第2章 HDFS文件系统

2.2 【实验】WebHDFS

需要复用2.1节的虚拟机进行本节的实验操作。

2.2.1 实验目的

在前文的内容中讲解了HDFS相关的内容，重点集中在如何使用Shell命令来实现对HDFS的管理操作。在本节内容中，将讲解WebHDFS，即通过Web命令来实现对HDFS的管理操作。

2.2.2 实验环境

操作系统：CentOS6.5操作系统

计算机资源：CPU 1核 0.5GHz 内存 4GB 硬盘 10.00GB

实验环境：易优云大数据实验实训平台 Hadoop_模板2

2.2.3 实验类型及实验课时

实验类型：验证性实验

实验课时：2课时

2.2.4 实验原理

curl是一个利用URL语法在命令行下工作的文件传输工具，它支持文件上传和下载，所以是综合传输工具，但按传统习惯称curl为下载工具。WebHDFS的原理是使用curl命令向指定的Hadoop集群对外接口发送页面请求，Hadoop集群的网络接口接收到请求之后，会将命令中的URL解析成HDFS上对应文件或者文件夹，URL后面的参数解析成命令、用户、权限、缓存大小等参数。待完成相应的操作之后，将结果返回给执行curl命令的客户端，并显示执行信息或者错误信息。

如果希望使用WebHDFS服务，则需要修改Hadoop集群的配置，需要配置hdfs-site.xml中的dfs.webhdfs.enabled属性。在hdfs-site.xml中开放WebHDFS服务配置参数如下所示（在模板中已经配置完成）：

```
<property>
<name>dfs.webhdfs.enabled</name>
<value>true</value>
</property>
```

WebHDFS命令的一般形式如下所示：

```
curl [-i/-X/-u/-T] [PUT]  "HTTP://<HOST>:<PORT>/webhdfs/v1/<PATH>?[user.name=<user>&]&op=<op
eration>&[doas=<user>] ..."
```

在上述命令里面，引号前面的部分是curl自己的参数；后面网页形式的内容代表着操作的命令、参数和路径。其中HTTP://<HOST>:<PORT>代表需要将命令所发送到的地址和端口，也就是Hadoop集群服务器的IP地址和HDFS管理端口（默认是50070）；在这个地址之后的部分/webhdfs/v1/<PATH>代表着需要操作的HDFS集群上的路径，比如/webhdfs/v1/master1-file,就代表着HDFS上/master1-file路径；再往后的内容就是操作的指令和参数了，其中最重要的是op参数，代表着具体的操作指令，接下来的内容我们会详细讲解。

常用的curl参数及意义如下所示：

- -i: 输出时包括protocol头信息
- -I:只显示请求头信息
- -u:设置服务器的用户和密码
- -x: 在给定的端口上使用HTTP代理
- -T:上传文件
- -L: 网址自动跳转
- -v:显示一次HTTP通信的整个过程

2.2.4.1 创建文件并写入内容

1. 使用WebHDFS创建文件的命令如下所示：

```
curl -i -X PUT "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=CREATE[&overwrite=<true|false>][&replication=<SHORT>][&permission=<OCTAL>][&buffersize=<INT>]"
```

1) 在命令中通过op=CREATE指定创建文件命令，通过<PATH>参数指定所创建的文件名称；
2) 通过可选参数overwrite=<true|false>指定当文件如果存在时是否进行覆盖；
3) 通过可选参数replication=<SHORT>指定文件的副本数；
4) 通过可选参数permission=<OCTAL>指定所创建文件的权限；
5) 通过可选参数buffersize=<INT>指定文件数据写入时的缓冲区大小。

使用WebHDFS创建文件命令之后，会返回一个Location位置信息，对应内容为http://<DATANODE>:<PORT>/webhdfs/v1/<PATH>?op=CREATE…，Location位置信息中包括了已创建文件所在的DataNode地址及创建路径。

2. 接下来就可以将需要写入的文件内容发送到所返回的Location对应的文件内，命令如下所示：

```
curl -i -X PUT -T <LOCAL_FILE> <Location>
```

命令的执行需要两个参数：

1) LOCAL_FILE对应的是客户端本地输入数据源的绝对路径；
2) Location对应的是运行上述创建文件命令后，返回的文件所在的DataNode地址及路径字符串。

2.2.4.2 文件内容追加

1. 如果希望向文件中追加内容，首先需要使用下面的命令获取待追加内容的文件所在地址：

```
curl -i -X POST "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=APPEND[&buffersize=<INT>]"
```

1) 在命令中通过op=APPEND指定向文件中追加内容，通过<PATH>参数指定所对应的文件名称；
2) 通过可选参数buffersize=<INT>指定数据缓存大小。

使用WebHDFS追加命令之后，会返回一个Location位置信息，对应内容为http://<DATANODE>:<PORT>/webhdfs/v1/<PATH>?op=APPEND…，Location位置信息中包括了文件所在的DataNode地址及文件路径。

2. 接下来结合返回的Location信息，使用如下命令进行内容追加：

```
curl -i -X POST -T <LOCAL_FILE> <Location>
```

1) LOCAL_FILE对应的是客户端本地输入数据源的绝对路径；
2) Location对应的是运行上述追加命令后，返回的文件所在的DataNode地址及路径字符串。

2.2.4.3 打开并读取文件内容

使用下面的命令可以打开HDFS上的文件并读取内容：

```
curl -i -L "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=OPEN[&offset=<LONG>][&buffersize=<INT>]"
```

1) 在命令中通过op=OPEN指定打开文件并读取内容，通过<PATH>参数指定所对应的文件名称；
2) 通过可选参数offset=<LONG>指定读取偏移量；
3) 通过可选参数buffersize=<INT>指定数据缓冲区大小。

需要注意的是，这个命令首先会返回文件所在的Location信息，然后打印文件的具体内容。

2.2.4.4 创建文件夹

通过下列的命令可以在HDFS中进行文件夹创建：

```
curl -i -X PUT http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=MKDIRS[&permission=<OCTAL>]
```

1) 在命令中通过op=MKDIRS指定创建文件夹命令，通过<PATH>参数指定所创建的文件夹名称；
2) 通过可选参数permission=<OCTAL>指定所创建文件夹的操作权限。

2.2.4.5 文件重命名

通过下列命令可以进行文件夹或文件的重命名：

```
curl -i -X PUT "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=RENAME&destination=<PATH>"
```

1) 在命令中通过op=RENAME指定重命名命令，通过<PATH>参数指定需要被重命名的文件夹或者文件名称；
2) 通过参数destination=<PATH>指定重命名后的名称。

2.2.4.6 文件删除

通过下列命令可以删除文件夹或者文件：

```
curl -i -X DELETE "http://<host>:<port>/webhdfs/v1/<PATH>?op=DELETE[&recursive=<true|false>]"
```

1) 在命令中通过op=DELETE指定删除命令，通过<PATH>参数指定需要被删除的文件夹或者文件名称；
2) 通过参数recursive=<true|false>指定是否对文件夹中的内容实现递归删除。

2.2.4.7 查看文件属性

通过指定op=GETFILESTATUS可以进行文件夹或文件信息的查看，如下所示：

```
curl -i "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=GETFILESTATUS"
```

2.2.4.8 查看文件夹内容

通过指定op=LISTSTATUS可以列举出文件夹中的内容，如下所示：

```
curl -i "http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=LISTSTATUS"
```

2.2.5 【学生端】实验步骤

上文中讲解如何配置WebHDFS以及WebHDFS的基本操作步骤，接下来我们将详细介绍WebHDFS命令的组织方式以及具体命令实现。

2.2.5.1 启动**Hadoop**集群

在进行实验之前，首先需要确保Hadoop集群已经正常启动。如果Hadoop没有启动，需要通过下列步骤启动Hadoop集群。

在模板中，我们已经配置好了**Hadoop**伪分布式环境，同学们不需要再次配置，可以直接启动使用。

步骤1. 启动Hadoop

打开一个终端模拟器，通过命令启动Hadoop。

步骤2. 验证Hadoop是否启动成功

通过命令，查看相应的JVM进程确定Hadoop是否启动成功。

步骤3. 检测WebHDFS是否可用

Hadoop启动之后，使用命令检测WebHDFS是否可用。

2.2.5.2 创建文件并写入内容

步骤1. 创建输入源文件

1. 通过命令在本地创建输入数据源文件。
2. 通过命令测试数据源文件是否创建成功。

步骤2. 创建webhdfsFile文件

通过命令，使用root用户在HDFS中创建文件webhdfsFile。

步骤3. 文件内容写入

结合上文返回的Location位置信息，通过命令将本地文件/home/webfile1.txt中的内容写入到DataNode对应路径下的文件内。

步骤4. 查看文件内容

使用命令验证webhdfsFile中的内容是否写入成功。

2.2.5.3 文件内容追加

如果希望向文件中追加内容，首先需要获取待追加内容的文件所在地址，然后根据获取到的地址信息实现向文件中追加内容。

步骤1. 创建输入源文件

1. 通过命令在本地创建输入数据源文件。
2. 测试数据源文件是否创建成功。

步骤2. 获取文件位置

获取待追加内容的文件/webhdfsFile所在地址。

步骤3. 文件内容追加

结合返回内容的Location信息实现文件内容追加。

步骤4. 查看文件内容

使用命令验证webhdfsFile中的内容是否追加成功。

2.2.5.4 打开并读取文件内容

通过命令，打开并读取/webhdfsFile文件中的内容。

2.2.5.5 创建文件夹

通过命令进行文件夹创建（在此创建一个名称为webhdfsDir的文件夹）。

2.2.5.6 文件重命名

步骤1. 将文件夹重新命名

通过命令实现将webhdfsDir文件夹重命名为webhdfsDir1。

步骤2. 验证重命名是否成功

确定文件夹是否重新命名。

2.2.5.7 删除文件

步骤1. 文件删除

通过命令删除文件/webhdfsFile。

步骤2. 验证删除命令是否执行成功

进行HDFS文件查看，确定文件是否被删除。

2.2.5.8 查看文件属性

通过命令查看/webhdfsDir1的属性信息。

2.2.5.9 查看文件夹内容

通过命令列举文件夹的内容。

2.2.6 常见问题

2.2.6.1 RPC通信错误

当执行WebHDFS命令时候，出现类似`Server IPC version 9 cannot communicate with client version 470`错误信息，如下所示：

```
[root@master ~]# curl -i -X PUT "http://master:9000/webhdfs/v1/wenhdfsFile?op=CREATE"
org.apache.hadoop.ipc.RPC$VersionMismatch*>Server IPC version 9 cannot communicate with c
lient version 470:@[root@master ~]#
```

此时应该着重考虑是否是请求的协议写错了，以及通信端口是否填写错误。在上文的报错信息中，由于错误的将http://master:50070写成了http://master:9000，因此会出现错误信息。

2.2.6.2 权限错误

当使用WebHDFS命令向HDFS文件系统中写入数据的时候，首先需要确保所使用的用户是否拥有HDFS的写入权限。

如果在WebHDFS命令中没有显示的指定用户，则会使用默认的dr.who作为访问用户，而如果没有在配置文件中进行配置，此用户是没有权限对HDFS中的文件进行写操作的，如果用户权限错误，往往会出现类似于下列的输出信息：

```
[root@master ~]# curl -i -X PUT "http://master:50070/webhdfs/v1/webhdfsDir?op=MKDIRS"
HTTP/1.1 403 Forbidden
Cache-Control: no-cache
Expires: Mon, 12 Nov 2018 08:42:59 GMT
Date: Mon, 12 Nov 2018 08:42:59 GMT
Pragma: no-cache
Expires: Mon, 12 Nov 2018 08:42:59 GMT
Date: Mon, 12 Nov 2018 08:42:59 GMT
Pragma: no-cache
Content-Type: application/json
Transfer-Encoding: chunked
Server: Jetty(6.1.26)

{"RemoteException":{"exception":"AccessControlException","javaClassName":"org.apache.hado
op.security.AccessControlException","message":"Permission denied: user=dr.who, access=WRI
TE, inode=\"/webhdfsDir\":root:supergroup:drwxr-xr-x"}}[root@master ~]#
```

```
[root@master ~]# curl -i -X POST -T /home/webfile2.txt "http://master:50075/webhdfs/v1/we
bhdfsFile?op=APPEND&namenoderpcaddress=master:9000"
HTTP/1.1 100 Continue

HTTP/1.1 403 Forbidden
Content-Type: application/json; charset=utf-8
Content-Length: 2109
Connection: close

{"RemoteException":{"exception":"AccessControlException","javaClassName":"org.apache.hado
op.security.AccessControlException","message":"Permission denied: user=dr.who, access=WRI
TE, inode=\"/webhdfsFile\":root:supergroup:-rwxr-xr-x\n\tat  org.apache.hadoop.hdfs.serve
r.namenode.FSPermissionChecker.check(FSPermissionChecker.java:319)\n\tat org.apache.hadoo
p.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:219)
\n\tat org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermi
ssionChecker.java:190)\n\tat org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPerm
ission(FSDirectory.java:1698)\n\tat org.apache.hadoop.hdfs.server.namenode.FSDirectory.ch
eckPermission(FSDirectory.java:1682)\n\tat org.apache.hadoop.hdfs.server.namenode.FSDirec
tory.checkPathAccess(FSDirectory.java:1656)\n\tat org.apache.hadoop.hdfs.server.namenode.
FSNamesystem.appendFileInternal(FSNamesystem.java:2668)\n\tat org.apache.hadoop.hdfs.serv
er.namenode.FSNamesystem.appendFileInt(FSNamesystem.java:2985)\n\tat org.apache.hadoop.hd
fs.server.namenode.FSNamesystem.appendFile(FSNamesystem.java:2952)\n\tat org.apache.hadoo
p.hdfs.server.namenode.NameNodeRpcServer.append(NameNodeRpcServer.java:653)\n\tat org.apa
che.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.append(ClientName
nodeProtocolServerSideTranslatorPB.java:421)\n\tat org.apache.hadoop.hdfs.protocol.proto.
ClientNamenodeProtocolProtos$ClientNamenodeProtocol$2.callBlockingMethod(ClientNamenodePr
otocolProtos.java)\n\tat org.apache.hadoop.ipc.ProtobufRpcEngine$Server$ProtoBufRpcInvoke
r.call(ProtobufRpcEngine.java:616)\n\tat org.apache.hadoop.ipc.RPC$Server.call(RPC.java:9
69)\n\tat org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2049)\n\tat org.apache.h
adoop.ipc.Server$Handler$1.run(Server.java:2045)\n\tat java.security.AccessController.doP
rivileged(Native Method)\n\tat javax.security.auth.Subject.doAs(Subject.java:422)\n\tat o
rg.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1657)\n\tat
org.apache.hadoop.ipc.Server$Handler.run(Server.java:2043)\n"}}[root@master ~]# curl -i -
X POST "http://master:500r.name=root&op=APPEND"ile?use
HTTP/1.1 307 TEMPORARY_REDIRECT
Cache-Control: no-cache
Expires: Mon, 12 Nov 2018 08:25:37 GMT
Date: Mon, 12 Nov 2018 08:25:37 GMT
Pragma: no-cache
Expires: Mon, 12 Nov 2018 08:25:37 GMT
Date: Mon, 12 Nov 2018 08:25:37 GMT
Pragma: no-cache
```

```
Content-Type: application/octet-stream
Set-Cookie: hadoop.auth="u=root&p=root&t=simple&e=1542047137389&s=emkW9j8r6EANJW60pc4ShuF
y/VQ="; Path=/; Expires=???, 12-???-2018 18:25:37 GMT; HttpOnly
Location: http://master:50075/webhdfs/v1/webhdfsFile?op=APPEND&user.name=root&namenoderpc
address=master:9000
Content-Length: 0
Server: Jetty(6.1.26)
```

解决权限错误的问题往往是比较简单的，可以在命令中通过显示的指定user.name=root，来设定以root用户的权限进行WebHDFS命令操作。

2.2.6.3 HTTP/1.1 400 Bad Request

出现HTTP/1.1 400 Bad Request错误，往往是由于命令行拼写错误所造成的，如下所示：

```
[root@master ~]# curl -i -X PUT "http://master:50070/webhdfs/v1/webhdfsDir?user.name=root
&op= RENAME&destination=/webhdfsDir1"
HTTP/1.1 400 Bad Request
Connection: close
Server: Jetty(6.1.26)
```

在上述错误中，由于在op= RENAME拼写中多写了一个空格，因此导出命令行识别失败，从而出现Bad Request错误。

2.2.6.4 其他错误

在执行写入或者追加等操作时，需要结合命令运行所返回的Location地址来进行内容写入，同时需要将返回的Location地址用双引号（""）括起来，否则可能会出现类似下面的错误：

```
[root@master /]# curl -i -X POST -T /home/webfile2.txt http://master:50075/webhdfs/v1/web
hdfsFile?op=APPEND&user.name=root&namenoderpcaddress=master:9000
[1] 3096
[2] 3097
[root@master /]# bash: user.name=root: command not found
HTTP/1.1 100 Continue

HTTP/1.1 400 Bad Request
Content-Type: application/json; charset=utf-8
Content-Length: 161
Connection: close

{"RemoteException":{"exception":"IllegalArgumentException","javaClassName":"java.lang.Ill
egalArgumentException","message":"java.net.UnknownHostException: null"}}
```

在上述错误中，由于在没有将Location地址/home/webfile2.txt http://master:50075/webhdfs/v1/webhdfsFile?op=APPEND&user.name=root&namenoderpcaddress=master:9000用双引号括起来，所以会出现命令解析错误。

2.2.7 课后思考

1. 请思考WebHDFS的作用及应用场景。
2. 在上文所述内容中，由于权限管理限制，需要手动指定user.name才能进行HDFS文件写操作，请查阅相关资料实现以默认的用户进行HDFS文件写操作。