# 搭建基于 HDFS 碎片文件存储服务

安装 JDK

HDFS 依赖 Java 环境，这里我们使用 yum 安装 JDK 8，在终端中键入如下命令：

yum -y install java-1.8.0-openjdk\*

使用如下命令查看下 Java 版本，我们可以验证 JDK 是否已成功安装：

java -version  
  
配置 Java 环境变量

在编辑器中打开文件 /etc/profile，在文件末尾追加如下内容，配置 Java 环境变量：

export JAVA\_HOME=/usr/lib/jvm/jre-1.8.0-openjdk

export PATH=$PATH:$JAVA\_HOME/bin

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar  
  
然后执行如下命令，让环境变量生效：

source /etc/profile  
  
通过如下命令，验证 Java 环境变量是否已成功配置并且生效：

echo $JAVA\_HOME

### **准备 HDFS 环境**

安装 Hadoop

首先创建 /data/hadoop 目录，然后进入该目录：

mkdir -p /data/hadoop && cd $\_  
  
接着下载 Hadoop 安装包到该目录下：  
wget http://archive.apache.org/dist/hadoop/core/hadoop-2.7.1/hadoop-2.7.1.tar.gz  
  
解压下载好的 Hadoop 安装包到当前目录：  
tar -zxvf hadoop-2.7.1.tar.gz  
然后将解压后的目录重命名为 hadoop，并且将其移至 /usr/local/ 目录下：  
mv hadoop-2.7.1 hadoop && mv $\_ /usr/local/  
  
在编辑器中打开 Hadoop 环境配置文件，使用 Ctrl + F 搜索如下行

export JAVA\_HOME=${JAVA\_HOME}

将其替换为：

export JAVA\_HOME=/usr/lib/jvm/jre-1.8.0-openjdk  
  
查看下 Hadoop 的版本：  
/usr/local/hadoop/bin/hadoop version  
可以看到 Hadoop 的版本信息：  
Hadoop 2.7.1  
安装 Hadoop 就是这么简单。至此 Hadoop 已经安装完成，紧接着需要做的就是修改 Hadoop 的配置信息。  
  
  
修改 Hadoop 配置

由于实践环境是在单机下进行的，所以此处把 Hadoop 配置为伪分布式模式。

首先新建若干临时文件夹，在后续的配置中以及 HDFS 和 Hadoop 的启动过程中会使用到这些文件夹：

cd /usr/local/hadoop && mkdir -p tmp dfs/name dfs/data

修改 HDFS 配置文件 core-site.xml，将

configuration

配置修改为如下内容

core-site.xml

<configuration>

<property>

<name>hadoop.tmp.dir</name>

<value>/usr/local/hadoop/tmp</value>

</property>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>

修改 HDFS 配置文件 yarn-site.xml，将 configuration 配置修改为如下内容：

hdfs-site.xml

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>file:/usr/local/hadoop/dfs/name</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>file:/usr/local/hadoop/dfs/data</value>

</property>

<property>

<name>dfs.permissions</name>

<value>false</value>

</property>

</configuration>

以上配置修改完毕后，尝试启动 Hadoop。

启动 Hadoop

首先进入如下目录：

cd /usr/local/hadoop/bin/

对 HDFS 文件系统进行格式化：  
./hdfs namenode -format  
接着进入如下目录：  
cd /usr/local/hadoop/sbin/  
先后执行如下两个脚本启动 Hadoop：  
./start-dfs.sh  
./start-yarn.sh  
接着执行如下命令，验证 Hadoop 是否启动成功：  
jps  
如果输出了类似如下的指令，则说明 Hadoop 启动成功啦

21713 Jps  
21089 SecondaryNameNode  
21364 NodeManager  
20795 NameNode  
20923 DataNode  
21245 ResourceManager

 HDFS 环境就已经搭建好了。在浏览器中访问如下链接，应该能正常访问（注：若出现安全拦截请选择通过）：

http://yourIP:50070/explorer.html#/

接下来，实践下如何将碎片文件存储到 HDFS 中。

### **存储碎片文件**

准备碎片文件

首先创建目录用于存放碎片文件，进入该目录：

mkdir -p /data/file && cd /data/file

然后执行如下 shell 命令，新建一批碎片文件到该目录下

 i=1; while [ $i -le 99 ]; do name=`printf "test%02d.txt"  $i`; touch "$name"; i=$(($i+1)); done

在终端中执行 ls 命令，可以看到已成功创建了一批碎片文件。

将碎片文件存储在 HDFS 中

首先在 HDFS 上新建目录：

/usr/local/hadoop/bin/hadoop fs -mkdir /dest

此时，在浏览器是访问如下链接，可以看到 /dest 目录已创建，但是暂时还没有内容：

http://yourIP:50070/explorer.html#/dest

接着可以上传碎片文件啦！

首先在终端中依次执行以下命令

 groupadd supergroup

usermod -a -G supergroup root

然后将之前创建的碎片文件上传到 HDFS 中：

cd /data/file && /usr/local/hadoop/bin/hadoop fs -put \*.txt /dest

这时，再使用如下命令，我们应该能看到碎片文件已成功上传到 HDFS 中：/usr/local/hadoop/bin/hadoop fs -ls /dest

### **部署完成**

访问服务

在浏览器是访问如下链接，可以看到 /dest 目录的文件内容：

http://yourIP:50070/explorer.html#/dest