FISCO BCOS AMOP实操

参考

<https://fisco-bcos-documentation.readthedocs.io/zh_CN/release-2.7.0/docs/sdk/java_sdk/amop.html>

执行git clone https://github.com/FISCO-BCOS/java-sdk-demo

文本

描述已自动生成

切换分支

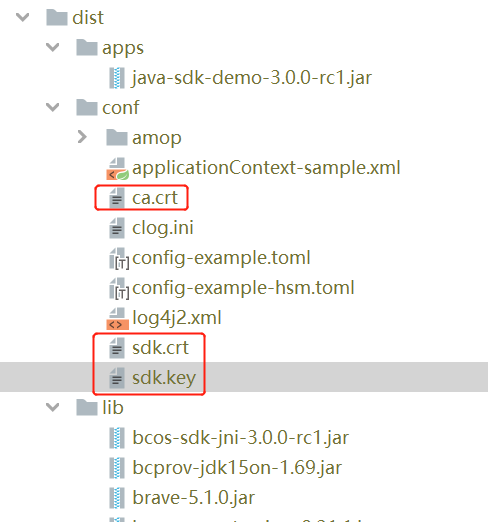
git checkout release-2.7.1

执行bash gradlew build或者用intellij打开工程进行build

搭建fisco bcos区块链，不赘述

配置

复制证书：将搭建FISCO BCOS网络节点nodes/${ip}/sdk/ 目录下的证书复制到java-sdk-demo/dist/conf目录下。



修改订阅者配置文件java-sdk-demo/dist/conf/amop/config-subscriber-for-test.toml，和发送者配置文件java-sdk-demo/dist/conf/amop/config-publisher-for-test.toml，修改配置文件中的节点信息。 注意：订阅者和发送者最好不连相同节点，如果连接了相同节点，则会被认为是同一个组织下的成员，私有话题无需认证即可通讯。这里我就简单化，连接相同节点。

图形用户界面, 应用程序

描述已自动生成

运行demo

公有话题Demo

打开两个终端，都进入dist目录，一个代表订阅者，一个代表发布者

运行订阅者

# 进入java-sdk-demo/dist目录

cd dist

# 使用第三节中所描述的工具

# 我们订阅名为”testTopic“的话题

java -cp apps/\*;lib/\*;conf/ org.fisco.bcos.sdk.demo.amop.tool.AmopSubscriber testTopic

运行发布者

cd dist

# 调用AmopPublisher发送AMOP消息

# 话题名：testTopic，是否广播：false(即使用单播)，内容：Tell you something， 发送次数：2次

java -cp "apps/\*;lib/\*;conf/" org.fisco.bcos.sdk.demo.amop.tool.AmopPublisher testTopic false "Tell you something" 2

运行结果

订阅者

图片包含 文本

描述已自动生成发布者

文本

低可信度描述已自动生成

发送文件

发送文件也是成功的，就不截图了。

对于发送文件的大小，经过实测，有如下结论：

本地验证AMOP订阅发布，结果如下：

1.1M的文件：秒级

2.60M：大概几秒

3.165M：10多秒钟

4.300M：报错，说文件太大或超时（超时时间我设置为5分钟）

5.1G：更不行了

感觉上了200M就不行了，一是可能传输失败，二是时间太长

看了源码，过程大概是

1.读取文件，把文件内容转换为byte[]存放在内存中，如果文件有1G就意味着这个byte[]有1G。这一步是底层的逻辑，基本改不了。

2.判断文件内容是否小于2^31-1 bytes，即2G多，如果小于，就OK，如果大于，就报错"file too big..."，这一步是工具类的逻辑，可以改这个大小

3.通过AMOP协议发送byte[]，点对点传输

所以瓶颈大概会发生在第1步，如果文件太大，内存就会撑爆。不是读一点传一点，是全部读出来，再传。这个机制是底层的，嵌套很深，基本改不了。所以，目前AMOP这种机制，不太适合传过大的文件，100M以内我感觉还可以，另外，如果文件过大，传输过久，也更容易遇到网络故障。所以，对于大文件的发送，必须分片发送。