

课堂实验

一、基本信息

文档编号		文档版本	1.0
实验名称	HDFS 基本操作		
所属课程	第 2 讲 分布式存储系统 HDFS	认证等级	数据工程师（初级）
授课形式	上机实验	实验批次	第 3 次 / 共 11 次

二、实验目的

- 掌握 HDFS 基本命令，具备管理 HDFS 目录和文件的能力。
- 理解 HDFS Block 与 Linux 文件的关系。
- 理解 Namenode 的 Active 和 Standby 状态，手动实现主备切换。
- 实践利用 HDFS 快照功能备份、恢复目录。

三、实验准备

- 下载并安装 TDH Client。
- 实验目录与命名规划
 - (1) 本地目录
工作目录: /mnt/disk1/{student_name}
 - (2) HDFS 目录
工作目录: /tmp/{student_name}
HDFS 实验目录: /tmp/{student_name}/hdfs_data
 - (3) {student_name} 为变量，代表学员姓名全拼
- 文件服务器
 - (1) IP: 172.16.140.111
 - (2) 目录: /mnt/disk1/de_training

四、实验内容

1、启动 TDH Client

- 任务：执行 TDH Client 的 init.sh 脚本，启动 TDH Client。

- 步骤

Linux:

```
// 执行 TDH Client 的 init.sh 脚本
1. source {TDH_Client_install_dir}/init.sh
```

2、查看 Java 服务进程

- 任务: 查看集群各节点的 Java 后台服务进程, 判断服务是否正常启动。

- 步骤

(1) 查看节点的 Java 后台服务进程。

Linux:

```
// 显示 Java 进程信息
1. jps
```

(2) 查看节点中指定 Docker (容器) 的 Java 后台服务进程。

Linux:

```
// 列出所有 HDFS 相关的 Docker 信息
1. docker ps | grep hdfs
// 列出所有 NameNode 相关的 Docker 信息
2. docker ps | grep hdfs-namenode
// 列出所有 DataNode 相关的 Docker 信息, 第一列 (红框内) 是 Docker Id
3. docker ps | grep hdfs-datanode
```

```
[root@tdh-204 ~]# docker ps | grep hdfs-datanode
b0af7699efde    tdh-204:5000/transwarp/hdfs:transwarp-5.1.2-final    "boot.sh HDFS_DATA..."
-n0rs1-1085830881-bh7qq_default_e4a3a0af-5294-11e8-89fc-000c29a3f51a_a4fc1e23
98d731099c9d    transwarp/pause:transwarp-5.1.2-final                "/pause"
default_e4a3a0af-5294-11e8-89fc-000c29a3f51a_b3349918
```

```
// 在指定 Docker 中启动 Bash
4. docker exec -it {dockerId} /usr/bin/bash
// 查看 Docker 中运行的 Java 后台服务进程
5. jps
```

```
[root@tdh-204 ~]# docker exec -it b0af7699efde /usr/bin/bash
[root@tdh-204 ~]# jps
5015 Jps
42 DataNode
```

```
// 退出 Docker Bash
6. exit
```

3、创建目录并修改权限

- 任务: 在 HDFS 上创建目录“/tmp/{student_name}/hdfs_data”, 并修改目录权限为 rwxrwxrwx。

- 步骤

Linux:

```
// 将 Hadoop 当前用户设置为 hdfs, 实现 HDFS 访问授权
1. export HADOOP_USER_NAME=hdfs
```

```
// 创建 HDFS 目录，并修改目录权限为 rwxrwxrwx
2. hadoop fs -mkdir -p /tmp/{student_name}/hdfs_data
3. hadoop fs -chmod -R 777 /tmp/{student_name}
```

4、上传文件

- 任务：将本地文件 “/var/log/transwarp-manager/master/transwarp-manager.log” 上传到 HDFS 目录 “/tmp/{student_name}/hdfs_data” 中。

- 步骤

Linux:

```
1. cd /var/log/transwarp-manager/master
2. hadoop fs -put transwarp-manager.log /tmp/{student_name}/hdfs_data
```

5、查看目录内容

- 任务：查看 HDFS 目录 “/tmp/{student_name}/hdfs_data” 的内容。

- 步骤

Linux:

```
1. hadoop fs -ls /tmp/{student_name}/hdfs_data
```

6、查看文件内容

- 任务：查看 HDFS 文件 “/tmp/{student_name}/hdfs_data/transwarp-manager.log” 的内容。

- 步骤

Linux:

```
1. hadoop fs -cat /tmp/{student_name}/hdfs_data/transwarp-manager.log
```

7、通过 Web 查看目录和文件数据块

- 任务：通过 Transwarp Manager 浏览 HDFS 目录，并找到文件“/tmp/{student_name}/hdfs_data/transwarp-manager.log” 的 Block Id。

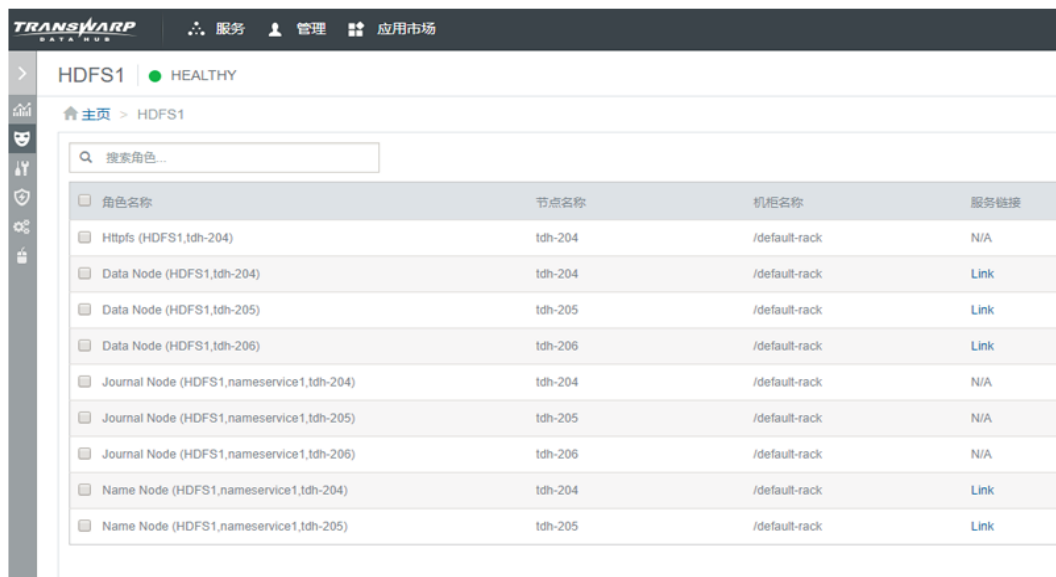
- 步骤

Web:

```
1. 登录 Transwarp Manager
```

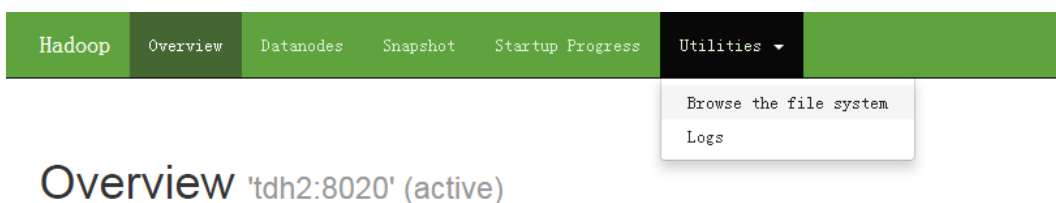


2. 进入 HDFS 服务，点击左侧“角色”按钮

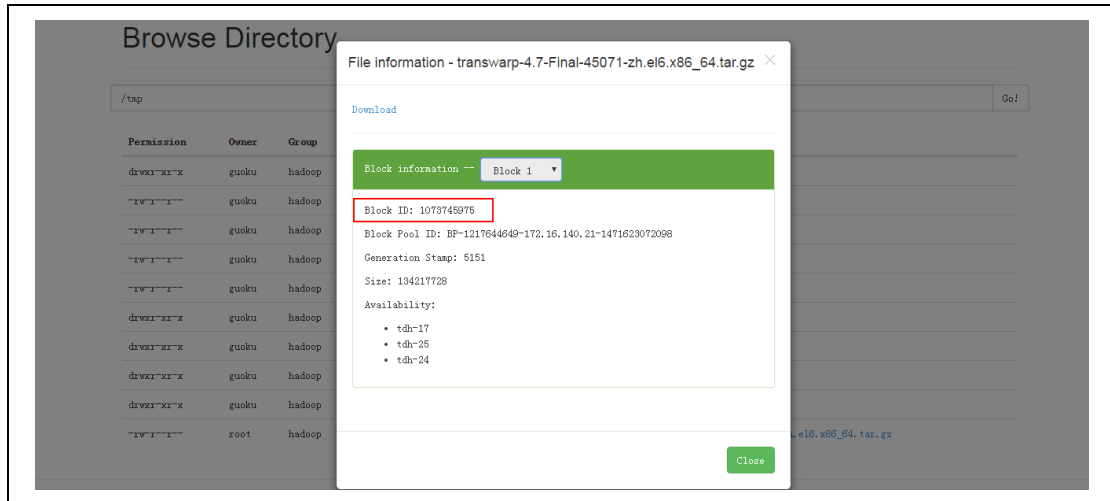


3. 点击 Namenode 后面的“Link”，确保进入的是 Active Namenode

4. 点击菜单项“Utilities/Browse the file system”，浏览 HDFS 目录



5. 找到文件“/tmp/{student_name}/hdfs_data/transwarp-manager.log”，查看它的 block ID



8、查找 Block 文件

- 任务：在集群各节点的本地文件系统中，查找任务 8 中 Block Id 对应的 Block 文件。
- 步骤

Linux:

```
1. find / -name blk* | grep {block_id}
```

9、查看 Namenode 状态

- 任务：查看 Namenode 的主备状态，Active 还是 Standby。
- 步骤

Linux:

// 查看 Namenode nn1 的主备状态，nn1 为 Namenode ID

1. hdfs haadmin -getServiceState nn1
2. hdfs haadmin -getServiceState nn2

10、Namenode 主备切换

- 任务：手工实现 Namenode 主备切换，即 Active 与 Standby 互换。
- 步骤

Linux:

// 切换 Namenode nn1 和 nn2 的主备状态，nn1 为 Active，nn2 为 Standby

1. hdfs haadmin -failover nn1 nn2

11、创建快照

- 任务：为 HDFS 文件夹 “/tmp/{student_name}/hdfs_data” 创建快照。

- 步骤

Linux:

// 设置允许快照

1. hdfs dfsadmin -allowSnapshot /tmp/{student_name}/hdfs_data

// 创建快照

2. hdfs dfs -createSnapshot /tmp/{student_name}/hdfs_data bak1

12、利用快照恢复文件

- 任务：删除目录 “/tmp/{student_name}/hdfs_data” 中的所有文件，然后利用快照恢复。

- 步骤

Linux:

// 删除目录下的所有文件，并确认删除情况

1. hadoop fs -rm -skipTrash /tmp/{student_name}/hdfs_data/*

2. hadoop fs -ls /tmp/{student_name}/hdfs_data

// 将快照目录下的文件复制到原目录下，实现快照恢复，并查看文件恢复情况

3. hadoop fs -cp /tmp/{student_name}/hdfs_data/.snapshot/bak1/*
/tmp/{student_name}/hdfs_data

4. hadoop fs -ls /tmp/{student_name}/hdfs_data