華中科技大學

大数据处理实验报告

实验三：MapReduce 的基本操作

姓　　名:

学　　号:

院　　系:

专　　业:

年　　级:

指导教师:

2021 年　月　日

一：实验目的

1、了解 MapReduce 的用途

2、掌握 MapReduce 的基本命令

二：实验要求

1、第四节中的实验内容要附上完整的实验过程截图以及必要的文字说明，每个人的 IP 地址等不同，不能直接套用样例的截图。

2、请同学们在完成报告后，将报告的 pdf 版本命名为：大数据实验三+姓名+学号.pdf，并在这周五前,发到邮箱:

aaaaltaaaa@126.com(石老师班)

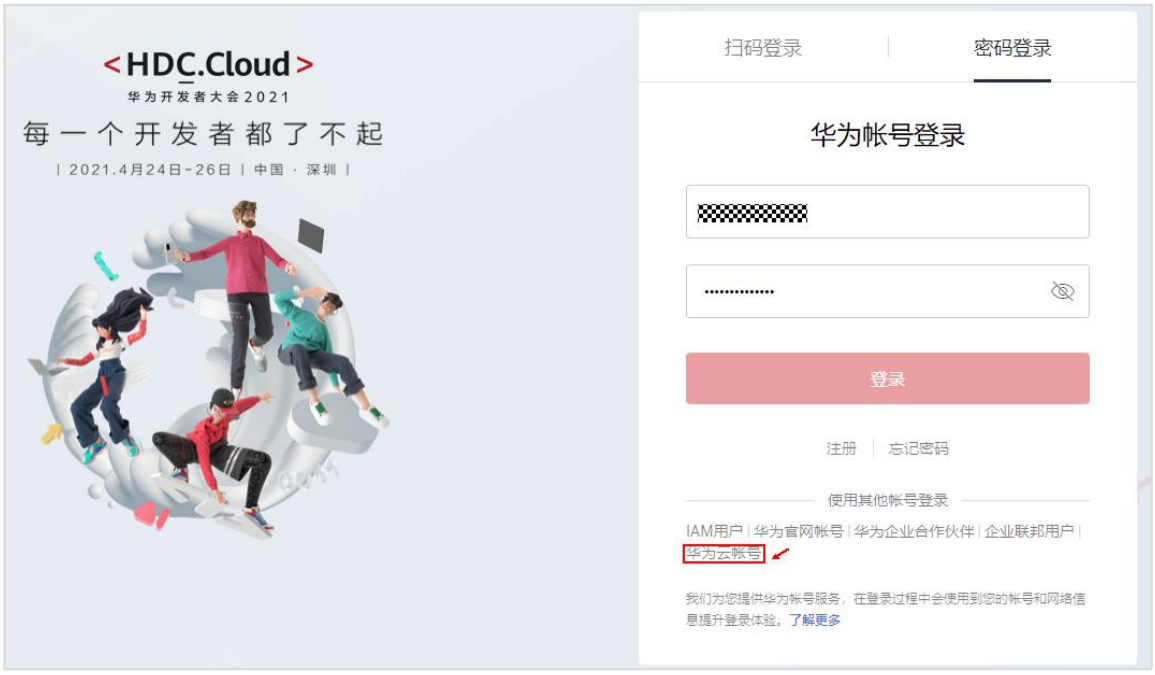
798792873@qq.com(郑老师班)

三：实验环境配置

步骤 1 登录华为云网站

https://www.huaweicloud.com

点击右上角登录，输入账号和密码

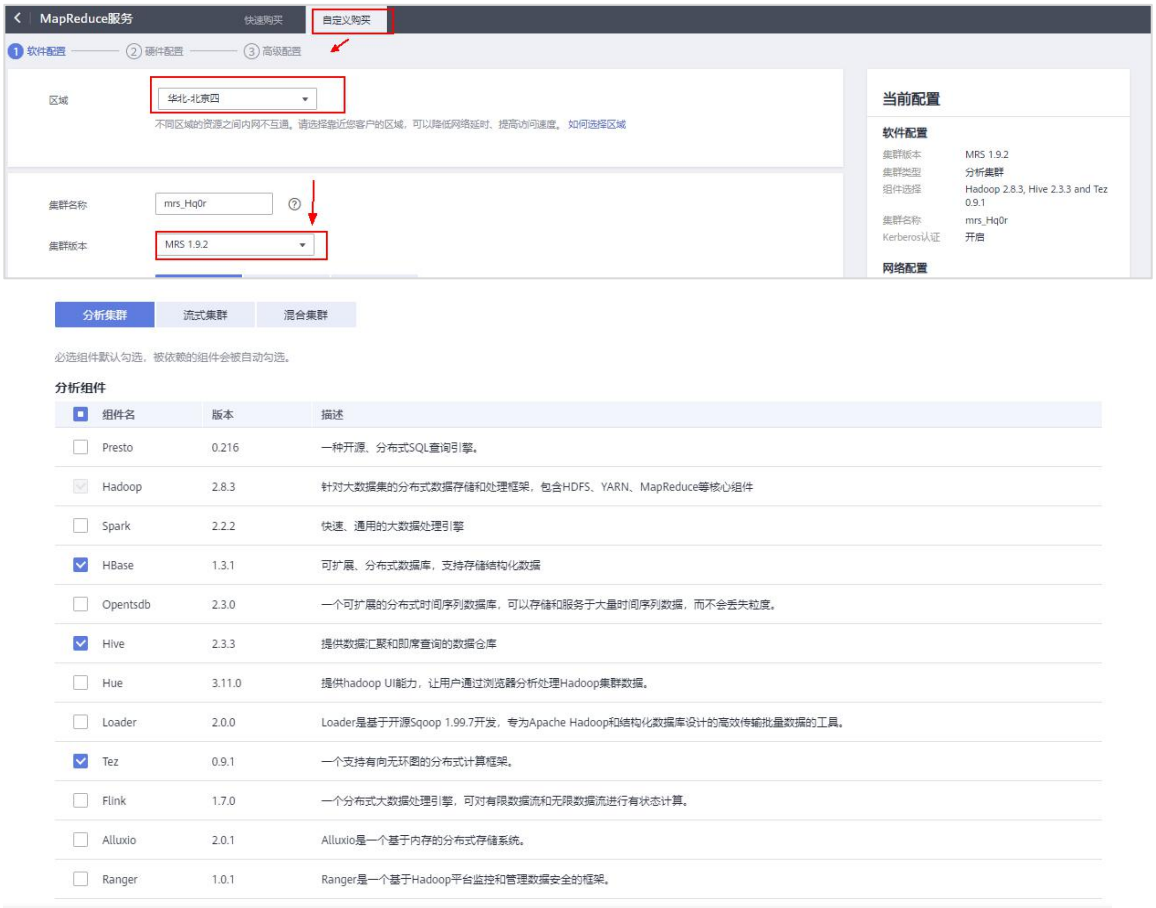


注意：华为云已统一登录入口，若仍不能登录则点击下方华为云账号进行登录。

步骤 2 点击“EI 企业智能”选择“MapReduce 服务”

步骤 3 点击“立即购买”

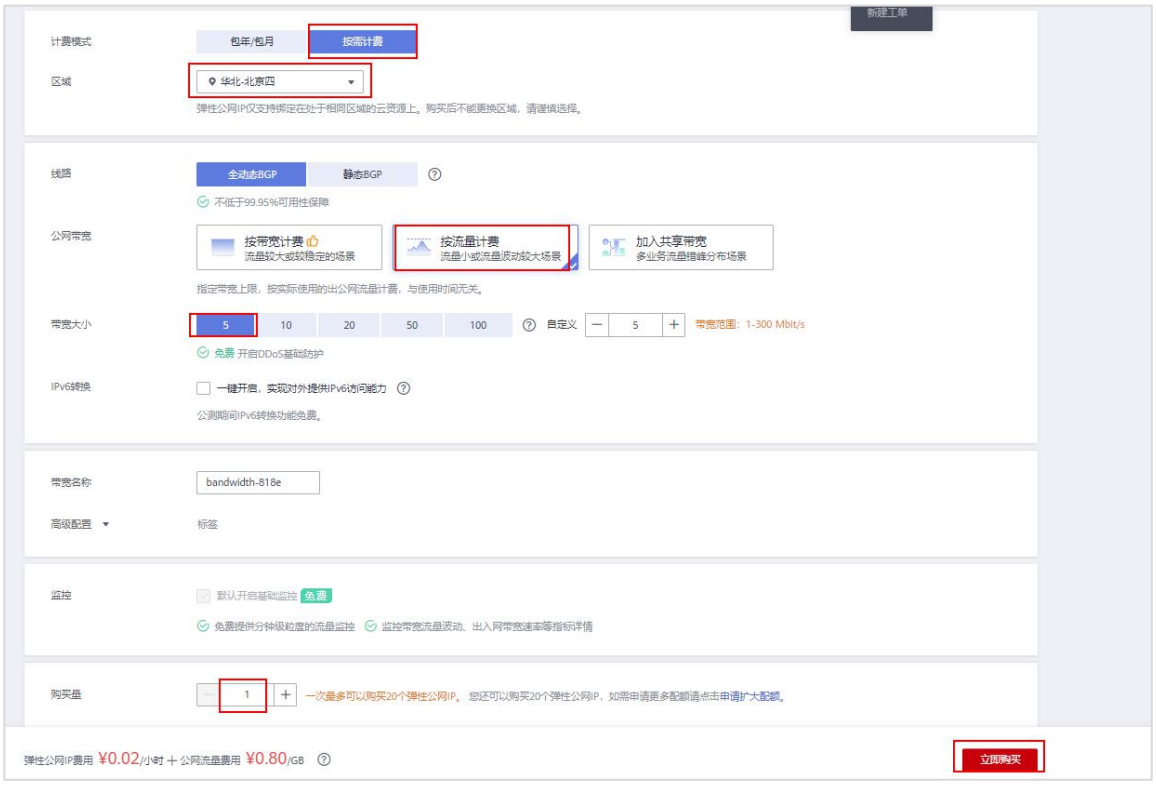
选择“自定义购买”(这里还要选上 Spark)



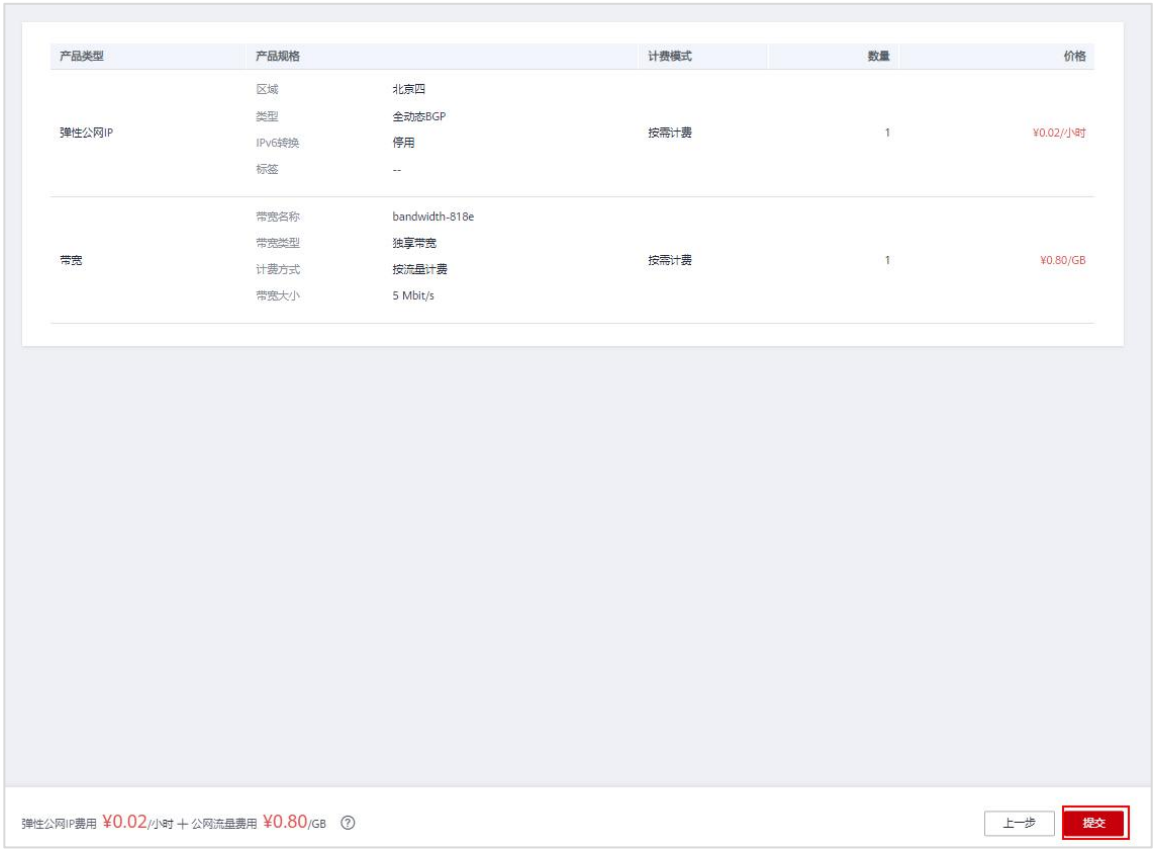
点击下一步，进入硬件配置

选择“按需计费”，“可用区 2”，点击“弹性公网 IP”，如下图：

点击“购买弹性公网 IP”，选择“按需计费”，“按流量计费”，“5M”,点击“立即购买”，如下图：



点击“提交”，如下图：



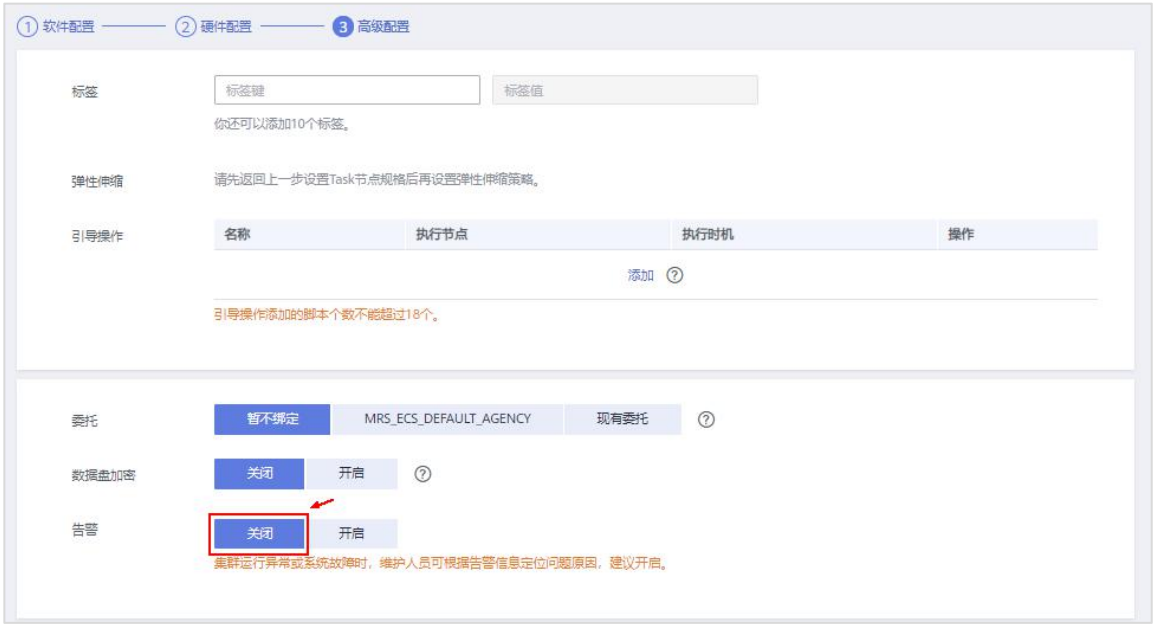
购买成功，如下图：

返回 MapReduce 服务自定义购买界面绑定 EIP

选择“鲲鹏计算”，关闭高可用，调整 core 节点数为 1,如下图：

点击“下一步”

高级配置项参考如下：



点击“确认授权”

点击“立即购买”

步骤 4 点击“返回集群列表”，如下图：

创建过程需要等待几分钟，待状态变为“运行中”集群创建完成

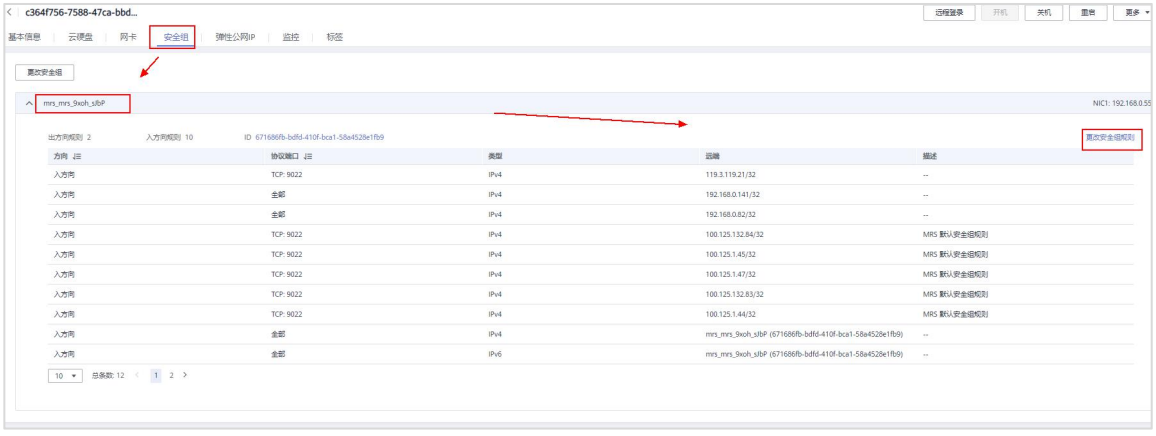
步骤 5 点击集群名称

步骤 6 配置安全组

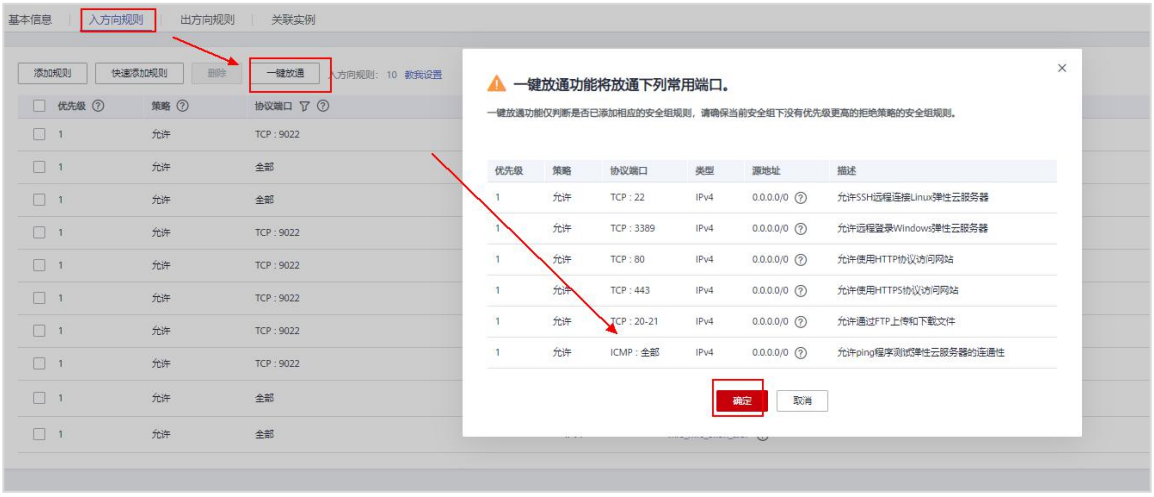
点击集群名称

选择“节点管理”，点击含有“master1”的节点

在弹出页面中选择“安全组”，点击“更改安全组规则”，如下图所示：

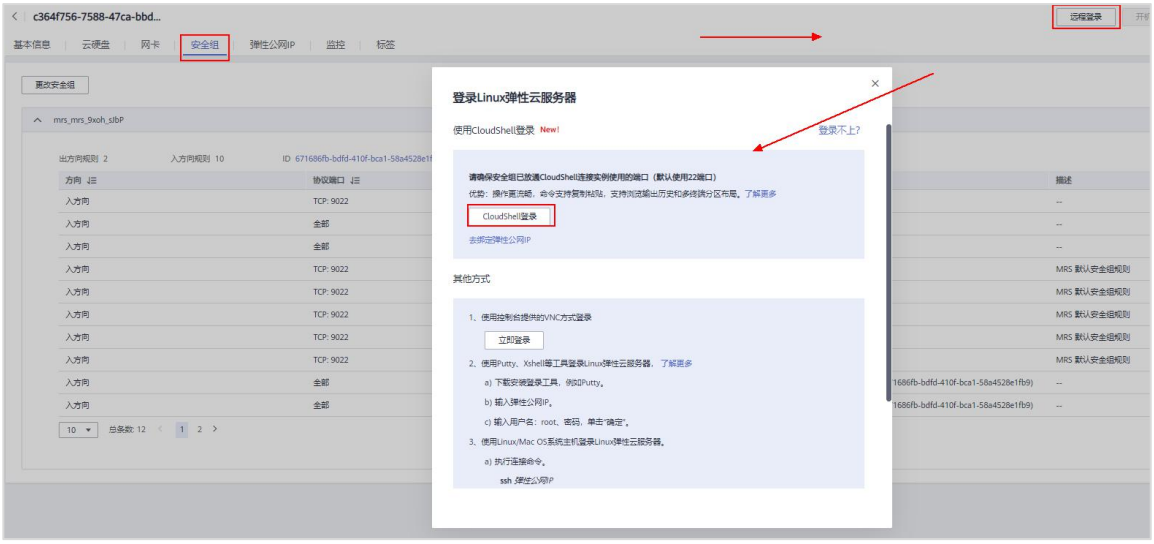


选择“入方向规则”，点击“一键放通”，确认即可。

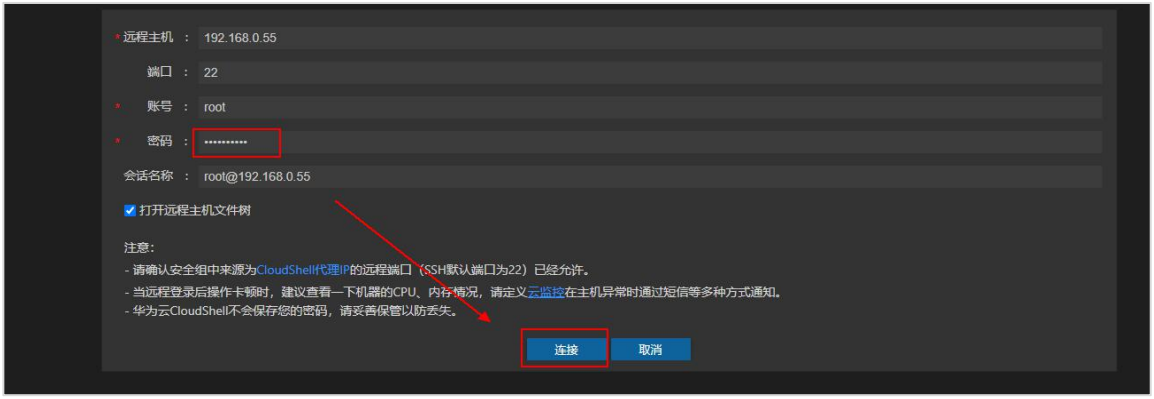


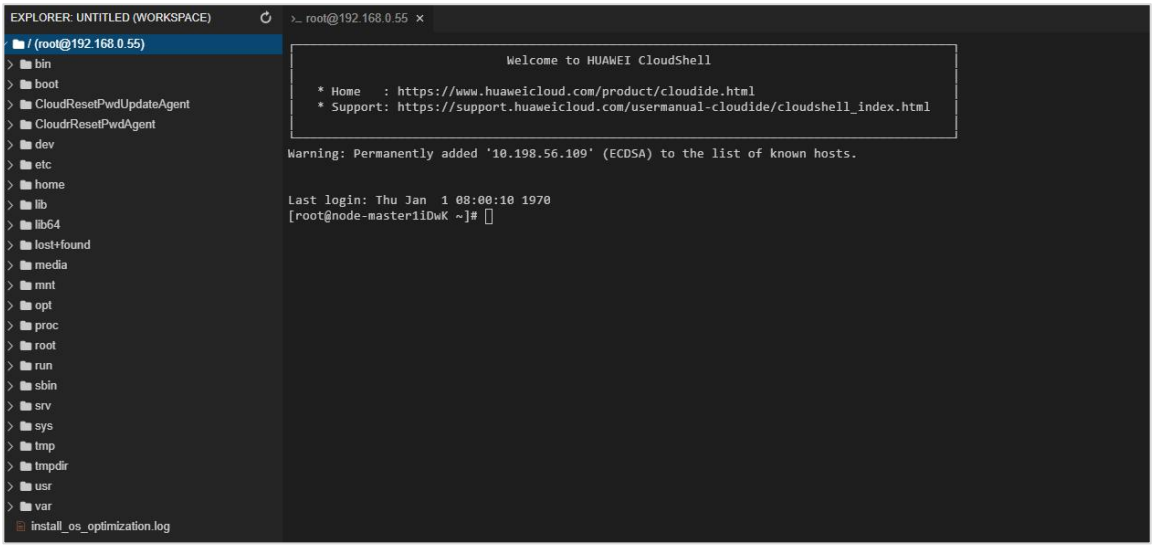
步骤 7 远程登录 master 节点

在安全组配置项，点击右上方“远程登录”，选择 cloudshell 登录。



输入密码，点击连接即可。





四：实验内容及步骤、实验的详细记录、实验结果分析

请附上实验过程截图（截图需包含指令）以及必要的文字分析

4.1 MapReduce

4.1.1 进入 hadoop(5’)

cd /opt/client/HDFS/hadoop

4.1.2 添加环境变量(10’)

export HADOOP="/opt/client/HDFS/hadoop/share/hadoop"

export CLASSPATH="$HADOOP/common/hadoop-common-2.8.3-mrs-1.9.0.jar:$HADOOP/mapreduce/hadoop-mapreduce-client-core-2.8.3-mrs-1.9.0.jar:$HADOOP/common/lib/commons-cli-1.2.jar:$CLASSPATH"

4.1.3 创建 java 程序 WordCount.java,在里面输入以下代码(5’)

```
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {

  public static class TokenizerMapper
       extends Mapper<Object, Text, Text, IntWritable>{

    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(Object key, Text value, Context context
                    ) throws IOException, InterruptedException {
      StringTokenizer itr = new StringTokenizer(value.toString());
      while (itr.hasMoreTokens()) {
        word.set(itr.nextToken());
        context.write(word, one);
      }
    }
  }

  public static class IntSumReducer
       extends Reducer<Text,IntWritable,Text,IntWritable> {
    private IntWritable result = new IntWritable();
```

```
    public void reduce(Text key, Iterable<IntWritable> values,
                       Context context
                       ) throws IOException, InterruptedException {
      int sum = 0;
      for (IntWritable val : values) {
        sum += val.get();
      }
      result.set(sum);
      context.write(key, result);
    }
  }

  public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
  }
}
```

4.1.4 编译 WordCount.java(5’)

4.1.5 创建文件 test1，内容为 hello hust，文件 test2，内容为 hello 学号，将他们放入 hdfs 的/input 文件夹内。(方法见实验一)(5’)

4.1.6运行WordCount.jar 将hdfs的/input作为输入，/output 作为输出，并打印/output 目录下的文件，显示出词频统计的结果(5’)

export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/client/HDFS/hadoop/WordCount.jar

hadoop jar WordCount.jar WordCount hdfs:///input hdfs:///output

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

4.2 Spark

4.2.1 打开 spark(5’)

Pyspark

4.2.2 读取 hdfs 文件内容(5’)

lines = spark.read.text("hdfs:///input").rdd.map(lambda r: r[0])

4.2.3 词频统计(5’)

counts = lines.flatMap(lambda x: x.split(' ')).map(lambda x: (x, 1)).reduceByKey(lambda x, y: x + y)

output = counts.collect()

4.2.4 输出结果(10’)

4.3 附加题

file1:

20210001 Math 90

20210002 Math 80

20210003 Math 70

file2:

20210001 English 80

20210002 English 70

20210003 English 60

1.将以上两个文件存入 hdfs(10’)

2.编写 mapreduce 的程序，输出每门课的平均成绩。(10’)

3.编写 mapreduce 的程序，输出每位同学有多少门课成绩低于 75 分。(10’)

五：实验总结(10’)

附录：

1 MapReduce 官方教程

https://hadoop.apache.org/docs/r2.8.3/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html

2 Spark 官方教程

https://spark.apache.org/examples.html