作业 1

钟诚 16307110259

March 6, 2020

1. 分析对比 HDFS 与传统的分布式/网络文件系统

- 从架构上，传统的分布式/网络文件系统，如 linux 的 NFS(Network File system，网络文件系统协议)，目的是是通过挂载另一台机器上的某个分区到自己机器上的分区从而可以访问对应的文件。HDFS 是分布在许多联网计算机或节点上的文件系统，架构是通过网络和机器节点把多个机器上的文件统一成一个文件系统。
- 在文件存储上，对于一个文件，在 NFS 上必然完整存储在一个节点的一个硬盘上，而在 HDFS 中，一个文件可能会被切割成多个小文件，存储在不同的机器上。
- 在容错能力上，HDFS 会在文件系统上储存文件的多个副本，默认数量为 3，而 NFS 没有容错功能。
- 在访问能力上，HDFS 支持文件的多个副本，从而可以缓解多个客户端访问单个文件时的速度问题，同时因副本存储在不同的物理磁盘上，可扩展性比 NFS 更强。
- 在并发控制上，分布式文件系统需要进行复杂的交互，而 HDFS 采用"Write-Once-Read-Many" 策略，即任何时间都只允许一个客户端进行写操作。
- 在资源移动策略上，HDFS 支持 "移动计算到数据"，即对某个数据块的操作必定在存储这个数据块的节点上完成的，同时存储任一副本的节点都可以执行 map 操作，由 Jobtracker 进行任务的分配。
- 综上所述，相较而言 HDFS 更适合进行海量数据的高吞吐量、批量处理任务，运行在廉价商用机器集群上，而不适合进行对文件的多用户写入随机读写，低延时访问、大量小文件的存储及处理等任务。

注：引自 CNBLOGS、Stackoverflow 相关博客与问答

---

2. 进行 HDFS 伪分布实际部署，报告实验过程

注：步骤引自厦门大学林子雨老师相关博客《大数据技术原理与应用第二章大数据处理架构 Hadoop 学习指南》
环境：虚拟机安装 64 位 Ubuntu LTS 14.04，采用伪分布式安装方式

2.1 伪分布式安装 Hadoop

先在终端中通过如下语句创建具有 root 权限的新用户

```
sudo useradd -m hadoop -s /bin/bash
sudo adduser hadoop sudo
```

2.1.1 JDK 安装

下载 JDK1.8 的安装包 jdk-8u162-linux-x64.tar.gz，随后在安装目录下执行如下命令：

```
sudo tar -zxvf ./jdk-8u162-linux-x64.tar.gz -C /usr/lib/jvm
```

在命令行中对环境变量配置文件.bashrc 进行修改

```
export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_162
export JRE_HOME=${JAVA_HOME}
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

最后通过如下命令使环境变量生效并检验设置是否正确

```
source  /.bashrc
java -version
```

如图，显示 JAVA 安装成功

```
hadoop@dblab-VirtualBox:~/下载$ java -version
java version "1.8.0_162"
Java(TM) SE Runtime Environment (build 1.8.0_162-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.162-b12, mixed mode)
```

Figure 1: 安装完成

2.1.2 Hadoop3 安装

在 Hadoop3 的安装目录下打开 shell，通过如下命令将其解压到指令目录下

```
sudo tar -zxf  /下载/hadoop-3.2.1.tar.gz -C /usr/local
cd /usr/local/
sudo mv ./hadoop-3.2.1/ ./hadoop
sudo chown -R hadoop ./hadoop
```

随后通过如下命令检测 Hadoop 是否成功安装

```
cd /usr/local/hadoop
./bin/hadoop version
```

显示如图，表示安装完成

```
hadoop@dblab-VirtualBox:~$ cd /usr/local/hadoop
hadoop@dblab-VirtualBox:/usr/local/hadoop$ ./bin/hadoop version
Hadoop 3.2.1
Source code repository https://gitbox.apache.org/repos/asf/hado
op.git -r b3cbbb467e22ea829b3808f4b7b01d07e0bf3842
Compiled by rohithsharmaks on 2019-09-10T15:56Z
Compiled with protoc 2.5.0
From source with checksum 776eaf9eee9c0ffc370bcbc1888737
This command was run using /usr/local/hadoop/share/hadoop/commo
n/hadoop-common-3.2.1.jar
```

Figure 2: Hadoop 安装完成

2.1.3 Hadoop3 伪分布式配置

通过 gedit 修改位于 /usr/local/hadoop/etc/hadoop/ 的配置文件 core-site.xml 和 hdfs-site.xml，修改配置文件 core-site.xml 和 hdfs-site.xml，将其中的

```
<configuration>
</configuration>
```

分别改为

```
<configuration>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>file:/usr/local/hadoop/tmp</value>
        <description>Abase for other temporary directories.
 </description>
    </property>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://localhost:9000</value>
    </property>
</configuration>
```

和

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
 </description>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/name</value>
        <name>dfs.namenode.data.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/data</value>
    </property>
</configuration>
```

随后执行 NameNode 格式化，

```
cd /usr/local/hadoop
./bin/hdfs namenode -format
```

显示如图所示“successfully formatted”界面，表示 NameNode 格式化成功

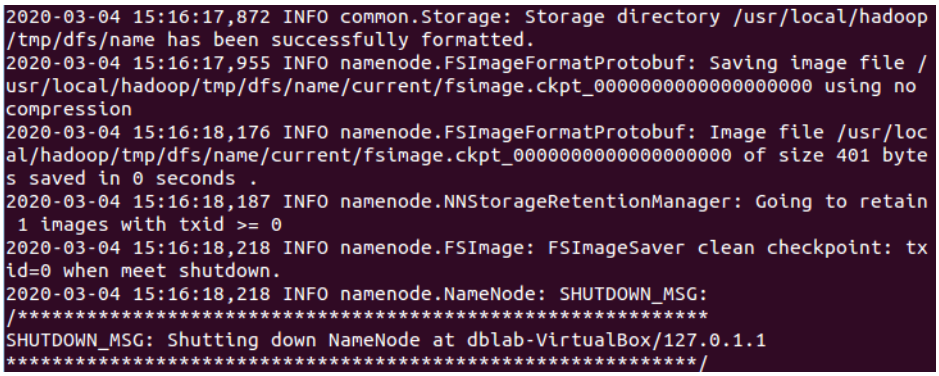

Figure 3: NameNode 格式化完成

在配置时，我们发现如果没有进行 ssh 的无密码设置，会出现 permission denied 的错误，故在该用户上配置无密码设置的 ssh。首先安装 SSH server 并登陆本机

```
sudo apt-get install openssh-server
ssh localhost
```

首次登陆时会出现提示，输入 yes 即可，然后退出 ssh, 利用 ssh-keygen 生成密钥并加入授权，使其可以无密码登录，指令如下，如有提示输入回车即可

```
exit
cd  /.ssh/
ssh-keygen -t rsa
cat ./id_rsa.pub » ./authorized_keys
```

配置完 ssh 后，可以在 hadoop 用户中开启 Namenode 和 Datanode

```
cd /usr/local/hadoop
./sbin/start-dfs.sh
```

开启后，可以通过 jps 命令检验设置情况，出现如图进程，则配置成功

```
hadoop@dblab-VirtualBox:/usr/local/hadoop$ jps
28002 SecondaryNameNode
28164 Jps
27780 DataNode
27623 NameNode
```

Figure 4: jps 指令结果

成功启动后，可以在 http://localhost:9870 中查看 NameNode、DataNode 相关信息和 HDFS 相关文件，如图所示：

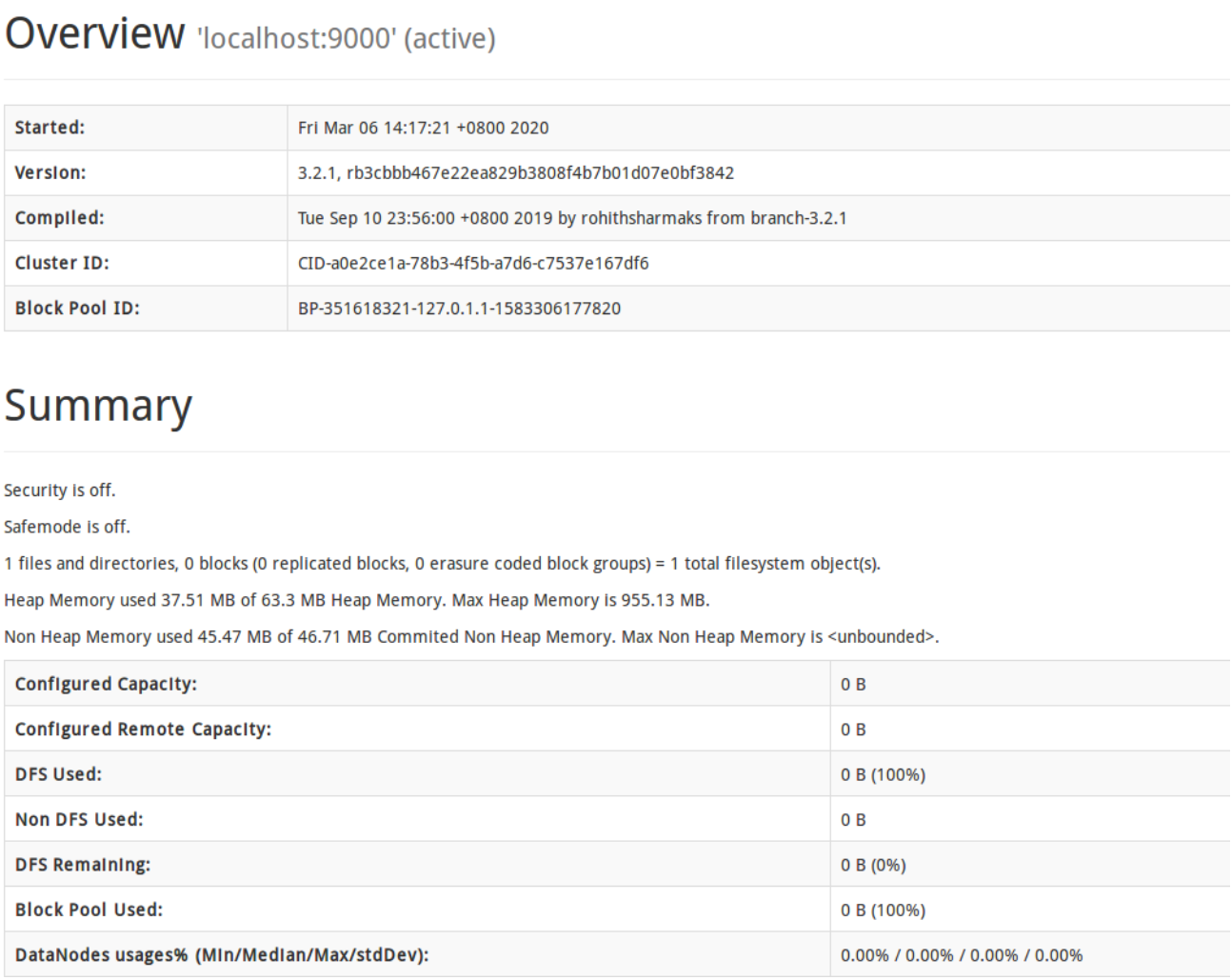

Figure 5: 网页查看 NameNode 信息

2.2 运行 Hadoop 伪分布式实例

在本例中，我们选择 Hadoop 附带的 grep 例子作为测试的实例。值得注意的是，在伪分布式模式下读取的是 HDFS 上的数据，故需要我们在 HDFS 中创建用户目录并将待输入的 xml 文件复制到文件系统中。我们运用如下命令，在 HDFS 中创建用户目录，将./etc/hadoop 中的 xml 文件作为输入复制到分布式文件系统中，并查看文件列表

```
./bin/hdfs dfs -mkdir -p /user/hadoop
./bin/hdfs dfs -mkdir input
./bin/hdfs dfs -put ./etc/hadoop/*.xml input
./bin/hdfs dfs -ls input
```

input 文件显示如下:

```
hadoop@dblab-VirtualBox:/usr/local/hadoop$ ./bin/hdfs dfs -ls input
Found 9 items
-rw-r--r--   1 hadoop supergroup       8260 2020-03-04 11:22 input/capacity-scheduler.xml
-rw-r--r--   1 hadoop supergroup       1071 2020-03-04 11:22 input/core-site.xml
-rw-r--r--   1 hadoop supergroup      11392 2020-03-04 11:22 input/hadoop-policy.xml
-rw-r--r--   1 hadoop supergroup       1129 2020-03-04 11:22 input/hdfs-site.xml
-rw-r--r--   1 hadoop supergroup        620 2020-03-04 11:22 input/httpfs-site.xml
-rw-r--r--   1 hadoop supergroup       3518 2020-03-04 11:22 input/kms-acls.xml
-rw-r--r--   1 hadoop supergroup        682 2020-03-04 11:22 input/kms-site.xml
-rw-r--r--   1 hadoop supergroup        758 2020-03-04 11:22 input/mapred-site.xml
-rw-r--r--   1 hadoop supergroup        690 2020-03-04 11:22 input/yarn-site.xml
```

Figure 6: Input 结果

随后我们执行这样一个实例，通过 grep 提取输入中符合'dfs[a-z.]+' 这一正则表达式的结果

```
./bin/hadoop jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-*.jar grep input output 'dfs[a-z.]+'
./bin/hdfs dfs -cat output/*
```

再查看结果，发现程序可以正常运行，至此，HDFS 配置完成。

```
2020-03-04 11:38:32,185 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHo
stTrusted = false
1       dfsadmin
1       dfs.replication
1       dfs.namenode.name.dir
1       dfs.datanode.data.dir
```

Figure 7: Output 结果

最后，我们关闭 Hadoop

```
./sbin/stop−dfs.sh
```

```
hadoop@dblab-VirtualBox:/usr/local/hadoop$ ./sbin/stop-dfs.sh
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [dblab-VirtualBox]
```

Figure 8: 关闭 Hadoop