestore \
    -authorized_proxy_user_config=mapr=* \
    -state_store_host=${IMPALA_STATE_STORE_HOST} \
    -catalog_service_host=${CATALOG_SERVICE_HOST} \
    -be_port=${IMPALA_BACKEND_PORT} \
    -mem_limit=20% \
"
```

```
# /opt/mapr/server/configure.sh -R
```

```

[root@z2 conf]# /opt/mapr/server/configure.sh -R
Configuring Hadoop-2.7.0 at /opt/mapr/hadoop/hadoop-2.7.0
Done configuring Hadoop
Node setup configuration:  fileserver hbase hbinternal hive impalacatalog impalaserver impalastore nodemanager zookeeper
Log can be found at:  /opt/mapr/logs/configure.log
Configuring hive
Configuring hbase
[root@z2 conf]# su - mapr
Last login: Sat Dec 22 02:35:16 UTC 2018
[mapr@z2 ~]$ maprlogin password
[Password for user 'mapr' at cluster 'maprcluster': ]
MapR credentials of user 'mapr' for cluster 'maprcluster' are written to '/home/mapr/impersonation_ticket'
[mapr@z2 ~]$ impala-shell -i z2.c.voltaic-racer-208109.internal:21000
Starting Impala Shell without Kerberos authentication
Connected to z2.c.voltaic-racer-208109.internal:21000
Server version: impalad version 2.10.0 RELEASE (build 289c879964b9c57f0ca326992f078631bb59a29d)
*****
Welcome to the Impala shell.
(Impala Shell v2.10.0 (289c879) built on Tue Sep 25 21:08:08 UTC 2018)

When you set a query option it lasts for the duration of the Impala shell session.
*****
[z2.c.voltaic-racer-208109.internal:21000] > show databases;
Query: show databases
+-----+-----+
| name          | comment                                     |
+-----+-----+
| _impala_builtins | System database for Impala builtin functions |
| default        | Default Hive database                     |
| siba           |                                           |
+-----+-----+
Fetched 3 row(s) in 0.33s
[z2.c.voltaic-racer-208109.internal:21000] > █

```

## MapR-DB

```
[mapr@nm ~]$ hadoop fs -mkdir /data
[mapr@nm ~]$ hadoop fs -ls /
Found 6 items
drwxr-xr-x  - mapr mapr      0 2018-09-26 15:22 /apps
drwxr-xr-x  - mapr mapr      0 2018-10-20 03:29 /data
drwxr-xr-x  - mapr mapr      0 2018-09-26 15:25 /opt
drwxrwxrwx  - mapr mapr      0 2018-10-20 03:17 /tmp
drwxr-xr-x  - mapr mapr      2 2018-09-30 03:31 /user
drwxr-xr-x  - mapr mapr      1 2018-09-26 15:23 /var
[mapr@nm ~]$ mapr dbshell
Warning: Unable to determine $DRILL_HOME
Warning: Unable to determine $DRILL_HOME
=====
*           MapR-DB Shell           *
* NOTE: This is a shell for JSON table operations. *
=====
Version: 6.0.1-mapr

MapR-DB Shell
maprdb mapr:> create /data/users
Table /data/users created.
maprdb mapr:> insert /data/users --id "1" --value '{"name":"siba","phone":"143"}'
Document with id: "1" inserted.
maprdb mapr:> find /data/users
{"_id":"1","name":"siba","phone":"143"}
1 document(s) found.
maprdb mapr:> findbyid /data/users --id "1"
{"_id":"1","name":"siba","phone":"143"}
1 document(s) found.
maprdb mapr:> █
```

Snapshot:

```
df -T
hadoop dfs -createSnapshot /user/cloudera snap1
hadoop dfs -deleteSnapshot /user/cloudera snap1
maprccli volume snapshot create -snapshotname snap1 -volume bruce
hadoop fs -ls /user/bruce/.snapshot/snap1
```

```
for i in {3..6}; do ssh 10.142.0.$i "hostname; echo -e '\n'";scp -r cldb.key ssl_keystore ssl_truststore maprserverticket
mapr@10.142.0.4:/tmp;/done
```

```
for i in {8..14}; do ssh 10.142.0.$i "hostname; echo -e '\n'";scp -r cldb.key ssl_keystore ssl_truststore maprserverticket
mapr@10.142.0.$i:/tmp;/done
```

```
scp -r cldb.key ssl_keystore ssl_truststore maprserverticket mapr@10.142.0.4:/tmp/
```

```
watch -n 1 ls -ltr
```

```
for i in {2..6}; do ssh mapr@10.142.0.$i "hostname;jps ;echo -e '\n'";done >> test.out
```

```
for i in {8..14}; do ssh mapr@10.142.0.$i "hostname;sudo systemctl start mapr-zookeeper;sudo systemctl status mapr-
zookeeper;sudo systemctl start mapr-warden;sudo systemctl status mapr-warden;echo -e '\n'";done >> service.out
```

```
rpm -qa | grep mapr | for i in `awk {'print$1'}`; do yum erase $i; done
rpm -qa | grep httpd | for i in `awk {'print$1'}`; do yum -y erase $i; done > test.txt
```

## Vagrant

```
yum install gcc make kernel-devel -y
yum groupinstall " X Window System"
wget https://download.virtualbox.org/virtualbox/5.2.18/VirtualBox-5.2-5.2.18_124319_el7-1.x86_64.rpm
```

```
wget https://releases.hashicorp.com/vagrant/2.1.4/vagrant_2.1.4_x86_64.rpm
```

```
yum localinstall VirtualBox-5.2-5.2.18_124319_el7-1.x86_64.rpm
```

```
yum localinstall vagrant_2.1.4_x86_64.rpm
```

```
/sbin/vboxconfig
```

```
vagrant --version
```

```
mkdir -p /siba/centos
```

```
cd /siba/centos
```

```
vagrant box add centos/7
```

```
vagrant init centos/7
```

```
vim Vagrantfile
```

```
vagrant init centos/7
```

```
vim Vagrantfile
```

```
vagrant box add centos/7
```

```
vim Vagrantfile
```

```
vagrant box add centos/7
```

```
vagrant ssh
```

```
vagrant up
```

```
vim Vagrantfile
```

```
vagrant ssh-config
```

```
vagrant box list
```

```
vboxmanage stproperty machinefolder /siba/centos
```

```
vboxmanage setproperty machinefolder /siba/centos
```

```
vagrant up
```

```
@@@@@@@
```

```
#!/bin/bash
```

```
for i in `cat prod-list`
```

```
do echo "-----$i-----"
```

```
ssh $i cp /etc/resolv.conf /etc/resolv.conf.org
```

```
cat /root/resolv.conf | ssh root@$i 'cat > /etc/resolv.conf'
```

```
done
```

```
@@@@@@@@@@@@@@@@
```

```
/opt/mapr/spark/spark-2.1.0/bin/run-example --master yarn --deploy-mode client SparkPi 10
```

```
./run-example --master yarn sql.hive.SparkHiveExample
```

```
cp -p log4j.properties.template log4j.properties
```

```
bin/spark-submit --class org.apache.spark.examples.SparkPi --master yarn ./examples/jars/spark-examples_2.11-2.1.0-mapr-1710.jar 10
```

```
./spark-shell --master="yarn"
```

```
Imp:
```

```
./spark-submit --class org.apache.spark.examples.SparkPi --master yarn ./examples/jars/spark-examples_2.11-2.1.0-mapr-1710.jar 10
```

```
#####$$$$
```

```
./run-example --master yarn SparkPi 20
```

```
@@@@@@@@@@@@@@@@ HAPROXY#####
```

```
listen mariadb_cluster 0.0.0.0:3030
```

```
## MariaDB balance leastconn - the cluster listening on port 3030.
```

```
    mode tcp
```

```
    balance leastconn
```

```
    option httpchk
```

```
    server db1 192.168.1.11:3306 check port 9200
```

```
    server db2 192.168.1.12:3306 check port 9200
```

```
listen stats 0.0.0.0:9000
```

```
## HAProxy stats web gui running on port 9000 - username and password: test
```

```
    mode http
```

```
    stats enable
```

```
    stats uri /stats
```

```
    stats realm HAProxy\ Statistics
```

```
    stats auth test:test
```

```
    stats admin if TRUE
```

```
#####@2222#####
```

```
cat haproxy.cfg
```

```
#-----
```

```
# Example configuration for a possible web application.  See the
# full configuration options online.
#
#   http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
#
#-----

#-----
# 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            4000
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
    log                 global
    option              httplog
    option              dontlognull
    option http-server-close
    option forwardfor    except 127.0.0.0/8
    option              redispatch
    retries              6
    timeout http-request 1m
    timeout queue        1m
    timeout connect      1m
    timeout client       1m
    timeout server       1m
    timeout http-keep-alive 1m
    timeout check        10s
    maxconn              6000

#-----
# main frontend which proxys to the backends
#-----
frontend main *:5000
    acl url_static      path_beg       -i /static /images /javascript /stylesheets
    acl url_static      path_end       -i .jpg .gif .png .css .js

    use_backend static   if url_static
    default_backend      app

#-----
# static backend for serving up images, stylesheets and such
#-----
backend static
    balance roundrobin
    server static 127.0.0.1:4331 check

#-----
# round robin balancing between the various backends
#-----
backend app

```



```
balance      roundrobin
server app1 127.0.0.1:5001 check
server app2 127.0.0.1:5002 check
server app3 127.0.0.1:5003 check
server app4 127.0.0.1:5004 check
```

```
# -----
# This is the setup for Impala. Impala client connect to load_balancer_host:2500
3.
# HAProxy will balance connections among the list of servers listed below.
# The list of Impalad is listening at port 21000 for beeswax (impala-shell) or o
riginal ODBC driver.
# For JDBC or ODBC version 2.x driver, use port 21050 instead of 21000.
# -----
listen impala :25001
    mode tcp
    option tcplog
    balance leastconn

server impala_ha_odbc_1 134.138.210.132:21000
server impala_ha_odbc_2 134.138.210.133:21000
server impala_ha_odbc_3 134.138.210.134:21000
server impala_ha_odbc_4 134.138.210.135:21000
server impala_ha_odbc_5 134.138.210.136:21000
server impala_ha_odbc_6 134.138.210.137:21000
server impala_ha_odbc_7 134.138.210.138:21000
server impala_ha_odbc_8 134.138.210.139:21000
server impala_ha_odbc_9 134.138.210.140:21000
server impala_ha_odbc_10 134.138.210.141:21000
server impala_ha_odbc_11 134.138.210.142:21000
server impala_ha_odbc_12 134.138.210.143:21000
# server impala_ha_odbc_13 134.138.210.144:21000
server impala_ha_odbc_14 134.138.210.146:21000
server impala_ha_odbc_15 134.138.210.147:21000
server impala_ha_odbc_16 134.138.210.148:21000
server impala_ha_odbc_17 134.138.210.149:21000
server impala_ha_odbc_18 134.138.210.153:21000
server impala_ha_odbc_19 134.138.210.154:21000
server impala_ha_odbc_20 134.138.210.152:21000
```

```

server impala_ha_odbc_21 134.138.210.155:21000
server impala_ha_odbc_22 134.138.210.156:21000
server impala_ha_odbc_23 134.138.210.157:21000
server impala_ha_odbc_24 134.138.210.158:21000
server impala_ha_odbc_25 134.138.210.151:21000
#server impala_ha_odbc_14 134.138.210.145:21000 (server offline atm)

# -----
# Setup for Hue or other JDBC-enabled applications.
# In particular, Hue requires sticky sessions.
# The application connects to load_balancer_host:21051, and HAProxy balances
# connections to the associated hosts, where Impala listens for JDBC
# requests on port 21050.
# -----
listen impalajdbc :21051
    mode tcp
    option tcplog
    balance source
    server impala_ha_jdbc_1 134.138.210.132:21050
    server impala_ha_jdbc_2 134.138.210.133:21050
    server impala_ha_jdbc_3 134.138.210.134:21050
    server impala_ha_jdbc_4 134.138.210.135:21050
    server impala_ha_jdbc_5 134.138.210.136:21050
    server impala_ha_jdbc_6 134.138.210.137:21050
    server impala_ha_jdbc_7 134.138.210.138:21050
    server impala_ha_jdbc_8 134.138.210.139:21050
    server impala_ha_jdbc_9 134.138.210.140:21050
    server impala_ha_jdbc_10 134.138.210.141:21050
    server impala_ha_jdbc_11 134.138.210.142:21050
    server impala_ha_jdbc_12 134.138.210.143:21050
#    server impala_ha_jdbc_13 134.138.210.144:21050
    server impala_ha_jdbc_14 134.138.210.146:21050
    server impala_ha_jdbc_15 134.138.210.147:21050
    server impala_ha_jdbc_16 134.138.210.148:21050
    server impala_ha_jdbc_17 134.138.210.149:21050
    server impala_ha_jdbc_18 134.138.210.153:21050
    server impala_ha_jdbc_19 134.138.210.154:21050
    server impala_ha_jdbc_20 134.138.210.152:21050
    server impala_ha_jdbc_21 134.138.210.155:21050
    server impala_ha_jdbc_22 134.138.210.156:21050

```

```
server impala_ha_jdbc_23 134.138.210.157:21050
server impala_ha_jdbc_24 134.138.210.158:21050
server impala_ha_jdbc_25 134.138.210.151:21050
#server impala_ha_jdbc_14 134.138.210.145:21050 (server offline atm)
```

```
sed -i -e 's/SELINUX=permissive/SELINUX=disabled/g' /etc/sysconfig/selinux
```

```
firewall-cmd --permanent --add-port=3306/tcp
```

```
firewall-cmd -reload
```

```
##### LUKENCRYPT#####
```

```
#!/bin/bash
```

```
#yum -y install cryptsetup
```

```
#yum -y update device-mapper
```

```
# timestamp="$(date +%Y-%m-%d.%H:%M:%S)"
```

```
#mv -v /etc/crypto /etc/crypto."$timestamp"
```

```
mkdir -p /etc/crypto
```

```
chmod -R go-rw /etc/crypto
```

```
#mv -v /etc/crypttab /etc/crypttab."$timestamp"
```

```
#mv -v /opt/mapr/disks.txt /opt/mapr/disks.txt."$timestamp"
```

```
tr -dc '[:graph:]' < /dev/random | head -c "${1:-512}" > /etc/crypto/lukskey.bin
```

```
chmod go-rw /etc/crypto/lukskey.bin
```

```
disks="sda sdb sdc sdd sde sdf sdg sdh sdi sdj sdk sdl sdm sdn sdo sdp"
```

```
for f in $disks
```

```
do
```

```
# rm -fv /etc/crypto/"$f"-key.bin
```

```
# cryptsetup close luks-"$f"
```

```
cryptsetup --batch-mode --use-random luksFormat /dev/"$f" /etc/crypto/lukskey.bin
```

```
cryptsetup luksOpen /dev/"$f" luks-"$f" < /etc/crypto/lukskey.bin
```

```
echo luks-"$f" /dev/"$f" /etc/crypto/lukskey.bin >> /etc/crypttab
```

```
echo /dev/mapper/luks-"$f" >> /root/setup_files/disks.txt
```

```
done
```

```
#echo "Backup files created..."
#ls -l {/etc/crypto."$timestamp",/etc/crypttab."$timestamp",/opt/mapr/disks.txt."$timestamp"}
```

## Troubleshooting Livy

```
curl -X POST --data '{"proxyUser":"ehasbj", "kind": "pyspark"}' -H "Content-Type: application/json"
esekilxgp02.rnd.ki.sw.ericsson.se:8998/sessions
curl esekilxgp07.rnd.ki.sw.ericsson.se:8998/sessions/ | python -m json.tool
```

## Run Yarn Job

```
/opt/mapr/spark/spark-2.3.1/bin/spark-submit --class org.apache.spark.examples.SparkPi --master yarn --
deploy-mode cluster /opt/mapr/spark/spark-2.3.1/examples/jars/spark-examples_2.11-2.3.1-mapr-1808.jar 2
```

```
[mapr@n5 hadoop-2.7.0]$ find . -name "*emaples*"
[mapr@n5 hadoop-2.7.0]$ find . -name "*examples*"
./share/hadoop/mapreduce/lib-examples
./share/hadoop/mapreduce/sources/hadoop-mapreduce-examples-2.7.0-mapr-1808-sources.jar
./share/hadoop/mapreduce/sources/hadoop-mapreduce-examples-2.7.0-mapr-1808-test-sources.jar
./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar
[mapr@n5 hadoop-2.7.0]$ yarn jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar
An example program must be given as the first argument.
Valid program names are:
    aggregatewordcount: An Aggregate based map/reduce program that counts the words in the input files.
    aggregatewordhist: An Aggregate based map/reduce program that computes the histogram of the words in
the input files.
    bbp: A map/reduce program that uses Bailey-Borwein-Plouffe to compute exact digits of Pi.
```

blocklocality: Checking Map job locality  
dbcount: An example job that count the pageview counts from a database.  
distbbp: A map/reduce program that uses a BBP-type formula to compute exact bits of Pi.  
grep: A map/reduce program that counts the matches of a regex in the input.  
join: A job that effects a join over sorted, equally partitioned datasets  
multifilewc: A job that counts words from several files.  
pentomino: A map/reduce tile laying program to find solutions to pentomino problems.  
pi: A map/reduce program that estimates Pi using a quasi-Monte Carlo method.  
randomtextwriter: A map/reduce program that writes 10GB of random textual data per node.  
randomwriter: A map/reduce program that writes 10GB of random data per node.  
secondariesort: An example defining a secondary sort to the reduce.  
sleep: A job that sleeps at each map and reduce task.  
sort: A map/reduce program that sorts the data written by the random writer.  
sudoku: A sudoku solver.  
terachecksum: Compute checksum of terasort output to compare with teragen checksum.  
teragen: Generate data for the terasort  
teragenwithcrc: Generate data for the terasort with CRC checksum  
terasort: Run the terasort  
terasortwithcrc: Run the terasort with CRC checksum  
teravalidate: Checking results of terasort  
teravalidaterecords: Checking results of terasort in terms of missing/duplicate records  
teravalidatewithcrc: Checking results of terasort along with crc verification  
wordcount: A map/reduce program that counts the words in the input files.  
wordmean: A map/reduce program that counts the average length of the words in the input files.  
wordmedian: A map/reduce program that counts the median length of the words in the input files.  
wordstandarddeviation: A map/reduce program that counts the standard deviation of the length of the words in the input files.

```
[mapr@n5 hadoop-2.7.0]$ vi /tmp/a.txt
[mapr@n5 hadoop-2.7.0]$ hadoop fs -put /tmp/a.txt /tmp/a.txt
[mapr@n5 hadoop-2.7.0]$ ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt
bash: ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar: Permission denied

[mapr@n5 hadoop-2.7.0]$ vi /tmp/a.txt
```

```
[mapr@n5 hadoop-2.7.0]$ hadoop fs -put /tmp/a.txt /tmp/a.txt
[mapr@n5 hadoop-2.7.0]$ ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt
bash: ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar: Permission denied
[mapr@n5 hadoop-2.7.0]$ exit
[root@n5 conf]# cd /opt/mapr/hadoop/hadoop-2.7.0/
[root@n5 hadoop-2.7.0]# ll
```

```
[root@n5 hadoop-2.7.0]# ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt
-bash: ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar: Permission denied
[root@n5 hadoop-2.7.0]# su mapr
[mapr@n5 hadoop-2.7.0]$ maprlogin password
[Password for user 'mapr' at cluster 'maprcluster': ]
MapR credentials of user 'mapr' for cluster 'maprcluster' are written to '/tmp/maprticket_5000'
[mapr@n5 hadoop-2.7.0]$ ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt
bash: ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0-mapr-1808.jar: Permission denied
[mapr@n5 hadoop-2.7.0]$ cd ./share/hadoop/mapreduce/
[mapr@n5 mapreduce]$ ls -lrt
```

```
[mapr@n5 mapreduce]$ chmod 777 hadoop-mapreduce-examples-2.7.0-mapr-1808.jar
chmod: changing permissions of 'hadoop-mapreduce-examples-2.7.0-mapr-1808.jar': Operation not permitted
```

```
[mapr@n5 mapreduce]$ exit
[root@n5 hadoop-2.7.0]# cd ./share/hadoop/mapreduce/
[root@n5 mapreduce]# chmod 777 hadoop-mapreduce-examples-2.7.0-mapr-1808.jar
[root@n5 mapreduce]# ./hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt
invalid file (bad magic number): Exec format error
[root@n5 mapreduce]# ./hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt /tmp/output
invalid file (bad magic number): Exec format error
[root@n5 mapreduce]# yarn jar /hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt /tmp/output
Not a valid JAR: /hadoop-mapreduce-examples-2.7.0-mapr-1808.jar
```

```
[root@n5 mapreduce]# yarn jar ./hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt /tmp/output
```

```
[root@n5 mapreduce]# su mapr
```

```
[mapr@n5 mapreduce]$ yarn jar ./hadoop-mapreduce-examples-2.7.0-mapr-1808.jar wordcount /tmp/a.txt /tmp/output
```

```
18/11/15 09:50:11 INFO client.RMPProxy: Connecting to ResourceManager at nl.us-east1-b.c.main-form-217005.internal/10.142.0.3:8032
```

```
18/11/15 09:50:12 INFO input.FileInputFormat: Total input paths to process : 1
```

```
18/11/15 09:50:12 INFO mapreduce.JobSubmitter: number of splits:1
```

```
18/11/15 09:50:12 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1542263789247_0007
```

```
18/11/15 09:50:13 INFO security.ExternalTokenManagerFactory: Initialized external token manager class - com.mapr.hadoop.yarn.security.MapRTicketManager
```

```
18/11/15 09:50:13 INFO impl.YarnClientImpl: Submitted application application_1542263789247_0007
```

```
18/11/15 09:50:13 INFO mapreduce.Job: The url to track the job: https://nl.us-east1-b.c.main-form-217005.internal:8090/proxy/application_1542263789247_0007/
```

```
18/11/15 09:50:13 INFO mapreduce.Job: Running job: job_1542263789247_0007
```

```
18/11/15 09:50:25 INFO mapreduce.Job: Job job_1542263789247_0007 running in uber mode : false
```

```
18/11/15 09:50:25 INFO mapreduce.Job: map 0% reduce 0%
```

```
18/11/15 09:50:45 INFO mapreduce.Job: map 100% reduce 0%
```

```
18/11/15 09:50:58 INFO mapreduce.Job: map 100% reduce 100%
```

```
18/11/15 09:50:59 INFO mapreduce.Job: Job job_1542263789247_0007 completed successfully
```

```
18/11/15 09:50:59 INFO mapreduce.Job: Counters: 46
```

#### File System Counters

```
FILE: Number of bytes read=0
```

```
FILE: Number of bytes written=199163
```

```
FILE: Number of read operations=0
```

```
FILE: Number of large read operations=0
```

```
FILE: Number of write operations=0
```

```
MAPRFS: Number of bytes read=6482
```

```
MAPRFS: Number of bytes written=5596
```

```
MAPRFS: Number of read operations=595
```

```
MAPRFS: Number of large read operations=0
```

```
MAPRFS: Number of write operations=1605
```

#### Job Counters

Launched map tasks=1

MAPRFS: Number of bytes read=6482  
MAPRFS: Number of bytes written=5596  
MAPRFS: Number of read operations=595  
MAPRFS: Number of large read operations=0  
MAPRFS: Number of write operations=1605

#### Job Counters

Launched map tasks=1  
Launched reduce tasks=1  
Data-local map tasks=1  
Total time spent by all maps in occupied slots (ms)=17773  
Total time spent by all reduces in occupied slots (ms)=31101  
Total time spent by all map tasks (ms)=17773  
Total time spent by all reduce tasks (ms)=10367  
Total vcore-seconds taken by all map tasks=17773  
Total vcore-seconds taken by all reduce tasks=10367  
Total megabyte-seconds taken by all map tasks=18199552  
Total megabyte-seconds taken by all reduce tasks=31847424  
DISK\_MILLIS\_MAPS=8887  
DISK\_MILLIS\_REDUCE=13788

#### Map-Reduce Framework

Map input records=31  
Map output records=336  
Map output bytes=3560  
Map output materialized bytes=0  
Input split bytes=84  
Combine input records=336  
Combine output records=145  
Reduce input groups=145  
Reduce shuffle bytes=2044  
Reduce input records=145



```
Reduce output records=145
Spilled Records=290
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=292
CPU time spent (ms)=1010
Physical memory (bytes) snapshot=777093120
Virtual memory (bytes) snapshot=7567470592
Total committed heap usage (bytes)=1013710848

Shuffle Errors
    IO_ERROR=0
File Input Format Counters
    Bytes Read=2274
File Output Format Counters
    Bytes Written=1468
```

```
[mapr@n5 mapreduce]$
```

```
##### Hue #####
1.cp -r /usr/lib/python2.7/site-packages/google_compute_engine /opt/mapr/hue/hue-
4.2.0/build/env/lib/python2.7/site-packages/
2.yum install mapr-hue
```

Changed in the hue.init

```
[root@esekilx5638 conf]# diff --suppress-common-lines -y hue.ini hue.ini_16_11_2018
hive_server_host=esekilx5636.rnd.ki.sw.ericsson.se |
hive_server_host=esekilx5636.rnd.ki.sw.ericsson.se,esekilx5 |
#hive_conf_dir=/opt/mapr/hive/hive-2.1/conf | # hive_conf_dir=/opt/mapr/hive/hive-
2.1 |
mechanism=none | mechanism=${mechanism}
```

```
server_host=esekilx5634.rnd.ki.sw.ericsson.se      | ## server_host=localhost
server_port=21051                                  | ## server_port=21050
```

For google cloud:

```
cp -r /usr/lib/python2.7/site-packages/google_compute_engine /opt/mapr/hue/hue-4.2.0/build/env/lib/python2.7/site-packages/
```

```
#####dynamic allocation#####
```

```
#vi /opt/mapr/spark/spark-2.1.0/conf/spark-defaults.conf
```

```
# END OF THE SECURITY CONFIGURATION BLOCK
```

```
#####
```

```
spark.dynamicAllocation.enabled true
spark.shuffle.service.enabled true
spark.dynamicAllocation.minExecutors 0
spark.executor.instances 0
spark.authenticate true
spark.executor.heartbeatInterval 1800s
spark.network.timeout 2400s
```

```
more spark-defaults.conf
```

```
# Default system properties included when running spark-submit.
# This is useful for setting default environmental settings.
```

```
# Log effective Spark configuration at startup on INFO level
spark.logConf true
```

```
# Enable event logs for HistoryServer
```

```
spark.eventLog.enabled            true
spark.eventLog.dir                maprfs:///apps/spark/logs
spark.history.fs.logDirectory     maprfs:///apps/spark/logs

# Default location for Warehouse, if not using Hive
spark.sql.warehouse.dir /warehouse/spark

# Fix for SPARK-7819
spark.sql.hive.metastore.sharedPrefixes
com.mysql.jdbc,org.postgresql,com.microsoft.sqlserver,oracle.jdbc,com.mapr.fs.shim.LibraryLoader,com.map
r.security.
JNISecurity,com.mapr.fs.jni,com.mapr.fs.ShimLoader

spark.executor.memory             2g

spark.yarn.archive maprfs:///apps/spark/spark-jars.zip

spark.history.ui.port 18080
# SECURITY BLOCK
# ALL SECURITY PROPERTIES MUST BE PLACED IN THIS BLOCKG
# Security
# - ACLS

spark.acls.enable                true
spark.admin.acls                 mapr
spark.admin.acls.groups maprg
spark.authenticate.secret        changeMe
spark.authenticate true
spark.ssl.enabled true
spark.io.encryption.enabled      true
spark.io.encryption.keySizeBits 128
spark.ssl.fs.enabled true
spark.ssl.keyPassword mapr123
spark.ssl.keyStore /opt/mapr/conf/ssl_keystore
spark.ssl.keyStorePassword mapr123
```

```
spark.ssl.trustStore /opt/mapr/conf/ssl_truststore
spark.ssl.trustStorePassword mapr123
spark.ssl.protocol tls
spark.ssl.enabledAlgorithms TLS_RSA_WITH_AES_128_CBC_SHA,TLS_RSA_WITH_AES_256_CBC_SHA
spark.authenticate.enableSaslEncryption true
spark.network.sasl.serverAlwaysEncrypt true
#####dynamic allocation#####
# END OF THE SECURITY CONFIGURATION BLOCK
```

```
#####
spark.dynamicAllocation.enabled true
spark.shuffle.service.enabled true
spark.dynamicAllocation.minExecutors 0
spark.executor.instances 0
spark.authenticate true
spark.executor.heartbeatInterval 1800s
spark.network.timeout 2400s
```

```
[root@esekilxgp01 ~/dynamic_alloction]# clush -b -g dev 'cp -p /opt/mapr/hadoop/hadoop-
2.7.0/etc/hadoop/yarn-env.sh /opt/mapr/hadoop/hadoop-2.7.0/etc/hadoop/yarn-env.sh.14NOV2018.bak'
```

```
[root@esekilxgp01 ~/dynamic_alloction]# clush -b -g dev --copy yarn-env.sh --dest
/opt/mapr/hadoop/hadoop-2.7.0/etc/hadoop/
```

```
[root@esekilxgp01 ~/dynamic_alloction]# clush -b -g dev 'cp -p /opt/mapr/hadoop/hadoop-
2.7.0/etc/hadoop/yarn-site.xml /opt/mapr/hadoop/hadoop-2.7.0/etc/hadoop/yarn-site.xml.14NOV2018.bak'
```

```
[root@esekilxgp01 ~/dynamic_alloction]# clush -b -g dev --copy yarn-site.xml --dest
/opt/mapr/hadoop/hadoop-2.7.0/etc/hadoop/
```

```
clush -b -g dev 'ls -lrt /opt/mapr/spark/spark-2.1.0/yarn/spark-2.1.0-mapr-1707-yarn-shuffle.jar'
```

```
clush -b -g dev 'cp /opt/mapr/spark/spark-2.1.0/yarn/spark-2.1.0-mapr-1710-yarn-shuffle.jar
/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/yarn/lib/'
```

\$\$\$\$\$\$\$#####yarn-site.xml#####\$\$\$\$\$

```
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle,mapr_direct_shuffle,spark_shuffle</value>
</property>
```

```
<property>
  <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
  <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

```
  <property>
    <name>spark.authenticate</name>
    <value>true</value>
  </property>
```

--

```
<property>
<name>yarn.resourcemanager.webapp.address</name>
<value>8088</value>
</property>
```

## Mapr loglive ticket

```
maprlogin generateticket -type service -out /tmp/long_ticket -duration 3650:0:0 -renewal 3650:0:0 -user mapr
```

or

```
maprlogin generateticket -type servicewithimpersonation -user mapr -out /var/tmp/impersonation_ticket -duration 30:0:0 -renewal 90:0:0
```

<https://mapr.com/docs/52/SecurityGuide/GeneratingServiceWithImpersonationTicket.html>

```
operation failed: user has no associated credentials on the cluster: maprcluster
[root@master01 tmp]# su - mapr
Last login: Wed Dec 12 06:36:23 UTC 2018 on pts/0
[mapr@master01 ~]$ maprlogin password
[Password for user 'mapr' at cluster 'maprcluster': ]
MapR credentials of user 'mapr' for cluster 'maprcluster' are written to '/tmp/maprticket_5000'
[mapr@master01 ~]$ maprlogin generateticket -type servicewithimpersonation -user mapr -out /var/tmp/impersonation_ticket -duration 30:0:0 -renewal 90:0:0
MapR credentials of user 'mapr' for cluster 'maprcluster' are written to '/var/tmp/impersonation_ticket'
[mapr@master01 ~]$ cat /var/tmp/impersonation_ticket
maprcluster WZMQ23CL/jNVgIY60uGAZp/RIEXJn1Ij3J5B4ShwquxERJal3c1ErgPY2JWoqQXgPqQDYCiD/YB/HHqzQefudTvZdRIwzNGNBoxvNDa+cRTb6upRlX/wz9bIKiNDtnlfXG1yTf1hshdLr8Vfjd3TH7/
ElcfnrXi57vNF5X3qe4WLhYrWjSWHnTw3OYCqXilb0N8V7rCsntmHFpTwN8eVIRpuW03WSeoQCKZHWjXZ2yTsdIGcCcmLobDYQC4uMVFyHTckUwX/yh/xT/2ooAuljqCsjhFjg=
[mapr@master01 ~]$ █
```

```

-rw----- 1 mapr maprg 601 Nov 10 2017 db.conf
-rw-r--r-- 1 mapr maprg 19 Nov 10 2017 dbclient.conf
-rw-r--r-- 1 mapr maprg 35855 Nov 10 2017 date_time_zonespec.csv
-rw-r--r-- 1 mapr maprg 84 Nov 10 2017 cliregistry
-rw-r--r-- 1 mapr maprg 2438 Nov 10 2017 BaseLicense.txt
-rw-r--r-- 1 mapr maprg 2065 Nov 10 2017 BaseLicensePosixClient.txt
-rw-r--r-- 1 mapr maprg 17 Jan 23 2018 hostid.15303
-rw-r--r-- 1 mapr maprg 60 Jan 23 2018 hadoop_version
drwxr-xr-x 2 mapr maprg 72 Jan 23 2018 conf.d.new
-rw-r--r-- 1 mapr maprg 407 Jan 23 2018 mapr_fstab
-r----- 1 mapr maprg 42 Jan 23 2018 jmxremote.password
-r----- 1 mapr maprg 14 Jan 23 2018 jmxremote.access
-rw----- 1 mapr maprg 89 Jan 23 2018 cldb.key
-rw----- 1 mapr maprg 293 Jan 23 2018 maprserverticket
-r----- 1 mapr maprg 2099 Jan 23 2018 ssl_keystore
-r--r--r-- 1 mapr maprg 809 Jan 23 2018 ssl_truststore
-rw-r--r-- 1 mapr maprg 19 Jan 23 2018 mapr-memory.db
-rw-r--r-- 1 mapr maprg 730 Jan 23 2018 disktab
-rw----- 1 mapr maprg 301 Jan 23 2018 mapruserticket
-rw-r--r-- 1 mapr maprg 62 Jan 23 2018 mapr-ports.db
-rw-r--r-- 1 mapr maprg 9774 Jan 31 2018 mapr.login.conf.bak
-rw-r--r-- 1 mapr maprg 9717 Feb 7 2018 mapr.login.conf.old
-rw-r--r-- 1 mapr maprg 9717 Feb 7 2018 mapr.login.conf
drwxr-xr-x 2 root maprg 45 Mar 23 2018 proxy
-rw-r--r-- 1 mapr maprg 205 May 5 2018 env_override.sh
-rw-r--r-- 1 mapr maprg 98 Jun 28 13:58 mapr-clusters.conf
-rw-r--r-- 1 mapr maprg 41 Jun 28 13:58 mapr-monitoring.conf
drwxr-xr-x 2 mapr maprg 6 Sep 14 08:47 restart
lrwxrwxrwx 1 root maprg 55 Nov 20 07:13 ssl-client.xml -> /opt/mapr/ha
lrwxrwxrwx 1 root maprg 55 Nov 20 07:13 ssl-server.xml -> /opt/mapr/ha
-rw-r--r-- 1 root root 75 Nov 20 07:13 daemon.conf
-rw-r--r-- 1 mapr maprg 2292 Nov 20 07:13 cldb.conf
-rw-r--r-- 1 mapr maprg 809 Nov 20 07:13 mfs.conf
-rw-r--r-- 1 mapr maprg 1504 Nov 20 07:13 hadoop-metrics.properties
drwxr-xr-x 2 mapr maprg 4096 Nov 20 07:13 conf.old
drwxr-xr-x 2 mapr maprg 4096 Nov 20 07:14 conf.d
-rw-r--r-- 1 mapr maprg 3667 Nov 21 07:13 warden.conf
-rw-r--r-- 1 mapr maprg 20 Nov 21 07:14 clusterid
-rw----- 1 mapr maprg 0 Nov 27 06:41 mfsinstances_1

```

## OTSDB-72

**Issue:** The memory allocated to OpenTSDB can be insufficient, resulting in empty graphs and out-of-memory or `GC overhead limit exceeded` errors.

**Workaround:** Increase the default memory for OpenTSDB by making the following changes on all OpenTSDB nodes:

1. Edit the `/opt/mapr/conf/conf.d/warden.opentsdb.conf` file to change:

2. `service.heapsize.max=2000`

```
service.heapsize.min=2000
```

to

```
service.heapsize.max=6000
```

```
service.heapsize.min=6000
```

3. Edit the `/opt/mapr/opentsdb/opentsdb-*/etc/init.d/opentsdb` file to change:

4. `$`

```
{JVMXMX:=-Xmx2000m -Xss1m -XX:MaxMetaspaceSize=128m}
```

to

```
$
```

```
{JVMXMX:=-Xmx6000m -Xss1m -XX:MaxMetaspaceSize=128m}
```

5. Restart the OpenTSDB service:

```
maprccli node services -name opentsdb -nodes <space-separated list of OpenTSDB nodes> -action restart
```



#####Hue #####

```
1.cp -r /usr/lib/python2.7/site-packages/google_compute_engine /opt/mapr/hue/hue-4.2.0/build/env/lib/python2.7/site-packages/
2.yum install mapr-hue
3. cd /opt/mapr/hue/hue-4.2.0/desktop/conf
   vi hue.ini
```

```
4./opt/mapr/server/configure.sh -R
```

-----R and sparkR and Spark and hive -----

```
[root@esekilxgp05 ~]# yum install R
```

```
[root@esekilxgp05 ~]# which R
Set $HOME in /usr/bin and its show in env
[root@esekilxgp05 ~]# cat .bash_profile
# .bash_profile
```

```
# Get the aliases and functions
if [ -f ~/.bashrc ]; then
. ~/.bashrc fi
```

```
# User specific environment and startup programs
```

```
PATH=$PATH:$HOME/bin
```

```
export PATH
```

```
[root@esekilxgp05 ~]# env
```

```
PATH=/usr/lib64/qt-  
3.3/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin:/usr/java/jdk1.8.0_161//bin:/usr/java/
```

**For sparkR and Spark and hive-----**

```
[root@esekilxgp05 ~]# yum install -y mapr-spark mapr-hive  
00000
```

```
yum install mapr-spark mapr-spark-master mapr-spark-historyserver mapr-spark-thriftserver
```

```
[mapr@n2 bin]$ hadoop fs -mkdir /apps/spark  
[mapr@n2 bin]$ hadoop fs -chmod 777 /apps/spark  
[mapr@n2 bin]$ logout  
[root@n2 conf]# /opt/mapr/server/configure.sh -R
```

```
00000
```

```
[root@esekilxgp05 ~]# cd /opt/mapr/spark/spark-2.1.0/conf/
```

```
[root@esekilxgp05 ~]# ls -ltar
```

```
-rwxr-xr-x  1 mapr maprg 7191 Oct 11 07:51 spark-env.sh  
-rw-r--r--  1 mapr maprg 3727 Oct 11 07:51 hive-site.xml  
-rw-r--r--  1 mapr maprg 1749 Oct 11 07:52 spark-defaults.conf
```

```
[root@esekilxgp05 ~]# vi spark-defaults.conf
```

Add below files:

```
spark.yarn.dist.files = /opt/mapr/spark/spark-2.1.0/conf/hive-site.xml
```

```
spark.sql.hive.metastore.version = 1.2.0
```

```
[root@esekilxgp05 conf]# cat hive-site.xml
```

```
<?xml version="1.0"?>
```

```
<!--
```

```
    Licensed to the Apache Software Foundation (ASF) under one
    or more contributor license agreements.  See the NOTICE file
    distributed with this work for additional information
    regarding copyright ownership.  The ASF licenses this file  to
    you under the Apache License, Version 2.0 (the
```

```
    "License"); you may not use this file except in compliance
    with the License.  You may obtain a copy of the License at
```

```
    http://www.apache.org/licenses/LICENSE-2.0
```

```
    Unless required by applicable law or agreed to in writing, software
    distributed under the License is distributed on an "AS IS" BASIS,  WITHOUT
    WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  See the
    License for the specific language governing permissions and  limitations
    under the License.
```

```
-->
```

```
<configuration>
```

```
    <property>
```

```
        <name>javax.jdo.option.ConnectionURL</name>
```

```
        <value>jdbc:mysql://esekilxgp06.rnd.ki.sw.ericsson.se:3306/hive?createDatabaseIfNotExist=true</value>
```

```
<description>JDBC connect string for a JDBC metastore</description>
```

```
    </property>
```

```
    <property>
```

```
        <name>javax.jdo.option.ConnectionDriverName</name>
```

```
        <value>com.mysql.jdbc.Driver</value>
```

```
        <description>Driver class name for a JDBC metastore</description>
```

```
</property>
```

```
    <property>
```

```
        <name>javax.jdo.option.ConnectionUserName</name>
```

```
        <value>hive</value>
```

```
        <description>username to use against metastore database</description>
```

```
</property>
```

```

<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>CyT4cPPfwDs</value>
  <description>password to use against metastore database</description>

</property>
  <property>
    <name>hive.metastore.schema.verification</name>
    <value>>false</value>
  </property>

  <property>
    <name>hive.metastore.uris</name>
    <value>thrift://esekilx5636.rnd.ki.sw.ericsson.se:9083,thrift://esekilx5637.rnd.ki.sw.ericsson.se:9083</value>
  </property>
  <!-- For hive server2 -->
  <property>
    <name>hive.server2.enable.doAs</name>
    <value>true</value>
  </property>
  <!-- For hive server2 and meta store -->
  <property>
    <name>hive.metastore.execute.setugi</name>
    <value>true</value>
  </property>

  <property>
    <name>hive.metastore.warehouse.dir</name>
    <value>/project/rdi/warehouse/hive</value>
    <description>location of default database for the warehouse</description>
  </property>

  <property>
    <name>hive.metastore.try.direct.sql</name>
    <value>true</value>
    <description>

```

Whether the Hive metastore should try to use direct SQL queries instead of the DataNucleus for certain read paths. This can improve metastore performance when fetching many partitions or column statistics by orders of magnitude; however, it is not guaranteed to work on all RDBMS-es and all versions. In case of SQL failures, the metastore will fall back to the DataNucleus, so it's safe even if SQL doesn't work for all queries on your datastore. If all SQL queries fail (for example, your

metastore is backed by MongoDB), you might want to disable this to save the try-and-fall-back cost.

```
</description>
</property>
<property>
  <name>hive.metastore.client.socket.timeout</name>
  <value>1800s</value>
  <description>
    Expects a time value with unit (d/day, h/hour, m/min, s/sec, ms/msec, us/usec, ns/nsec), which is sec if not
    specified.
    MetaStore Client socket timeout in seconds
  </description>
</property>

<property>
<name>hive.metastore.sasl.enabled </name>
<value>false</value>
</property>

</configuration>
```

```
[root@esekilxgp05 conf]# cat spark-env.sh
```

```
#!/usr/bin/env bash

#
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with #
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at #
#   http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and #
# limitations under the License.
#
```

```

# This file is sourced when running various Spark programs.
# Copy it as spark-env.sh and edit that to configure Spark for your site.

# Options read when launching programs locally with
# ./bin/run-example or ./bin/spark-submit
# - HADOOP_CONF_DIR, to point Spark towards Hadoop configuration files
# - SPARK_LOCAL_IP, to set the IP address Spark binds to on this node
# - SPARK_PUBLIC_DNS, to set the public dns name of the driver program #
# - SPARK_CLASSPATH, default classpath entries to append

# Options read by executors and drivers running inside the cluster
# - SPARK_LOCAL_IP, to set the IP address Spark binds to on this node
# - SPARK_PUBLIC_DNS, to set the public DNS name of the driver program
# - SPARK_CLASSPATH, default classpath entries to append
# - SPARK_LOCAL_DIRS, storage directories to use on this node for shuffle and RDD data
# - MESOS_NATIVE_JAVA_LIBRARY, to point to your libmesos.so if you use Mesos
# Options read in YARN client mode
# - HADOOP_CONF_DIR, to point Spark towards Hadoop configuration files
# - SPARK_EXECUTOR_INSTANCES, Number of executors to start (Default: 2) #
# - SPARK_EXECUTOR_CORES, Number of cores for the executors (Default: 1).
# - SPARK_EXECUTOR_MEMORY, Memory per Executor (e.g. 1000M, 2G) (Default: 1G)
# - SPARK_DRIVER_MEMORY, Memory for Driver (e.g. 1000M, 2G) (Default: 1G)
# Options for the daemons used in the standalone deploy mode
# - SPARK_MASTER_HOST, to bind the master to a different IP address or hostname
# - SPARK_MASTER_PORT / SPARK_MASTER_WEBUI_PORT, to use non-default ports for the master
# - SPARK_MASTER_OPTS, to set config properties only for the master (e.g. "-Dx=y")
# - SPARK_WORKER_CORES, to set the number of cores to use on this machine
# - SPARK_WORKER_MEMORY, to set how much total memory workers have to give executors (e.g. 1000m, 2g)
# - SPARK_WORKER_PORT / SPARK_WORKER_WEBUI_PORT, to use non-default ports for the worker
# - SPARK_WORKER_INSTANCES, to set the number of worker processes per node
# - SPARK_WORKER_DIR, to set the working directory of worker processes
# - SPARK_WORKER_OPTS, to set config properties only for the worker (e.g. "-Dx=y")
# - SPARK_DAEMON_MEMORY, to allocate to the master, worker and history server themselves (default: 1g). #
# - SPARK_HISTORY_OPTS, to set config properties only for the history server (e.g. "-Dx=y")
# - SPARK_SHUFFLE_OPTS, to set config properties only for the external shuffle service (e.g. "-Dx=y")
# - SPARK_DAEMON_JAVA_OPTS, to set config properties for all daemons (e.g. "-Dx=y")
# - SPARK_PUBLIC_DNS, to set the public dns name of the master or workers
# Generic options for the daemons used in the standalone deploy mode
# - SPARK_CONF_DIR      Alternate conf dir. (Default: ${SPARK_HOME}/conf)
# - SPARK_LOG_DIR       Where log files are stored. (Default: ${SPARK_HOME}/logs)
# - SPARK_PID_DIR       Where the pid file is stored. (Default: /tmp)
# - SPARK_IDENT_STRING  A string representing this instance of spark. (Default: $USER)

```

```

# - SPARK_NICENESS      The scheduling priority for daemons. (Default: 0)
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground. It will not output a PID file.

##### #
Set MapR attributes and compute classpath
#####

# Set the spark attributes
if [ -d "/opt/mapr/spark/spark-2.1.0" ]; then
export SPARK_HOME=/opt/mapr/spark/spark-2.1.0 fi

# Load the hadoop version attributes
source /opt/mapr/spark/spark-2.1.0/mapr-util/hadoop-version-picker.sh
export HADOOP_HOME=$hadoop_home_dir export
HADOOP_CONF_DIR=$hadoop_conf_dir

# Enable mapr impersonation export
MAPR_IMPERSONATION_ENABLED=1

MAPR_HADOOP_CLASSPATH=`/opt/mapr/spark/spark-2.1.0/bin/mapr-classpath.sh`
MAPR_HADOOP_JNI_PATH=`hadoop jnipath`
MAPR_SPARK_CLASSPATH="$MAPR_HADOOP_CLASSPATH"

SPARK_MAPR_HOME=/opt/mapr

export SPARK_LIBRARY_PATH=$MAPR_HADOOP_JNI_PATH
export LD_LIBRARY_PATH="$MAPR_HADOOP_JNI_PATH:$LD_LIBRARY_PATH"

# Load the classpath generator script
source /opt/mapr/spark/spark-2.1.0/mapr-util/generate-classpath.sh

# Calculate hive jars to include in classpath generate_compatible_classpath
"spark" "2.1.0" "hive"
MAPR_HIVE_CLASSPATH=${generated_classpath} if
[ ! -z "$MAPR_HIVE_CLASSPATH" ]; then
    MAPR_SPARK_CLASSPATH="$MAPR_SPARK_CLASSPATH:$MAPR_HIVE_CLASSPATH"
fi
# Calculate hbase jars to include in classpath generate_compatible_classpath
"spark" "2.1.0" "hbase"
MAPR_HBASE_CLASSPATH=${generated_classpath} if
[ ! -z "$MAPR_HBASE_CLASSPATH" ]; then
    MAPR_SPARK_CLASSPATH="$MAPR_SPARK_CLASSPATH:$MAPR_HBASE_CLASSPATH"

```

```

    SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dspark.driver.extraClassPath=$MAPR_HBASE_CLASSPATH"
fi
# Set executor classpath for MESOS. Uncomment following string if you want deploy spark jobs on Mesos
#MAPR_MESOS_CLASSPATH=$MAPR_SPARK_CLASSPATH
SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dspark.executor.extraClassPath=$MAPR_HBASE_CLASSPATH:$MAPR_MESOS_CLASSPATH"

# Set SPARK_DIST_CLASSPATH
export SPARK_DIST_CLASSPATH=$MAPR_SPARK_CLASSPATH

# Security status source
/opt/mapr/conf/env.sh
if [ "$MAPR_SECURITY_STATUS" = "true" ]; then
    SPARK_SUBMIT_OPTS="$SPARK_SUBMIT_OPTS -Dhadoop.login=hybrid -Dmapr_sec_enabled=true" fi
#
scala
export SCALA_VERSION=2.11
export SPARK_SCALA_VERSION=$SCALA_VERSION export
SCALA_HOME=/opt/mapr/spark/spark-2.1.0/scala export
SCALA_LIBRARY_PATH=$SCALA_HOME/lib

# Use a fixed identifier for pid files
export SPARK_IDENT_STRING="mapr"
##### #
:::CAUTION::: DO NOT EDIT ANYTHING ON OR ABOVE THIS LINE
#####

#
# MASTER HA SETTINGS
#
#export SPARK_DAEMON_JAVA_OPTS="-Dspark.deploy.recoveryMode=ZOOKEEPER -Dspark.deploy.zookeeper.url=<zookeeperper1:5181,zookeep
Djava.security.auth.login.config=/opt/mapr/conf/mapr.login.conf -Dzookeeper.sasl.client=false"

# MEMORY SETTINGS
export SPARK_DAEMON_MEMORY=1g
export SPARK_WORKER_MEMORY=16g
# Worker Directory
export SPARK_WORKER_DIR=$SPARK_HOME/tmp

# Environment variable for printing spark command everytime you run spark.Set to "1" to print. #
export SPARK_PRINT_LAUNCH_COMMAND=1

```



```
/opt/mapr/spark/spark-2.2.1/bin
```

http://35.227.126.16:4040

[illegible]

Below are the instruction on how to install library for sparklyr incase asked in future.

- Go inside R as root.
- `install.packages("sparklyr")`
- choose 33 (for Sweden)
- it should work, incase dependencies not getting installed. We may need to do “yum install <>” for curl, xml, openssl, http packages.

For **"sparklyr.nested"** library

- Go inside R as root.
- `install.packages("devtools")`
- `devtools::install_github("mitre/sparklyr.nested")`
- `library(sparklyr.nested)`