Instalação Debian 8.9 Jessie

Instalação openjdk-8-jdk (normamelnte utilizamos o oracle jdk - oracle-j2sdk1.7)

$ apt-get install openjdk-8-jdk

$ echo $PATH

/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

Instalando cloudera-manager-daemons cloudera-manager-server

$ apt-get install cloudera-manager-daemons cloudera-manager-server

<https://www.jurisic.org/index.php?post/2017/02/28/How-to-install-PostgreSQL-9.6-server-on-Debian-8-Jessie>

The following packages have unmet dependencies:

cloudera-manager-agent : Depends: libssl1.0.0 (>= 1.0.1) but it is not installable

\*$ wget http://security.debian.org/debian-security/pool/updates/main/o/openssl/libssl1.0.0\_1.0.1t-1+deb8u11\_amd64.deb

$ dpkg -i libssl1.0.0\_1.0.1t-1+deb8u11\_amd64.deb

Instalando o Agent do cloudera

$ apt-get install cloudera-manager-agent

Editando o arquivo config do cloudera agent

$ vi /etc/cloudera-scm-agent/config.ini

Edit server\_host and server\_port

Subindo o serviço do cloudera agent

$ service cloudera-scm-agent start

Instalando o mariadb

$ apt-get install mariadb-server

Parando o serviço mariadb

$ service mariadb stop

Movendo o ib\_logfile0 e ib\_logfile1 para /root/ path

$ cd /var/lib/mysql/

$ mv ib\_logfile\* /root/

Editando o my.cnf do mysql

$ vi /etc/my.cnf

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

[mysqld]

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

transaction-isolation = READ-COMMITTED

# Disabling symbolic-links is recommended to prevent assorted security risks;

# to do so, uncomment this line:

symbolic-links = 0

# Settings user and group are ignored when systemd is used.

# If you need to run mysqld under a different user or group,

# customize your systemd unit file for mariadb according to the

# instructions in http://fedoraproject.org/wiki/Systemd

######################

# key\_buffer = 16M

######################

key\_buffer\_size = 32M

max\_allowed\_packet = 32M

thread\_stack = 256K

thread\_cache\_size = 64

query\_cache\_limit = 8M

query\_cache\_size = 64M

query\_cache\_type = 1

max\_connections = 550

#expire\_logs\_days = 10

#max\_binlog\_size = 100M

#log\_bin should be on a disk with enough free space.

#Replace '/var/lib/mysql/mysql\_binary\_log' with an appropriate path for your

#system and chown the specified folder to the mysql user.

log\_bin=/var/lib/mysql/mysql\_binary\_log

#In later versions of MariaDB, if you enable the binary log and do not set

#a server\_id, MariaDB will not start. The server\_id must be unique within

#the replicating group.

server\_id=1

binlog\_format = mixed

read\_buffer\_size = 2M

read\_rnd\_buffer\_size = 16M

sort\_buffer\_size = 8M

join\_buffer\_size = 8M

# InnoDB settings

innodb\_file\_per\_table = 1

innodb\_flush\_log\_at\_trx\_commit = 2

innodb\_log\_buffer\_size = 64M

innodb\_buffer\_pool\_size = 4G

innodb\_thread\_concurrency = 8

innodb\_flush\_method = O\_DIRECT

Comentar daqui pra baixo pq essas porras nao funcionam

#innodb\_log\_file\_size = 512M

#[mysqld\_safe]

#log-error=/var/log/mariadb/mariadb.log

#pid-file=/var/run/mariadb/mariadb.pid

#

# include all files from the config directory

#

#!includedir /etc/my.cnf.d

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

$ service mariadb start

$ /usr/bin/mysql\_secure\_installation \*\*\*\*\*\*Senha default do banco password

Change the root password? [Y/n] y

New password:

Re-enter new password:

Password updated successfully!

Reloading privilege tables..

... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone

to log into MariaDB without having to have a user account created for

them. This is intended only for testing, and to make the installation

go a bit smoother. You should remove them before moving into a

production environment.

Remove anonymous users? [Y/n] y

... Success!

Normally, root should only be allowed to connect from 'localhost'. This

ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] n

... skipping.

By default, MariaDB comes with a database named 'test' that anyone can

access. This is also intended only for testing, and should be removed

before moving into a production environment.

Remove test database and access to it? [Y/n] y

- Dropping test database...

... Success!

- Removing privileges on test database...

... Success!

Reloading the privilege tables will ensure that all changes made so far

will take effect immediately.

Reload privilege tables now? [Y/n] y

... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB

installation should now be secure.

Thanks for using MariaDB!

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

$ apt-get install libmysql-java

$ mysql -u root -p

CREATE DATABASE scm DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4\_general\_ci;

CREATE DATABASE amon DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4\_general\_ci;

CREATE DATABASE rman DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4\_general\_ci;

CREATE DATABASE nav DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4\_general\_ci;

CREATE DATABASE navms DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4\_general\_ci;

CREATE DATABASE hue DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci;

CREATE DATABASE metastore DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci;

CREATE DATABASE sentry DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci;

CREATE DATABASE oozie DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci;

GRANT ALL ON scm.\* TO 'scm'@'%' IDENTIFIED BY 'scm';

GRANT ALL ON amon.\* TO 'amon'@'%' IDENTIFIED BY 'amon';

GRANT ALL ON rman.\* TO 'rman'@'%' IDENTIFIED BY 'rman';

GRANT ALL ON nav.\* TO 'nav'@'%' IDENTIFIED BY 'nav';

GRANT ALL ON navms.\* TO 'navms'@'%' IDENTIFIED BY 'navms';

GRANT ALL ON hue.\* TO 'hue'@'%' IDENTIFIED BY 'hue';

GRANT ALL ON metastore.\* TO 'hive'@'%' IDENTIFIED BY 'hive';

GRANT ALL ON sentry.\* TO 'sentry'@'%' IDENTIFIED BY 'sentry';

GRANT ALL ON oozie.\* TO 'oozie'@'%' IDENTIFIED BY 'oozie';

SHOW DATABASES;

SHOW GRANTS FOR 'scm'@'%';

SHOW GRANTS FOR 'amon'@'%';

SHOW GRANTS FOR 'rman'@'%';

SHOW GRANTS FOR 'nav'@'%';

SHOW GRANTS FOR 'navms'@'%';

SHOW GRANTS FOR 'hue'@'%';

SHOW GRANTS FOR 'hive'@'%';

SHOW GRANTS FOR 'sentry'@'%';

SHOW GRANTS FOR 'oozie'@'%';

CREATE USER 'temp'@'%' IDENTIFIED BY 'password';

GRANT ALL ON \*.\* TO 'temp'@'%' WITH GRANT OPTION;

quit;

$ /usr/share/cmf/schema/scm\_prepare\_database.sh mysql -utemp -p --verbose scm scm scm Enter database password:

JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64

Verifying that we can write to /etc/cloudera-scm-server

Database type: mysql

Database user: temp

Executing: /usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java -cp /usr/share/java/mysql-connector-java.jar:/usr/share/java/oracle-connector-java.jar:/usr/share/java/postgresql-connector-java.jar:/usr/share/cmf/schema/../lib/\* com.cloudera.enterprise.dbutil.DbProvisioner --create -h localhost -u temp -H localhost -U scm -d scm -t mysql

Creating SCM configuration file in /etc/cloudera-scm-server

Created db.properties file:

# Auto-generated by scm\_prepare\_database.sh on Thu Mar 21 23:05:22 -03 2019

#

# For information describing how to configure the Cloudera Manager Server

# to connect to databases, see the "Cloudera Manager Installation Guide."

#

com.cloudera.cmf.db.type=mysql

com.cloudera.cmf.db.host=localhost

com.cloudera.cmf.db.name=scm

com.cloudera.cmf.db.user=scm

com.cloudera.cmf.db.setupType=EXTERNAL

com.cloudera.cmf.db.password=scm

Executing: /usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java -cp /usr/share/java/mysql-connector-java.jar:/usr/share/java/oracle-connector-java.jar:/usr/share/java/postgresql-connector-java.jar:/usr/share/cmf/schema/../lib/\* com.cloudera.enterprise.dbutil.DbCommandExecutor /etc/cloudera-scm-server/db.properties com.cloudera.cmf.db.

2019-03-21 23:05:25,305 [main] INFO com.cloudera.enterprise.dbutil.DbCommandExecutor - Successfully connected to database.

All done, your SCM database is configured correctly!

#$ /usr/share/cmf/schema/scm\_prepare\_database.sh mysql scm scm scm

root@debian9:/var/lib/mysql# mysql -u root -ppassword

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 19

Server version: 10.1.37-MariaDB-0+deb9u1 Debian 9.6

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 scm;

Database changed

MariaDB [scm]> show tables;

Empty set (0.00 sec)

$ service cloudera-scm-server start

Open another terminal

$ tail -f /var/log/cloudera-scm-server/cloudera-scm-server.