POD

Instalação e Configuração do Framework Hadoop

Versão: 04 de maio/2019

Créditos:

MsC. Jorge Ximendes

Dr. Cláudio Fernando Resin Geyer

Dr. Julio C. S. dos Anjos

Agenda

- Recomendações
- Instalação do Java
- Instalação do Hadoop
- Configuração do Hadoop
- Exemplo: **wordcount**

Recomendações:

- Instalar - Java 8 em diante (observar versão do Hadoop recomendações do Java)

- Ter espaço em disco suficiente para HDFS no caso de máquina single mode. (Máquina para testes).

- Usar S.O. Linux - preferencialmente

- Arquivos especiais para o Deploy e máquinas virtuais disponíveis em:
  http://www.inf.ufrgs.br/~jcsanjos/BigData/

Java

Java

Para instalar o java 8 no linux:

sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update

sudo apt-get install oracle-java8-installer
ou
sudo apt-get install openjdk-8-jre

Após a instalação do Java, adicionar ao /etc/profile:

update-alternatives --config java → indica caminho

export JAVA_HOME= "/usr/lib/jvm/java-8-openjdk-amd64/"

Hadoop

Instalação

Hadoop

Neste tutorial iremos utilizar a versão 2.9.2

Baixar do Apache ou de seus mirrors:(https://hadoop.apache.org/releases.html):

wget <endereço_do_site>

wget http://mirror.nbtelecom.com.br/apache/hadoop/common/hadoop-2.9.2/hadoop-2.9.2.tar.gz

Descompactar em /opt

tar -xzf hadoop-2.9.2.tar.gz

Hadoop Configurações Ambiente

Ir até o arquivo hadoop-env.sh

cd /opt/hadoop-2.9.2/etc/hadoop

nano hadoop-env.sh

Hadoop Configurações Ambiente

No hadoop-env.sh encontrar:

The java implementation to use. By default, this environment
variable is REQUIRED on ALL platforms except OS X!
 # export JAVA_HOME=

Remova o **#** do export e acrescente o caminho de instalação do Java

Hadoop Configurações Ambiente

Verificar a instalação do Hadoop:

cd <pasta da sua escolha>/bin

./hadoop checknative -a

./hadoop version

Configuração

Hadoop

Antes de configurar o Hadoop, devemos configurar a máquina para podermos fazer ssh no localhost.

Digitar o seguinte comando e colocar a senha

ssh root@localhost ou **ssh root@ip**

Comando para criar uma chave de ssh para acesso automático sem pedir senha, no próximo slide.

Hadoop

Gerar **chaves privada** do ssh:

ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa

Para copia a **chave pública** no arquivo de autorizações em cada máquina:

cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

chmod 0600 ~/.ssh/authorized_keys

Hadoop

Para acessar as máquinas remotamente sem pedir senha **deve-se copiar a chave publica** do ssh:

ssh root@$n1 'cat /root/.ssh/id_rsa.pub'|ssh root@$n2 'cat >> /root/.ssh/authorized_keys'

n1 = Ip source

n2 = IP destination

Hadoop

Agora temos duas possibilidades de configurar o Hadoop:

Hadoop

1. **Local**: o Hadoop executa somente na máquina em que está instalado;
2. **Cluster**: o hadoop executa sobre um conjunto de máquinas.

Hadoop

Navegar a até a pasta onde estão os binários do Hadoop. E depois ir até a pasta “etc/hadoop/”.

Hadoop

Para configurar em modo cluster partimos da configuração em **modo local**.

http://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html

Hadoop

Editar os seguintes arquivos:

core-site.xml, hdfs-site.xml, yarn-site.xml, slaves

Abrir o arquivo mapred-site.xml.template e salvá-lo como mapred-site.xml.

Hadoop

core-site.xml

<property>

<name>hadoop.tmp.dir</name>

<value>**/home/<usuário>/tmp**</value>

<description> Local onde fica a pasta criada para armazenar os arquivos do HDFS. Acima está um exemplo de um local nos **data_nodes**. </description>

</property>

<property>

<name>fs.defaultFS</name>

<value>**hdfs://localhost:54310**</value>

<description>máquina e porta utilizada pelo HDFS. Para configurar localmente "localhost" de outra forma colocar nome da máquina Master</description>

</property>

Hadoop

hdfs-site.xml

<property>

<name>dfs.replication</name>

<value>1</value>

<description>Localmente, aqui é colocado
o número 1 caso contrário 3 (padrão)
</description>

</property>

Hadoop

Abrir o arquivo mapred-site.xml.template e salvá-lo como mapred-site.xml

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

<description>Definição do scheduler a ser utilizado pelo

Hadoop</description>

</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>localhost:8025</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado localmente vai o “localhost”</description>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>localhost:8035</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado localmente vai o “localhost”</description>

</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.resourcemanager.address</name>

<value>localhost:8050</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado localmente vai o “localhost”</description>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce_shuffle</value>

</property>

Hadoop

yarn-site.xml

<property>
<name>yarn.nodemanager.resource.cpu-vcores</name>
<value>N</value>
<description>Diz ao nodemanager do YARN quantos cores N a disposição dele para executar as tarefas do Hadoop nesta máquina. O usuário indica quantos N cores ele disponibiliza. Pode variar de máquina para máquina no cluster.</description>
</property>

<property>
<name>yarn.nodemanager.resource.memory-mb</name>
<value>M</value>
<description>Diz ao nodemanager do YARN quanto de memória RAM M está a disposição dele para executar as tarefas do Hadoop nesta máquina. O usuário indica quanto de M de memória ele disponibiliza. Pode variar de máquina para máquina no cluster.</description>
</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.scheduler.minimum-allocation-mb</name>

<value>m</value>

<description>Diz ao scheduler do YARN quanto de memória RAM mínima m ele deve alocar para executar as tarefas do Hadoop no cluster. Pedimos menores que este são recusados pelo YARN. O usuário indica de quanto será está quantidade de memória m</description>

</property>

<property>

<name>yarn.scheduler.maximum-allocation-mb</name>

<value>M</value>

<description>Diz ao scheduler do YARN quanto de memória RAM máxima M ele deve alocar para executar as tarefas do Hadoop. Pedimos maiores que este são recusados pelo YARN. O usuário indica quanto de será está quantidade de memória M, mas o correto é que deva este valor seja igual a soma de memória disponibilizada no cluster.</description>

</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.scheduler.minimum-allocation-vcores</name>

<value>c</value>

<description>Diz ao scheduler do YARN quantos cores de processamento mínimo c ele deve alocar para executar as tarefas do Hadoop. Pedimos menores que este são recusados pelo YARN. O usuário indica quanto de será está quantidade de cores c</description>

</property>

<property>

<name>yarn.scheduler.maximum-allocation-vcores</name>

<value>C</value>

<description>Diz ao scheduler do YARN quantos cores de processamento máximo C ele deve alocar para executar as tarefas do Hadoop. Pedimos maiores que este são recusados pelo YARN. O usuário indica quanto de será está quantidade de cores C, mas o correto é que deva este valor seja igual a soma de cores disponibilizada no cluster.</description>

</property>

Hadoop

slaves

Deve conter somente uma linha com o seguinte conteúdo:

nome_maquina

Hadoop

Feitas estas configurações conforme as necessidades do usuário, então procedemos para a inicialização dos serviços do HDFS e do YARN. Os passos a seguir demonstram como isso deve ser feito.

Navegar a até a pasta onde estão os binários do Hadoop. E depois ir até a pasta “bin”.

./hdfs namenode -format

e em ../sbin/

./start-dfs.sh

./start-yarn.sh

Agora é só executar as aplicações do Hadoop que elas devem funcionar sem problemas.

Hadoop

Para parar os serviços do HDFS e do YARN, navegue até a pasta onde estão os binários do Hadoop e depois vá até a pasta “sbin”. Então execute:

./stop-dfs.sh

./stop-yarn.sh

Deve ser feita a limpeza da pasta utilizada pelo HDFS.

rm -rf /home/<usuário>/tmp/*

Hadoop

Uma vez configurado em modo local, podemos configurar o Hadoop para ser utilizado **em modo cluster.** A seguir são mostrados as alterações necessárias a serem feitas nos arquivos.

A configuração feita a seguir deve ser feita em todas as máquinas que venham a fazer parte do cluster.

Hadoop

As chaves geradas para o ssh devem ser copiadas para todas as máquinas presentes no cluster, ou seja, o arquivo “/.ssh/authorized_keys” deve conter todas as chaves geradas em todos as máquinas do cluster.

Navegar a até a pasta onde estão os binários do Hadoop. E depois ir até a pasta “etc/hadoop/”.

Hadoop

Editar os seguintes arquivos:

core-site.xml, hdfs-site.xml, yarn-site.xml, slaves.

Abrir o arquivo mapred-site.xml.template e salvá-lo como mapred-site.xml.

Adicionar as configurações sempre entre as tags de <configuration></configuration>:

Hadoop

core-site.xml

<property>
<name>hadoop.tmp.dir</name>
<value>/home/<usuário>/tmp</value>
</property>

<property>
<name>fs.defaultFS</name>
<value>hdfs://<master>:54310</value>
</property>

Para o caso de Single Node o **fs.defaultFS**
<value>hdfs://localhost:9000</value>

Hadoop

hdfs-site.xml

Aqui é colocado o número de réplicas a serem distribuídas entre os nós. Default = 3. Se for só uma máquina(Single Node) =1

```
<configuration>
    <property>
    <name>dfs.replication</name>
    <value>3</value>
    </property>
</configuration>
```

Hadoop

Abrir o arquivo mapred-site.xml.template e salvá-lo como mapred-site.xml

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

<description>Definição do scheduler a ser utilizado pelo

Hadoop</description>

</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value><master>:8025</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado para cluster vai o “master” do YARN</description>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value><master>:8035</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado para cluster vai o “master” do YARN</description>

</property>

Hadoop

yarn-site.xml

<property>

<name>yarn.resourcemanager.address</name>

<value><master>:8050</value>

<description>máquina e porta utilizada pelo YARN. Como está sendo configurado para cluster vai o “master” do YARN</description>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce_shuffle</value>

</property>

Hadoop

yarn-site.xml

```
<property>
<name>yarn.nodemanager.resource.cpu-vcores</name>
<value>N</value>
</property>

<property>
<name>yarn.nodemanager.resource.memory-mb</name>
<value>M</value>
</property>
```

Hadoop

yarn-site.xml

```
<property>
<name>yarn.scheduler.minimum-allocation-mb</name>
<value>m</value>
</property>

<property>
<name>yarn.scheduler.maximum-allocation-mb</name>
<value>M</value>
</property>
```

Hadoop

yarn-site.xml

```
<property>
<name>yarn.scheduler.minimum-allocation-vcores</name>
<value>c</value>
</property>

<property>
<name>yarn.scheduler.maximum-allocation-vcores</name>
<value>C</value>
</property>
```

Hadoop

Arquivo slaves - no nosso exemplo

Deve conter em cada linha o nome de cada máquina utilizada como worker no cluster, normalmente não é colocado o nome do master do HDFS e do YARN:

clt-01-deb.local

clt-02-deb.local

No arquivo **"/etc/hosts"**, deve ser colocado o ip e nome de cada máquina pertencente ao cluster.

```
127.0.0.1    localhost
127.0.1.1    clt-02-deb

# Cluster Demo
192.168.10.1    linux-srv.local
192.168.10.2    clt-01-deb.local
192.168.10.3    clt-02-deb.local
```

Hadoop

Feitas estas configurações conforme as necessidades do usuário, então procedemos para a inicialização dos serviços do HDFS e do YARN. Os passos a seguir demonstram como isso deve ser feito.

Navegar a até a pasta onde estão os binários do Hadoop. E depois ir até a pasta “bin”.

cd bin/
./hdfs namenode -format

cd ../sbin/

./start-dfs.sh

./start-yarn.sh

Agora é só executar as aplicações do Hadoop que elas devem funcionar sem problemas.

Hadoop

Para parar os serviços do HDFS e do YARN, navegue a até a pasta onde estão os binários do Hadoop. E depois ir até a pasta “sbin”. Então execute:

./stop-dfs.sh

./stop-yarn.sh

Hadoop

Deve ser feita a limpeza da pasta utilizada pelo HDFS em cada máquina que participa do cluster. Se não for apagada, isto pode fazer com que ocorram erros ao executar aplicações Hadoop.

rm -rf /home/<usuário>/tmp/*

Hadoop

Como compilar:

Adicionar as seguintes linhas ao arquivo .bashrc:

```
export PATH=$JAVA_HOME/bin:$PATH

export
HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar
```

Hadoop

Como compilar:

Seguir Tutorial MapReduce na página do Apache

Copiar o código do wordcount da página a seguir para um arquivo chamado WordCount.java

https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html

Hadoop

Como compilar:

Executar os seguintes comandos:

./hadoop com.sun.tools.javac.Main WordCount.java jar cf wc.jar WordCount*.class

Pronto, agora é somente executar.

Hadoop

Executar o Exemplo wordcount.

Primeiro criar a pasta input dentro do hdfs.

./hdfs dfs -mkdir /input

Hadoop

Executar o Exemplo wordcount.

Colocar alguns arquivos dentro da pasta input.

./hdfs dfs -put <arquivos de entrada para o wordcount > /input

Hadoop

Executar o Exemplo wordcount.

Para executar, utilize o seguinte comando:

./hadoop jar wc.jar WordCount /input /output

Hadoop

Executar o Exemplo wordcount.

Para visualizar o resultado, utilize o seguinte comando:

./hdfs dfs -cat /output/part-r-00000