

Prácticas BigData

1. YARN EN ENTORNOS PSEUDO-DISTRIBUIDO

- Copiamos el fichero “mapred-site.xml.template” a “mapred-site.xml”
- Ponemos la siguiente propiedad.

```
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>
```

- Y en el fichero yarn-site.xml ponemos las siguientes:

```
<property>
  <name>yarn.resourcemanager.hostname</name>
  <value>nodo1</value>
</property>
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
<property>
  <name>yarn.nodemanager.aux-services.mapreduce_shuffle.class</name>
  <value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
```

- Arrancamos HDFS si no lo está
- Arrancamos los servicios YARN

```
start-yarn.sh
```

- Comprobamos con jps que tenemos los dos procesos de YARN
- Arrancamos HDFS si no lo está
- Arrancamos los servicios YARN
- Start-yarn.sh
- Arrancamos el servicio que permite guardar el histórico de los Jobs lanzados
- En el caso de tener hadoop versión 2 ejecutamos el comando

```
/mr-jobhistory-daemon.sh starthistoryserver
```

- En el caso de tener haddop 3 ejecutamos el comando

mapred --daemon start historyserver

- Comprobamos con “jps” que tenemos los procesos lanzados
- Arrancamos HDFS si no lo está
- Arrancamos los servicios YARN

Start-yarn.sh

- Comprobamos con jps que está funcionando

```
jps
1137 Jps
22962 DataNode
23187 SecondaryNameNode
419 ResourceManager
22837 NameNode
541 NodeManager
```

- Arrancamos el servicio que permite guardar el histórico de los Jobs lanzados
 - En versión 2 de Hadoop


mr-jobhistory-daemon.sh start historyserver

starting historyserver, logging to /opt/hadoop/logs/mapred-hadoop-historyserver-nodo1.out

- En Versión 3 de Hadoop
- Comprobamos con jps que está funcionando

```
jps
22962 DataNode
23187 SecondaryNameNode
419 ResourceManager
1475 JobHistoryServer
22837 NameNode
1515 Jps
541 NodeManager
```

- Accedemos a la página WEB de Administración para comprobar que está funcionando.



All Applications

Cluster

- About
- Nodes
- Node Labels
- Applications
- NEW
- NEW SAVING
- SUBMITTED
- ACCEPTED
- RUNNING
- FINISHED
- FAILED
- KILLED
- Scheduler

Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used
0	0	0	0	0	0 B

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes
1	0	0	0

Scheduler Metrics


Scheduler Type	Scheduling Resource Type	Minimum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Running Containers
No data available in table										

Showing 0 to 0 of 0 entries

- Comprobamos en Nodes que tenemos el nodo activo



Nodes of the cluster

Cluster

- About
- Nodes
- Node Labels
- Applications
- NEW
- NEW SAVING
- SUBMITTED
- ACCEPTED
- RUNNING
- FINISHED
- FAILED
- KILLED
- Scheduler

Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used
0	0	0	0	0	0 B	8 GB	0 B	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted
1	0	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0

Show 20 entries

Node Labels	Rack	Node State	Node Address	Node HTTP Address	Last health-update	Health-report	Containers	Mem Used	Mem Avail
/default-rack	RUNNING	nodo1:42534	nodo1:8042	sáb ene 06 22:11:26 +0100 2018	0	0 B	8 GB		