```
These are the commands for AWS CM install
** update yum
sudo yum update
sudo yum install -y wget

** add centos to sudoers

sudo visudo
        add -> centos ALL=(ALL) ALL

** Change the run level to multi-user text mode
sudo systemctl get-default
sudo systemctl set-default multi-user.target

** Disable firewall
sudo systemctl disable firewalld
sudo systemctl status firewalld


** Change VM Swappiness to 1
cat /proc/sys/vm/swappiness
sudo sysctl -w vm.swappiness=1

** Change VM Swappiness permanently
sudo vi /etc/sysctl.conf
        add ->
vm.swappiness=1

** Disable SE Linux
sudo vi /etc/selinux/config
        change-> SELINUX=disabled


** Disable Transparent Hugepage Support
sudo vi /etc/rc.d/rc.local
        add ->
echo "never" > /sys/kernel/mm/transparent_hugepage/enabled
echo "never" > /sys/kernel/mm/transparent_hugepage/defrag

sudo chmod +x /etc/rc.d/rc.local
sudo vi /etc/default/grub
   add -> transparent_hugepage=never (on line GRUB_CMDLINE_LINUX )
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
sudo systemctl start tuned
sudo tuned-adm off
sudo tuned-adm list
sudo systemctl stop tuned
sudo systemctl disable tuned

** Check to see that nscd service is running
sudo yum install -y nscd
sudo systemctl enable nscd
sudo systemctl start nscd
sudo systemctl status nscd
```

```
** Check to see that ntp service is running
sudo systemctl stop chronyd
sudo systemctl disable chronyd
sudo yum -y install ntp
sudo systemctl enable ntpd.service
sudo systemctl start ntpd.service
sudo ntpdate -u 0.rhel.pool.ntp.org
sudo hwclock --systohc
sudo systemctl status ntpd.service

** check to make sure that File lookup has priority
sudo vi /etc/nsswitch.conf

** Disable IPV6
sudo sysctl -w net.ipv6.conf.all.disable_ipv6=1
sudo sysctl -w net.ipv6.conf.default.disable_ipv6=1

** check hostname resolution
** first put something in /etc/hosts and then

getent hosts cm.skcc.com

** setup a password for centos
sudo passwd centos
sudo vi /etc/ssh/sshd_config
        change ->
PasswordAuthentication yes
sudo systemctl restart sshd.service

** If desired, setup passwordless key entry from CM node
ssh-keygen -t rsa
ssh cm mkdir -p .ssh
ssh dn1 mkdir -p .ssh
ssh dn2 mkdir -p .ssh
ssh dn3 mkdir -p .ssh
cat .ssh/id_rsa.pub | ssh mn  'cat >> .ssh/authorized_keys'
cat .ssh/id_rsa.pub | ssh dn1 'cat >> .ssh/authorized_keys'
cat .ssh/id_rsa.pub | ssh dn2 'cat >> .ssh/authorized_keys'
cat .ssh/id_rsa.pub | ssh dn3 'cat >> .ssh/authorized_keys'

** Update /etc/host
sudo vi /etc/hosts
        add ->
******* THIS IS PUBLIC IP
52.79.36.119     cm.skcc.com      cm
52.79.37.179     mn.skcc.com      mn
13.124.128.147   dn1.skcc.com     dn1
13.124.243.28    dn2.skcc.com     dn2
13.125.74.115    dn3.skcc.com     dn3

******* THIS IS PRIVATE IP
172.31.15.244    cm.skcc.com      cm
172.31.2.225     mn.skcc.com      mn
```

```
172.31.7.42      dn1.skcc.com     dn1
172.31.10.138    dn2.skcc.com     dn2
172.31.0.219     dn3.skcc.com     dn3


** Change the hostname on each node
sudo hostnamectl set-hostname cm.skcc.com
hostname -f

sudo hostnamectl set-hostname mn.skcc.com
hostname -f

sudo hostnamectl set-hostname dn1.skcc.com
hostname -f

sudo hostnamectl set-hostname dn2.skcc.com
hostname -f

sudo hostnamectl set-hostname dn3.skcc.com
hostname -f

** REBOOT THE SERVER AND CHECK

** Install JDK

Sudo yum -y install oracle-j2sdk1.7
java -version

** Configure repository
sudo yum install -y wget
sudo wget https://archive.cloudera.com/cm5/redhat/7/x86_64/cm/
cloudera-manager.repo \
-P /etc/yum.repos.d/

** change the baseurl within cloudera-manager.repo to fit the
version you want to install
baseurl=https://archive.cloudera.com/cm5/redhat/7/x86_64/cm/5.7.4/
for example: https://archive.cloudera.com/cm5/redhat/7/x86_64/cm/
5.15.2/

sudo rpm --import \
https://archive.cloudera.com/cm5/redhat/7/x86_64/cm/RPM-GPG-KEY-
cloudera


** Install Cloudera Manager
sudo yum install -y cloudera-manager-daemons cloudera-manager-server

** Install MariaDB

sudo yum install -y mariadb-server
{ // use this repo in case yum install does not work
sudo vi /etc/yum.repos.d/MariaDB.repo
        add ->
```

```
sudo rpm --import https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
}
sudo vi /etc/my.cnf
        add ->
****************************************** Don't add this line
[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
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
****************************************** Don't add this line
sudo systemctl enable mariadb
sudo systemctl start mariadb
sudo /usr/bin/mysql_secure_installation

** Install mysql connector
// From the mac
scp -i ~/KeyPair/SEBC_HP.pem ~/Downloads/mysql-connector-
java-5.1.47.tar.gz centos@acm:.
// From acm node
cd /home/centos
tar zxvf mysql-connector-java-5.1.47.tar.gz
sudo mkdir -p /usr/share/java/
cd mysql-connector-java-5.1.47
sudo cp mysql-connector-java-5.1.47-bin.jar /usr/share/java/mysql-
connector-java.jar

** Create the databases and users in MariaDB
mysql -u root -p

CREATE DATABASE scm DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON scm.* TO 'scm-user'@'%' IDENTIFIED BY 'password';

CREATE DATABASE amon DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON amon.* TO 'amon-user'@'%' IDENTIFIED BY 'password';

CREATE DATABASE rman DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON rman.* TO 'rman-user'@'%' IDENTIFIED BY 'password';

CREATE DATABASE hue DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON hue.* TO 'hue-user'@'%' IDENTIFIED BY 'password';

CREATE DATABASE metastore DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON metastore.* TO 'metastore-user'@'%' IDENTIFIED BY
```

```
'password';

CREATE DATABASE sentry DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON sentry.* TO 'sentry-user'@'%' IDENTIFIED BY 'password';

CREATE DATABASE oozie DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL ON oozie.* TO 'oozie-user'@'%' IDENTIFIED BY 'password';

FLUSH PRIVILEGES;
SHOW DATABASES;
EXIT;

** Setup the CM database
sudo /usr/share/cmf/schema/scm_prepare_database.sh mysql scm scm-
user password
sudo rm /etc/cloudera-scm-server/db.mgmt.properties
sudo systemctl start cloudera-scm-server
```