吱吱吱 (piperck) XD

Be more cautious.
github.com/piperck weibo.com/pieprck

利用 Docker 搭建单机的 Cloudera CDH 以及使用实践

想用 CDH 大礼包，于是先在 Mac 上和 Centos7.4 上分别搞个了单机的测试用。其实操作的流和使用到的命令差不多就一并说了：

首先前往官方下载包：

https://www.cloudera.com/downloads/quickstart_vms/5-13.html

如果使用 mac 并且安装 docker。 我们可以很轻松的使用 kitematic 来获取最新版本的 cloudera docker 镜像。只需要搜 cloudera/quickstar 即可这是地址：

https://hub.docker.com/r/cloudera/quickstart/

当我们下载好镜像之后就可以愉快的将进行加载起来。macos 基本是全程无脑，linux 稍微麻烦一点使用

docker import cloudera-quickstart-vm-5.13.0-0-beta-docker.tar

将镜像 import 进来。

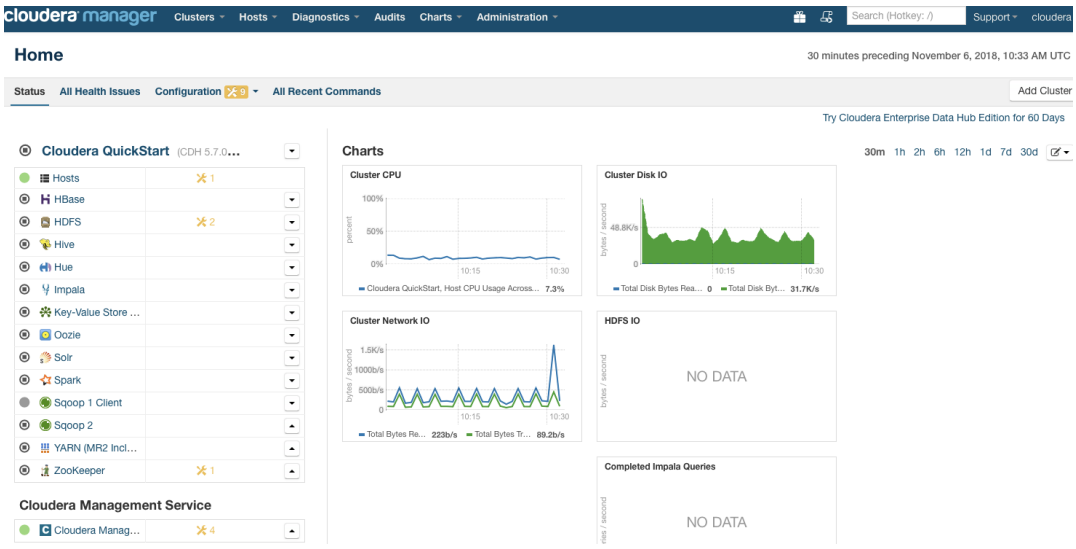
然后使用命令启动就可以了。

Cloudera 的 docker 版本分成两部分启动。一方面是大礼包的启动 /usr/bin/docker-quickstart，一方面是 Cloudera manager 本身的启动 /home/cloudera/cloudera-manager

这里我们使用命令

```
docker run --name cdh --hostname=quickstart.cloudera --privileged=true -t -i -p 8020:8020 -p 8022:8022 -p 7180:7180 -p 21050:21050 -p 50070:50070 -p 50075:50075 -p 50010:50010 -p 50020:50020 -p 8890:8890 -p 60010:60010 -p 10002:10002 -p 25010:25010 -p 25020:25020 -p 18088:18088 -p 8088:8088 -p 19888:19888 -p 7187:7187 -p 11000:11000 cloudera/quickstart /bin/bash -c '/usr/bin/docker-quickstart && /home/cloudera/cloudera-manager --express && service ntpd start'
```

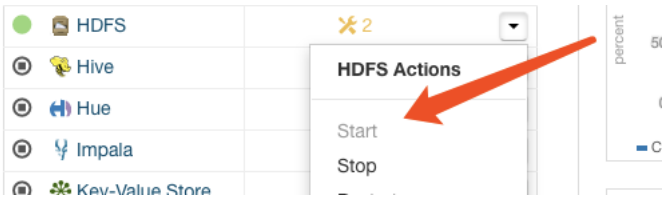
直接启动两个程序。这里注意参数都可以从下面 refrence 查询到大概是什么意思，合理之所以要写这么多端口映射也是为了方便我们外面的机器可以方面的访问 docker 内部的这些端口，访问这些服务。 Cloudera 本身的 manager 是 7180 端口。当这些启动起来之后就可以访问目标机器 ip 的 7180 端口访问 Cloudera manager 了。



上图就是一个 dashbord 的样子。另外在 linux 机器有一个地方需要注意的是，可能你的 docker 用上面命令起起来之后，docker 内的实例没有办法访问外网，这里我们配置一下 docker 创建容器时候的参数增加 -net host 即可。也可以在宿主机器上在 /etc/default/docker 文件。并且配置上 DOCKER_OPTS="--dns host_ip" 即可。

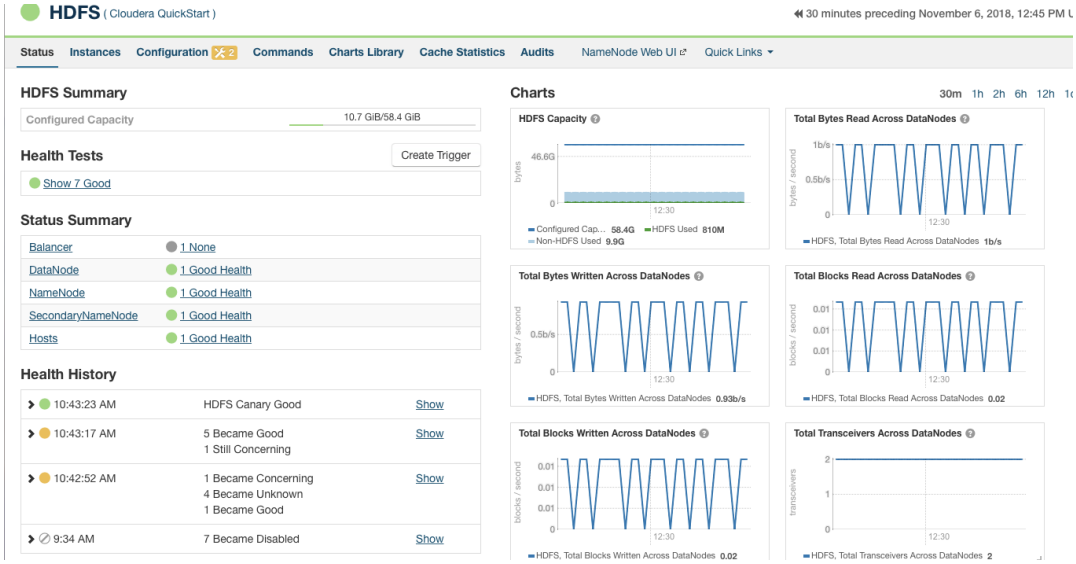
从上图我们还可以注意到另外一个问题，除了主机和 manager 都没有启动。在 Cloudera 大礼包中，只有 hue 和 manager 本身是什么服务都不依赖的可以在任何时候选择启动和关闭。其他的应用多多少少存在着一些启动顺序上的依赖这个要注意。

现在我们来启动几个我们关心的服务，我们先来启动 HDFS。



这里我已经把它启动起来了当没有启动的时候点击 start cm(Cloudera manager) 就会把这个给启动起来。

点一下已经启动起来的 HDFS 就会到这个应用的 dashborad cm 给我们提供了非常多的图表以及面板可以关注目前机器和集群的情况如下图：



目前看到的都是单节点的情况。让我意外的是启动的时候竟然还会有 Canary 模式。在这个界面点击右上角的 NameNode Web UI 就可以看到老板我们熟悉的

社区版的 HDFS 界面了。比较方便的是当我们点击 Configuration 就可以进到 HDFS 的一些配置包括块大小之类的配置这里都可以方便设置。

可以看到这一套东西真的是把能包好的东西都已经给我们列出来了。

随笔档案

我暂时在单机上面启了两个 app 一个 HDFS 一个 Spark ，内存基本被打到了 5个G. 可以看出来其实 CDH 大礼包其实还是非常吃内存的。当我们在进行线上环境配置的

时候占用的资源肯定是只增不减。这里抛砖引玉了一个 app 接下来大家可以按照这个方法继续探索。

既然 HDFS 已经启动让我们来尝试使用 python 来操作一下 HDFS

pip install hdfs 安装 hdfscli 包

```
from hdfs.client import Client
client = Client("http://127.0.0.1:50070", root="/", timeout=100)

print(client.list("/"))
client.upload("/", "/Users/piperck/Desktop/About_me/dragen.wma")
```

可以看到我们可以直接创建连接，client.list 是列出 HDFS 目前根目录的情况。 下面我们调用 client.upload 上传文件。

上传文件的时候可能遇到很多问题，因为我们这里使用的是 docker 搭建的 CDH ,所以一般会报这个错误：

```
('<requests.packages.urllib3.connection.HTTPConnection object at 0x00000000043A3FD0>: Failed to establish a new
connection:
 [Errno 11004] getaddrinfo failed',))
```

这个时候我们需要去 docker 里面 hostname 一下会得到 quickstart.cloudera。我用的 macos 所以把这个直接配置进我电脑的 /etc/hosts 里。

127.0.0.1 quickstart.cloudera

否则，永远报错。。这里搞了非常久需要注意一下。

之后继续尝试连接，应该还会报另外一个错误：

```
Permission denied: user=root, access=WRITE, inode=&amp;quot;/&amp;quot;:hdfs:supergroup:drwxr-xr-x
```

很明显权限问题，因为我们并没有登陆而且在本地使用的权限也不明。755 的权限导致我们无法上传文件，这里的 root 权限是 hdfs 用户，所以会失败

这里有两个办法可以解决这个问题：

1. 调整 hdfs 的权限检查将

```
<property>
  <name>dfs.permissions</name>
  <value>false</value>
</property>
```

设置为 False 关闭权限检查。

2. 增加一个由这个用户创建的文件夹在根目录，然后将文件往那里面传就可以了。

现在我们再将传上去的文件下载回来：

```
from hdfs.client import Client
client = Client("http://127.0.0.1:50070", root="/", timeout=100)

print(client.list("/"))
client.download("/dragen.wma", "/Users/piperck/Desktop")
```

很轻松成功了，没有再出什么幺蛾子。

Reference:

https://www.cloudera.com/documentation/enterprise/5-15-x/topics/quickstart_docker_container.html ---docker 安装启动文档

https://www.cloudera.com/documentation/enterprise/5-15-x/topics/cm_mc_start_stop_service.html#cmug_topic_5_6 ---启动 hdfs 服务教程

https://blog.csdn.net/g11d111/article/details/72902112

https://dxysun.com/2018/07/19/hadoopForPythonHdfs/ PYTHONHDFS 使用教程

https://blog.csdn.net/Gamer_gyt/article/details/52446757 使用python的hdfs包操作分布式文件系统（HDFS）

https://segmentfault.com/a/1190000002672666 hadoop 常用文件的操作命令

□

分类： Devops，Linux or 工具功能，Spark，BigData

piperck
关注 - 7
粉丝 - 54
+加关注

0 0

« 上一篇：【纪录】Hash about

» 下一篇：CDH 6.0.1 集群搭建 「Before install」

posted @ 2018-11-06 18:38 piperck 阅读(1557) 评论(2) 编辑 收藏

评论列表

#1楼 2018-11-06 19:12 东方翔

恭喜 x 总喜提 CDH
话说有木有耍过 k8s ？

支持(0) 反对(0)

#2楼[楼主] 2018-11-06 19:26 piperck

@ 东方翔
我估计快了。。。。不过这个文章其实还没写完。。估计后面要出个全家桶

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

2015年4月 (9)

文章分类

django

python

前端杂谈

最新评论

1. Re:Golang的聊天服
广播）（一）

期待下一篇啊，大神。

2. Re:【MySQL 读书笔
行该查询语句的时候我
@东方翔又被大佬视奸

3. Re:【MySQL 读书笔
如何执行的

大佬大佬

4. Re:Python魔法方法
d)细解几个常用魔法方

优秀

5. Re:python 类和元类
理解和简单运用

@gantao(i for i in ra
是一个生成器；[i for i
这样写就是一个数组了
器的效果了。...

阅读排行榜

1. pip和conda到底有
10)

2. nginx服务器配置／
x 配置笔记(36718)

3. linux命令重定向>、
>、 1>>、 2>>、 <

4. 排查mysql innodb
ut exceeded; try res
on的问题(23432)

5. 使用ssh config配置
接(20602)

评论排行榜

1. 排查 Maxwell can
e 并且使用 MySQL bir
题(9)

2. python 类和元类(m
和简单运用(7)

3. nginx服务器配置／
x 配置笔记(4)

4. websocket协议握手

5. python 字符串，数
巩固。(3)

推荐排行榜

1. 关于封装了gevent的
st库的使用与讨论(4)

2. linux命令重定向>、
>、 1>>、 2>>、 <

3. python 类和元类(m
和简单运用(4)

4. Python魔法方法(m
解几个常用魔法方法（_

5. Golang的时间生成，
取函数执行时间的方法(