

IPFS测试文档

IPFS服务器

校内

```
# IP
222.20.126.154
# Port
node0-node3:5000-5002
```

校外

```
# IP
101.200.162.128
# Port
node0-node3:5000-5002
```

测试过程

- 使用Docker, 多虚拟机或多主机搭建有三个节点的IPFS私有网络
- 使用IPFS命令行工具, 在一个节点上传文件, 并在另一个节点下载文件
- 改进IPFS Java客户端代码, 实现上传和下载文件的HTTP接口, 每次调用接口不再重新连接IPFS服务器
- 为每个节点分别部署IPFS Java客户端, 在一个节点上传文件, 并在另一个节点下载文件

IPFS私有网络搭建

详细见:

https://github.com/mapkkkk/ipfs/blob/master/doc/private_ipfs_cluster_build.md

不过有一个问题: docker-compose部署的cluster没办法配置策略(只能从localhost发出http请求), 所以需要打洞出去(先代理进本机, 再端口转发)

docker-compose的config见git repo的config:

<https://github.com/mapkkkk/ipfs/blob/master/config/docker-compose.yml>

使用IPFS命令行工具上传文件截图及描述

```
ipfs@node1: ~/.ipfs
ipfs@node1: ~/.ipfs$ chmod 400 ~/.ipfs/swarm.key
ipfs@node1: ~/.ipfs$ ipfs swarm connect /ip4/192.168.31.169/tcp/4001/ipfs/12D3KooWEz8i3MfjFLQq4CjEFABGoCPVWaEpztpCM1tzB78wN7VW
connect 12D3KooWEz8i3MfjFLQq4CjEFABGoCPVWaEpztpCM1tzB78wN7VW success
ipfs@node1: ~/.ipfs$ mkdir test
ipfs@node1: ~/.ipfs$ cd test
ipfs@node1: ~/.ipfs/test$ nano test_node1.txt
ipfs@node1: ~/.ipfs/test$ cd ..
ipfs@node1: ~/.ipfs$ ls
api      config      datastore_spec  keystore      swarm.key  version
blocks   datastore   gateway         repo.lock     test
ipfs@node1: ~/.ipfs$ ipfs add test/test_node1.txt
added QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK test_node1.txt
17 B / 17 B [=====] 100.00%
ipfs@node1: ~/.ipfs$
```

创建了一个test_node1.txt，使用ipfs add上传成功，CID为
QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK

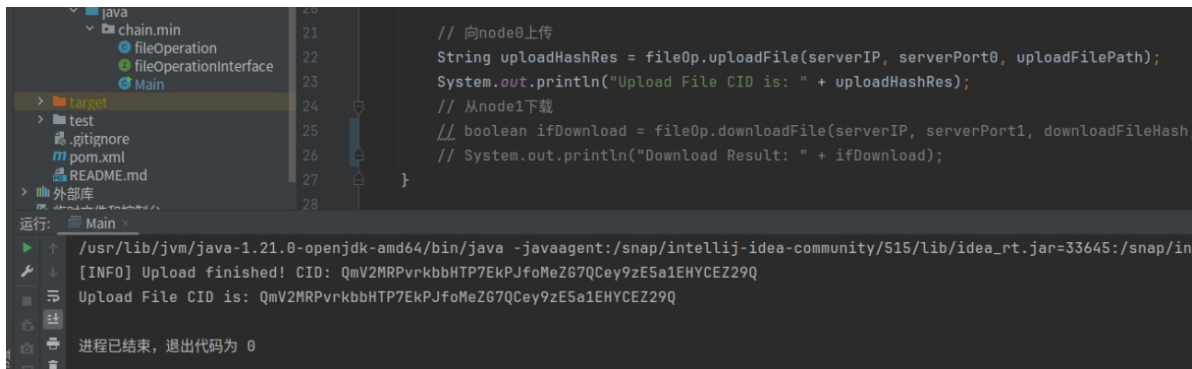
使用IPFS命令行工具下载文件截图及描述

```
ipfs@node1: ~/.ipfs/download$ ipfs get QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK
Saving file(s) to QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK
17 B / 17 B [=====] 100.00% 0s
ipfs@node1: ~/.ipfs/download$ ls
QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK
ipfs@node1: ~/.ipfs/download$ cat QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK
test_node1
12345
ipfs@node1: ~/.ipfs/download$
```

下载CID为QmQNs9dEAjGnUHNViFertELcEYgFdjLGeoe4QfkWavy4YK的文件，cat之后确实是刚才上传的文件

```
ipfs@node0: ~/ipfs-cluster-ctl$ ./ipfs-cluster-ctl status QmQiXsdDuqrikmahhYxmigi2a5uFtBWGArbCKWXGgDJvSB
QmQiXsdDuqrikmahhYxmigi2a5uFtBWGArbCKWXGgDJvSB:
  > cluster0          : UNPINNED | 2024-06-30T00:17:53.316177611Z | Attempts: 0 | Priority: false
  > cluster1          : UNPINNED | 2024-06-30T00:17:53.316177611Z | Attempts: 0 | Priority: false
  > cluster2          : UNPINNED | 2024-06-30T00:17:53.316177611Z | Attempts: 0 | Priority: false
ipfs@node0: ~/ipfs-cluster-ctl$
```

使用IPFS Java客户端上传文件截图及描述



The screenshot shows the IntelliJ IDEA IDE with a Java project. The left sidebar displays the project structure with folders like 'java', 'chain.min', 'fileOperation', 'fileOperationInterface', and 'Main'. The main editor shows a Java file with the following code:

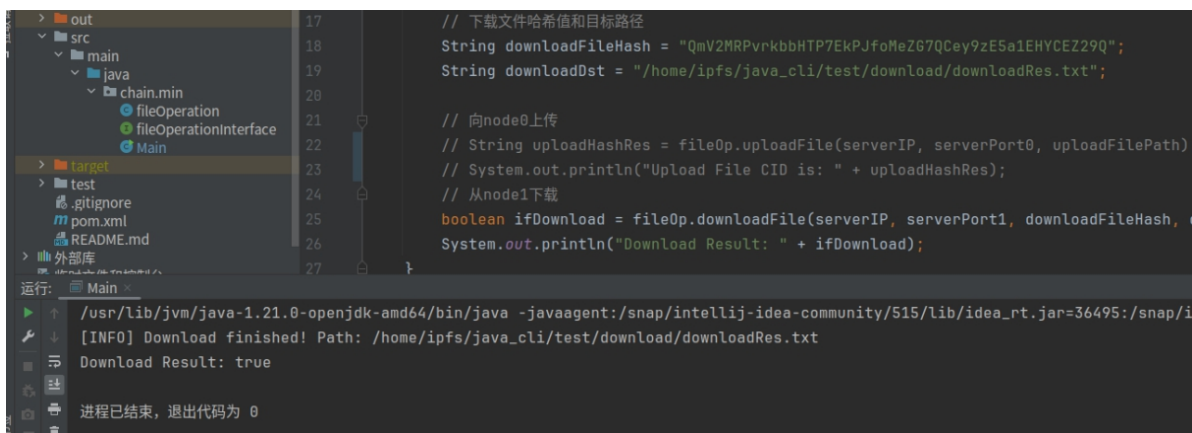
```
21 // 向node0上传
22 String uploadHashRes = fileOp.uploadFile(serverIP, serverPort0, uploadFilePath);
23 System.out.println("Upload File CID is: " + uploadHashRes);
24 // 从node1下载
25 // boolean ifDownload = fileOp.downloadFile(serverIP, serverPort1, downloadFileHash,
26 // System.out.println("Download Result: " + ifDownload);
27 }
```

The bottom console shows the execution output:

```
运行: Main
/usr/lib/jvm/java-1.21.0-openjdk-amd64/bin/java -javaagent:/snap/intellij-idea-community/515/lib/idea_rt.jar=33645:/snap/in
[INFO] Upload finished! CID: QmV2MRPvrkbbHTP7EkPJfoMeZG7QCey9zE5a1EHYCEZ29Q
Upload File CID is: QmV2MRPvrkbbHTP7EkPJfoMeZG7QCey9zE5a1EHYCEZ29Q
进程已结束,退出代码为 0
```

node0映射到5000端口, 上传测试文件后CID为:
QmV2MRPvrkbbHTP7EkPJfoMeZG7QCey9zE5a1EHYCEZ29Q

使用IPFS Java客户端下载文件截图及描述



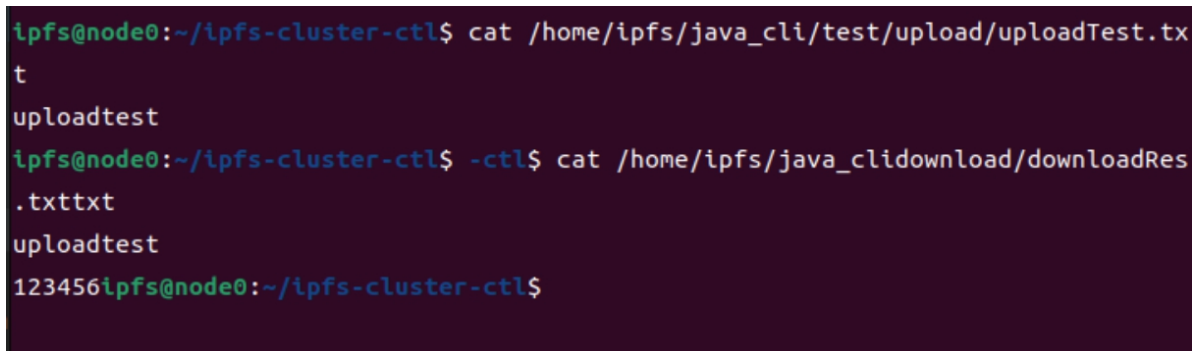
The screenshot shows the IntelliJ IDEA IDE with a Java project. The left sidebar displays the project structure with folders like 'out', 'src', 'main', 'java', 'chain.min', 'fileOperation', 'fileOperationInterface', and 'Main'. The main editor shows a Java file with the following code:

```
17 // 下载文件哈希值和目标路径
18 String downloadFileHash = "QmV2MRPvrkbbHTP7EkPJfoMeZG7QCey9zE5a1EHYCEZ29Q";
19 String downloadDst = "/home/ipfs/java_cli/test/download/downloadRes.txt";
20
21 // 向node0上传
22 String uploadHashRes = fileOp.uploadFile(serverIP, serverPort0, uploadFilePath);
23 // System.out.println("Upload File CID is: " + uploadHashRes);
24 // 从node1下载
25 boolean ifDownload = fileOp.downloadFile(serverIP, serverPort1, downloadFileHash, c
26 System.out.println("Download Result: " + ifDownload);
27 }
```

The bottom console shows the execution output:

```
运行: Main
/usr/lib/jvm/java-1.21.0-openjdk-amd64/bin/java -javaagent:/snap/intellij-idea-community/515/lib/idea_rt.jar=36495:/snap/i
[INFO] Download finished! Path: /home/ipfs/java_cli/test/download/downloadRes.txt
Download Result: true
进程已结束,退出代码为 0
```

从node1下载刚才向node0上传的文件, 成功查找到CID, 下载成功



The screenshot shows a terminal window with the following commands and output:

```
ipfs@node0:~/ipfs-cluster-ctl$ cat /home/ipfs/java_cli/test/upload/uploadTest.tx
t
uploadtest
ipfs@node0:~/ipfs-cluster-ctl$ -ctl$ cat /home/ipfs/java_clidownload/downloadRes
.txttxt
uploadtest
123456ipfs@node0:~/ipfs-cluster-ctl$
```

可见内容确实是之前上传的文件内容