# Cassandra Installation and Clustering

The below section covers the Installation of Cassandra on multiple Nodes of clustering the nodes on a distributed environment.

**Pre-requisites:**

* JDK 64 Bit
* Cassandra Installables

**Installation Steps:**

## Step 1: Verifying Java Installation

Java is the main prerequisite for Hadoop. Verify the existence of java in your system using “java -version”

# java -version

If java is not installed in your system, then follow the steps for installing java.

## Step 2: Creating User Account

We used “rathinamh” user created by YNAP

## Step 3: Cassandra installation

download cassandra tar file and extract it

sudo wget http://archive.apache.org/dist/cassandra/3.9/apache-cassandra-3.9-bin.tar.gz

tar xzvf apache-cassandra-3.9-bin.tar.gz

## Step 4: Configure Cassandra Cluster

create the data folder in the /opt/apache-cassandra-3.9/ and inside that folder create folders for commitlog, data, hints & saved\_caches using mkdir command and give the permissions with the following command(you have to do this in both machines where you are going to install Cassandra)

>sudo chmod -R 777 data

set the cassandra path in .bashrc

export CASSANDRA\_HOME=/opt/apache-cassandra-3.9

export PATH=$PATH:$CASSANDRA\_HOME/bin

edit the $CASSANDRA\_HOME/conf/cassandra.yaml file in all the cluster nodes

cluster\_name: 'Test cluster'

seeds: "10.228.217.45,10.228.218.176"

listen\_address: 10.228.217.45

rpc\_address: 10.228.217.45

startrpc:true

broadcast\_rpc\_address:10.228.217.45

enable\_user\_defined\_functions: true

enable\_scripted\_user\_defined\_functions: true

for the the node with ipaddress 10.228.218.176 change the configurations in the place of 10.228.217.45

open the terminal and type the command in both systems

>cassandra or >cassandra -f

this terminal will wait for the client process called cqlsh

open another terminal and type the command

>cqlsh ipaddress

open another terminal and check the cluster status by following command

>nodetool status

if it gives the output u/n with all nodes in cluster if up and normal, cluster is successfully formeds

## Step 5: Starting the Nodes in the Cluster

After you have installed and configured Cassandra on all nodes, start the seed nodes one at a time, and then start the rest of the nodes.

Note: If the node has restarted because of automatic restart, you must first stop the node and clear the data directories

Starting Cassandra Node:

# cd /opt/apache-cassandra-3.9/bin

# cassandra

## Step 6: Verifying the Cassandra Cluster

To check that the ring is up and running, run:

Package installations:

# nodetool status

Each node should be listed and it's status and state should be UN (Up Normal).

Datacenter: dc1

===============

Status=Up/Down

|/ State=Normal/Leaving/Joining/Moving

-- Address Load Tokens Owns Host ID Rack

UN 10.228.217.45 23.71 MB 256 ? 4cd9e3fc-f3a9-4f09-a2cb-23e20560282f rack1

UN 10.228.218.176 24.69 MB 256 ? faa774b2-46f8-4c79-b056-34c73cbfcf3b rack1

Note: Non-system keyspaces don't have the same replication settings, effective ownership information is meaningless

## Step 7: Connecting to CQLSH

Connect to a specific node using cqlsh

# cqlsh 10.228.217.45

Connected to MasterCluster at 10.228.217.45:9042.

[cqlsh 5.0.1 | Cassandra 3.0.9 | CQL spec 3.4.0 | Native protocol v4]

Use HELP for help.

cqlsh>

# Integration of Hadoop & Cassandra

Go to machine “full.poc01.dld.ewe1” and in terminal run the following command for Hadoop to integrate with Cassandra

spark-submit --class com.ynap.datalake.RealTimeStreaming --master spark://10.228.217.45:7077 --deploy-mode client <path of the file> StreamingLOCAL.jar

Note: IBM to provide the file StreamingLOCAL.jar which will be used for connection

# Elastic Search Installation and Clustering

The below section covers the Installation of Elastic search on multiple Nodes of Elastic search Cluster (Master- Slave) on a distributed environment.

**Pre-requisites:**

Elasticsearch requires java to run. Install Oracle java or OpenJDK. Check the Elasticsearch support matrix to make sure it will work.

Elasticsearch support matrix: https://www.elastic.co/support/matrix#show\_jvm

**Installation Steps:**

## Step 1: Verifying Java Installation

Java is the main prerequisite for Hadoop. Verify the existence of java in your system using “java -version”

# java -version

If java is not installed in your system, then follow the steps for installing java.

## Step 2: Elastic Search installation

first import the Elasticsearch GPG key with:

rpm --import http://packages.elastic.co/GPG-KEY-elasticsearch

Next, open a blank text file in your favourite text editor (be sure to run the editor as root, so that you have the necessary save permissions) and fill it with these lines:

name=Elasticsearch repository for 2.x packages

baseurl=http://packages.elastic.co/elasticsearch/2.x/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

Save the file as /etc/yum.repos.d/elasticsearch.repo

Next, install the Elasticsearch package with:

sudo yum -y install elasticsearch

## Step 3: Configure and Start Elasticsearch

We should tweak Elasticsearch a little and start it up.

Specifically, we want to prevent outside connections to the Elasticsearch HTTP API. To do this, open the file /etc/elasticsearch/elasticsearch.yml in an editor, set the below configuration. Under network section, change the “network.host” parameter with the IP address of your VM (or localhost). Then save the file and exit.

You can now start/stop/check status Elasticsearch using the following commands:

* sudo service elasticsearch start
* sudo service elasticsearch stop
* sudo service elasticsearch status

Verify the elasticsearch service by sending a HTTP request to port 9200. By default, elasticsearch run on port 9200.

*curl* <http://localhost:9200>

[root@pcs\_virtual\_01 ~]# curl http://localhost:9200

{

"name" : "es-master-01",

"cluster\_name" : "ynap\_cluster",

"cluster\_uuid" : "bu4WI6PbSmGhT69Ku64RXg",

"version" : {

"number" : "2.4.4",

"build\_hash" : "fcbb46dfd45562a9cf00c604b30849a6dec6b017",

"build\_timestamp" : "2017-01-03T11:33:16Z",

"build\_snapshot" : false,

"lucene\_version" : "5.5.2"

},

"tagline" : "You Know, for Search"

}

## Step 4: Setup and Elastic Search Cluster

The below steps are to setup a basic Elasticsearch cluster of 2 nodes (one Master node and one Data node). For this setup to work, as a prerequisite, you need two virtual machines with enough memory.

Do the following before we start configuring the server for elasticsearch.

1. Create 2 VMs (say the IPs are: 192.168.4.1, 192.168.4.2) installed with Elasticsearch.
2. Setting up the Master Node:

* All the elasticsearch configurations are present in elasticsearch.yml file, which is located in /etc/elasticsearch folder. Now, the elasticsearch.yml file has to be edited for the configuring the node as a Master node. The configuration file has many sections like cluster, node, paths etc.
* Under the cluster section, change the cluster name parameter.

cluster.name: es\_cluster

* Under node section, change the node name parameter and add other parameters as shown below.

node.name: es-master-01

node.master: true

node.data: false

* Under network section, change the “network.host” parameter with the IP address of your master node.

network. host: 192.168.4.1

* Under discover section add the following:

discovery.zen.ping.multicast.enabled: false

discovery.zen.ping.unicast.hosts: ["192.168.4.1", "192.168.4.2"]

The above parameters disable the multicast and send unicast message to the specified hosts.

1. Setting up the Data node:

* Follow all the steps we used to setup the Master node for the data node. Only while configuring the elasticsearch.yml file just uses the data given below. All the other steps are same for all the nodes.
* Under node section, add the following

node.name: es-data-01

node.client: false

node.data: true

If you do not specify a node name in the configuration, elasticsearch assigns a random name on every restart.

* Under network section, change the “network.host” parameter with the IP address of your data node.

network.host: 192.168.4.2

1. Once you configure all the three nodes, restart the elasticsearch service on all the three nodes.

sudo service Elasticsearch restart

1. Now you will have working elasticsearch cluster.

# Node JS Installation

The below section covers the Installation of Node JS instance in an environment.

## Step 1: Install Node.js using Yum

Install the Node.js using yum as shown below. This will also ask you to review and accept the Nodesource GPG key before installing nodejs.

# yum install nodejs

...

...

Public key for nodejs-0.10.40-1nodesource.el7.centos.x86\_64.rpm is not installed

nodejs-0.10.40-1nodesource.el7.centos.x86\_64.rpm | 4.5 MB 00:01

Retrieving key from file:///etc/pki/rpm-gpg/NODESOURCE-GPG-SIGNING-KEY-EL

Importing GPG key 0x34FA74DD:

Userid : "NodeSource <gpg-rpm@nodesource.com>"

Fingerprint: 2e55 207a 95d9 944b 0cc9 3261 5ddb e8d4 34fa 74dd

Package : nodesource-release-el7-1.noarch (installed)

From : /etc/pki/rpm-gpg/NODESOURCE-GPG-SIGNING-KEY-EL

Is this ok [y/N]: y

..

..

Installing : nodejs-0.10.40-1nodesource.el7.centos.x86\_64

Complete!

## Step 2: Verify Node.js Installation

Verify that the node.js and npm are successfully installed.

# node -v

v0.10.40

# npm version

{ http\_parser: '1.0',

node: '0.10.40',

v8: '3.14.5.9',

ares: '1.9.0-DEV',

uv: '0.10.36',

zlib: '1.2.8',

modules: '11',

openssl: '1.0.1p',

npm: '1.4.28' }

In this method, you’ll see that node and npm executable are installed under /usr/bin directory.

# whereis node

node: /usr/bin/node /usr/share/node /usr/share/man/man1/node.1.gz

# whereis npm

npm: /usr/bin/npm

# Reference

CDC - <https://www.ibm.com/support/knowledgecenter/en/SSTRGZ_11.4.0/com.ibm.cdcdoc.installingasandmc.doc/tasks/installMC.html>

<https://www.ibm.com/support/knowledgecenter/en/SSTRGZ_11.4.0/com.ibm.cdcdoc.cdcfororacle.doc/tasks/installingcdc_unixandlinux.html>

Hadoop - <https://hadoop.apache.org/docs/r2.7.1/>

Hadoop high availability - https://www.edureka.co/blog/how-to-set-up-hadoop-cluster-with-hdfs-high-availability/

Spark - <http://data-flair.training/blogs/apache-spark-installation-on-multi-node-cluster-step-by-step-guide/>

Cassandra - https://www.unixmen.com/step-step-procedure-installing-cassandra/

Cassandra - <https://docs.datastax.com/en/cassandra/3.9/cassandra/initialize/initSingleDS.html>

Elastic search - <https://developers.redhat.com/blog/2016/06/07/how-to-install-elastic-stack-elk-on-red-hat-enterprise-linux-rhel/>

Node JS - <http://www.thegeekstuff.com/2015/10/install-nodejs-npm-linux/>