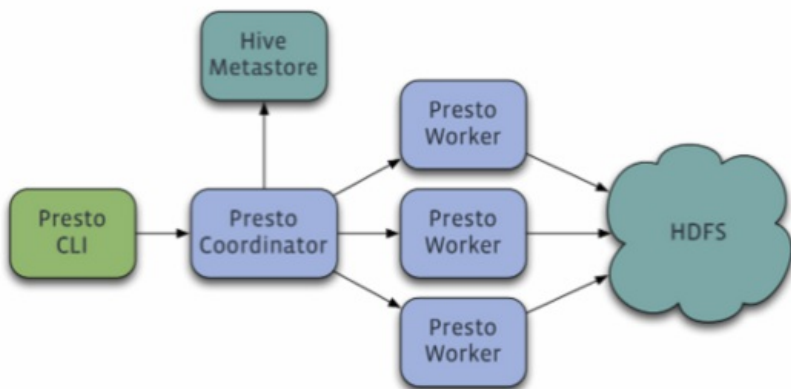


presto集群搭建

目录

- 架构
- 集群规划
- 基础环境
- 安装步骤
- 运行presto集群
- 测试
- 参考文档

1. 架构



Presto查询引擎是一个**Master-Slave**的架构，由下面三部分组成：

- 一个Coordinator节点
- 一个Discovery Server节点
- 多个Worker节点

Coordinator: 负责解析SQL语句，生成执行计划，分发执行任务给Worker节点执行

Discovery Server: 通常内嵌于Coordinator节点中

Worker节点: 负责实际执行查询任务,负责与HDFS交互读取数据

2. 集群规划

- 192.168.101.53 (coordinator)
- 192.168.101.54 (worker)
- 192.168.101.55 (worker)

3. 基础环境

- **OS:** CentOS Linux release 7.3.1611
- **Cpu:** 4C Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz
- **Mem:** 16G
- **mysql version:** mysql57-community-release-el7-11.noarch
- **服务器IP:** 192.168.101.53(**coordinator**), 192.168.101.54/55(**worker**)

4. 安装步骤

4.1. 下载 presto-server-0.161

下载地址: <https://repo1.maven.org/maven2/com/facebook/presto/presto-server/0.161/presto-server-0.161.tar.gz>

4.2. 修改配置文件

在安装目录下创建etc目录，在etc下创建一下文件：

- config.properties: Presto 服务配置 - node.properties : 环境变量配置，每个节点特定配置
- jvm.config : Java虚拟机的命令行选项
- log.properties: 允许你根据不同的日志结构设置不同的日志级别
- catalog目录 : 每个连接者配置 (data sources)

4.2.1. config.properties

- 在**coordinator**节点下进行以下配置：
coordinator=true
node-scheduler.include-coordinator=false
http-server.http.port=18080
query.max-memory=16GB
query.max-memory-per-node=1GB
discovery-server.enabled=true
discovery.uri=http://192.168.101.53:18080
- 在**worker**节点下进行以下配置：
coordinator=false
http-server.http.port=18080
query.max-memory=16GB
query.max-memory-per-node=1GB
discovery.uri=http://192.168.101.53:18080

4.2.2. node.properties

- 在**coordinator**节点下进行以下配置： node.environment=prestoclustert
node.id=cdhsd-presto-test-coordinator
node.data-dir=/data/presto/data
- 在**worker**节点下进行以下配置： node.environment=prestoclustert
node.id=cdhsd-presto-test-worker-54
node.data-dir=/data/presto/data

4.2.3. jvm.config

```
-server  
-Xmx16G  
-XX:+UseConcMarkSweepGC  
-XX:+ExplicitGCInvokesConcurrent  
-XX:+CMSClassUnloadingEnabled  
-XX:+AggressiveOpts  
-XX:+HeapDumpOnOutOfMemoryError  
-XX:OnOutOfMemoryError=kill -9 %p  
-XX:ReservedCodeCacheSize=150M
```

4.2.4. log.properties

```
com.facebook.presto=INFO
```

4.2.5. catalog目录

在catalog下进行以下配置：

- hive.properties文件
connector.name=hive-cdh5
hive.metastore.uri=thrift://192.168.101.62:9083
hive.config.resources=/etc/hive/conf.cloudera.hive/core-site.xml, /etc/hive/conf.cloudera.hive/hdfs-site.xml

hive.allow-drop-table=true

- jmx.properties文件:
connector.name=jmx

5. 运行presto集群

首先启动coordinator节点，然后再分别启动worker节点。

先后在coordinator，worker的bin目录下执行：

```
./launcher start
```

关闭服务：

```
./launcher stop
```

6. 测试

- 下载 presto-cli-0.161-executable.jar：Presto CLI为用户提供了一个用于查询的可交互终端窗口。CLI是一个可执行JAR文件，这也就意味着你可以像UNIX终端窗口一样来使用CLI，下载地址（<https://repo1.maven.org/maven2/com/facebook/presto/presto-cli/0.161/presto-cli-0.161-executable.jar>）

文件下载后，重命名为 presto，使用 chmod +x 命令设置可执行权限：

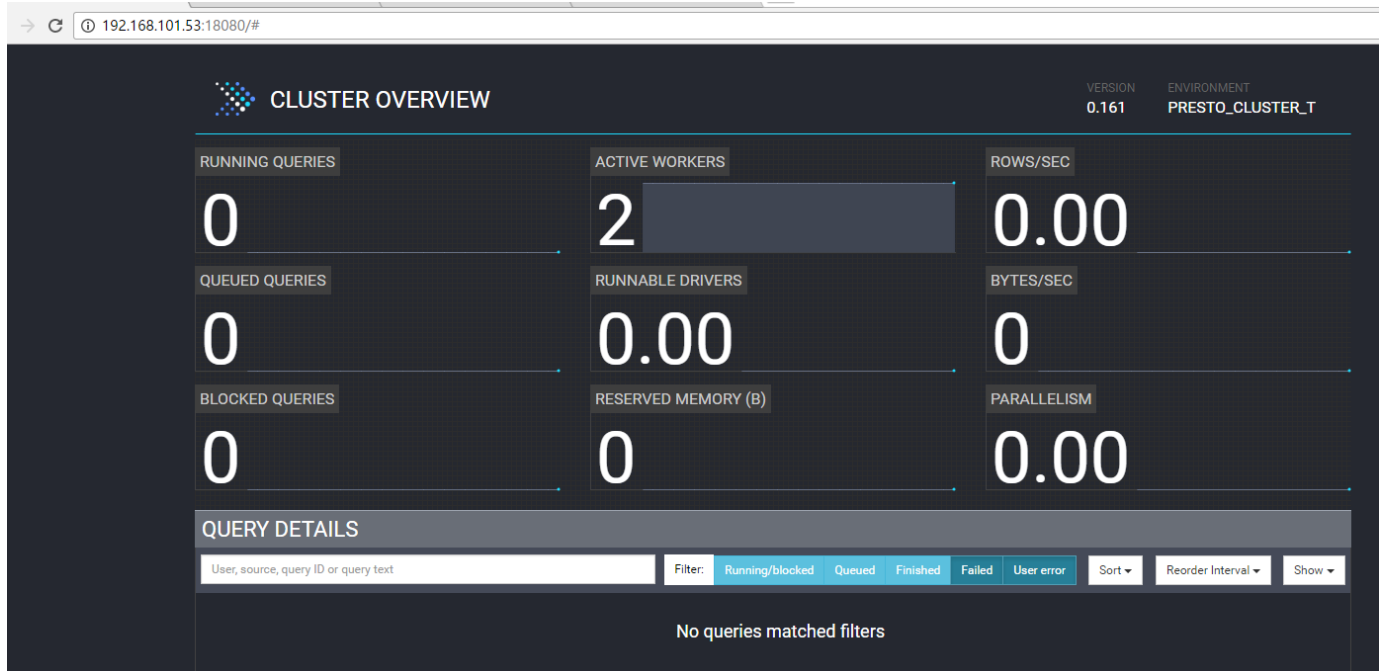
```
cp -f presto-cli-0.161-executable.jar presto
```

```
chmod +x presto 执行：./presto --server 192.168.101.53:18080 --catalog hive --schema default
```

```
[root@localhost presto]# ./presto --server 192.168.101.53:18080 --catalog hive --schema default
presto:default> show tables;
  Table
-----
customers
sample_07
sample_08
web_logs
(4 rows)

Query 20170720_095714_00005_ujfwu, FINISHED, 2 nodes
Splits: 2 total, 2 done (100.00%)
0:00 [4 rows, 103B] [20 rows/s, 528B/s]
```

- 也可通过页面访问：



7. 参考文档

- <http://prestodb-china.com/>
- <https://prestodb.io>

- <http://blog.csdn.net/jiangshouzhuang/article/details/52461274>
- <http://www.cnblogs.com/tgzhu/p/6038123.html>