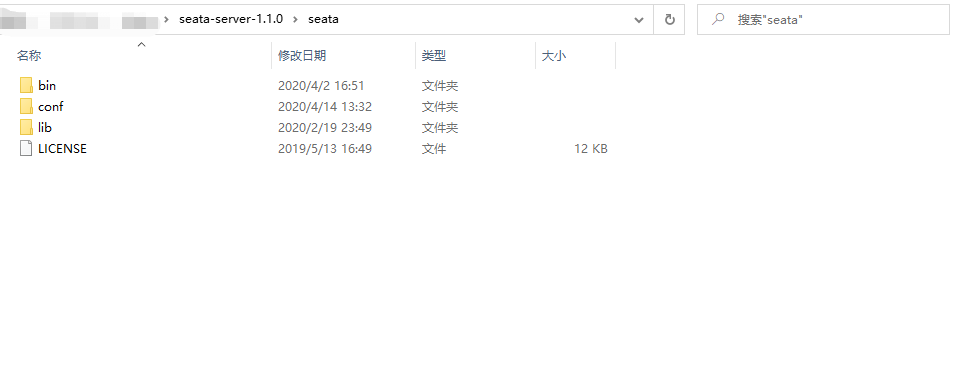
Seata使用指南

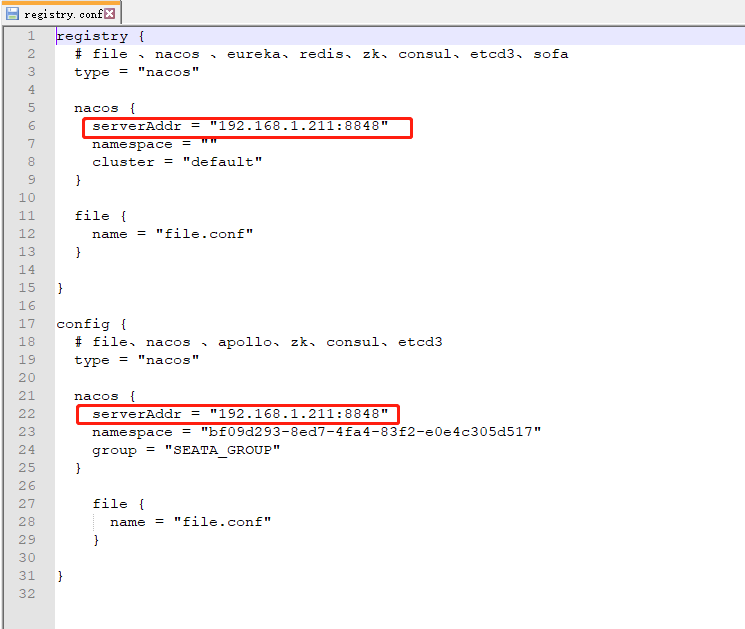
# 一、使用说明

Seata 分为Seata Server 和 client，Seata Server就是指分布式事务中的全局事务管理器（TC），因此想要使用分布式全局事务，先行启动Seata Server

1、解压seata-server-1.1.0.zip，解压后目录如下



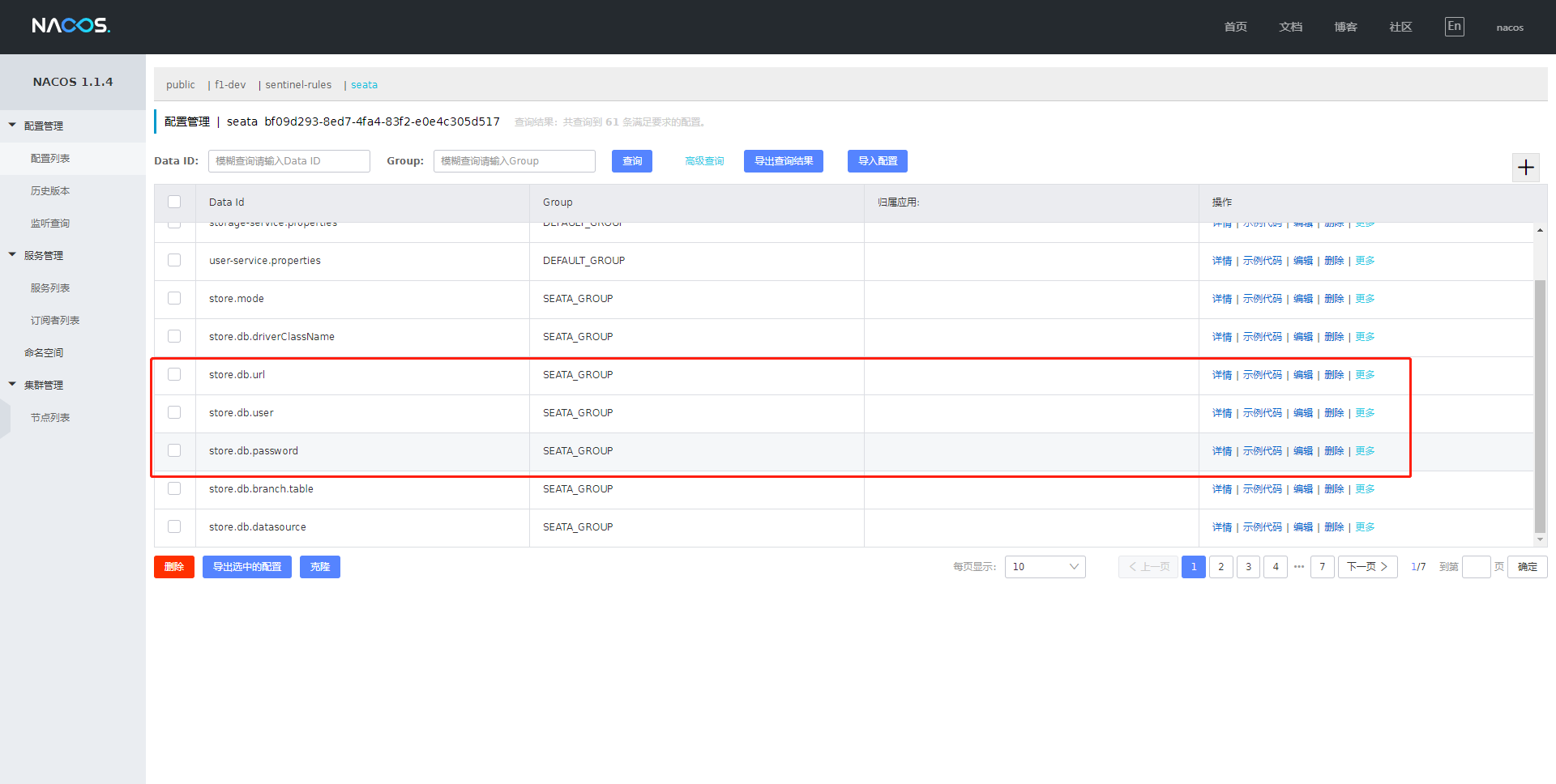
2、进入/conf 目录



需要修改registry.conf 文件中的Nacos 的地址和端口。

3、Nacos中修改配置

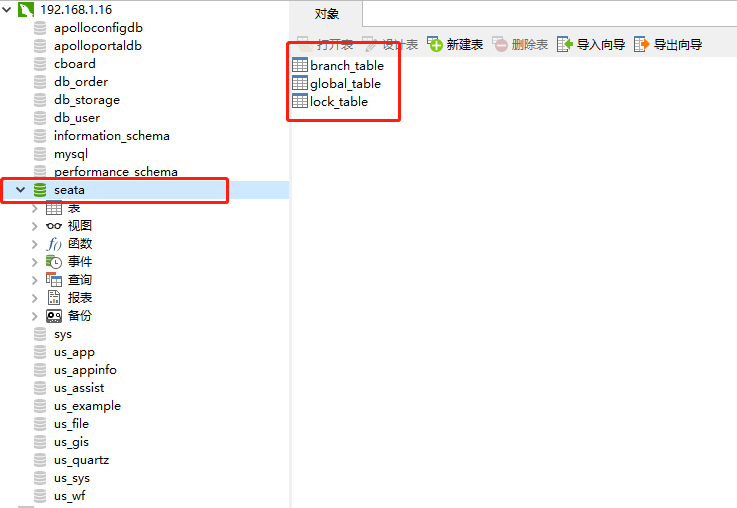
在Nacos配置列表中选择“seata”选项卡



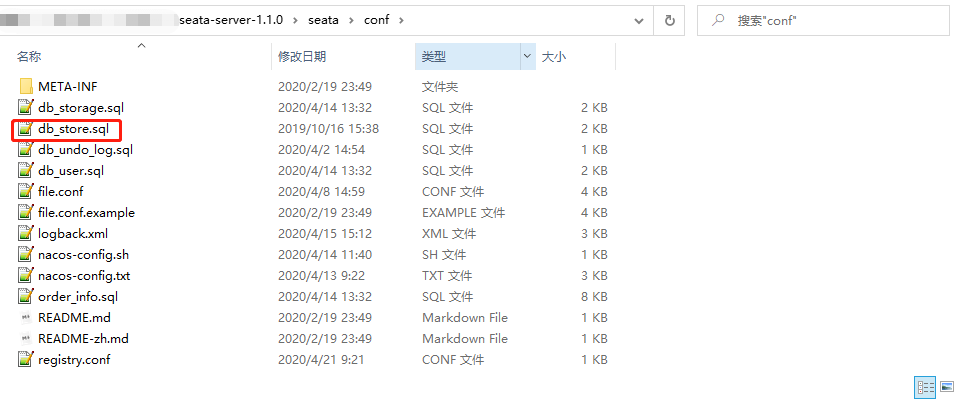
修改store.db.url、store.db.user、store.db.password三项，对应数据库URL、用户名和密码

4、进入 /bin 目录，根据所在环境执行对应的脚本

启动之前请确保数据库服务器存在Seata数据库



如果没有，需运行/conf目录下 db\_store.sql文件

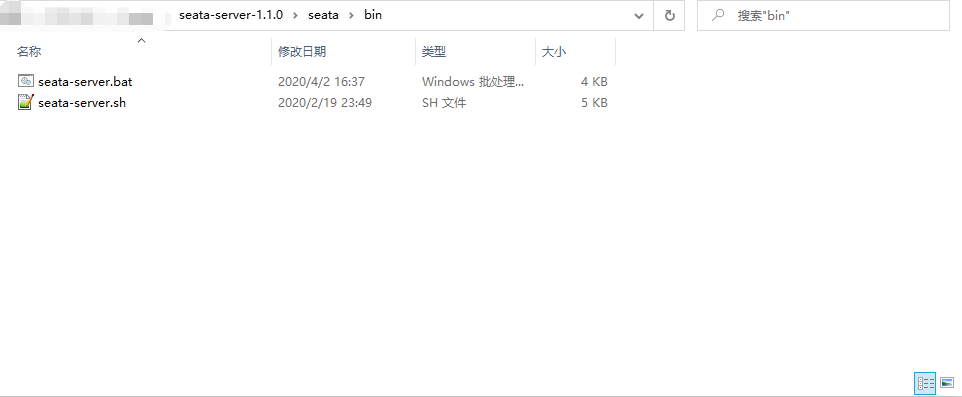


Linux运行命令:

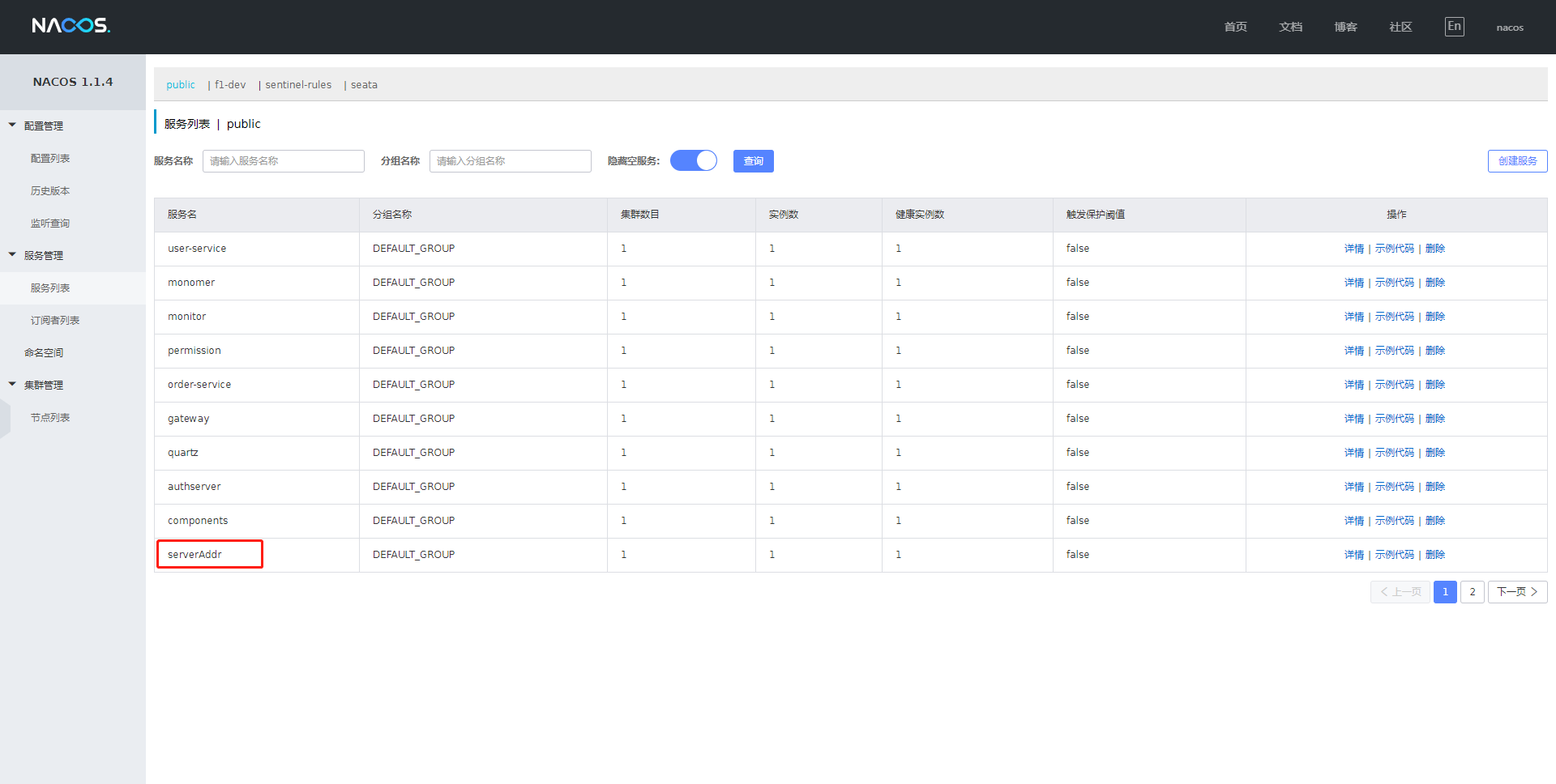
nohup sh seata-server.sh -p 8092 &

Windows运行命令：

seata-server.bat -p 8092



注册中心发现TC服务即为启动成功



5、引入jar包

<dependency>

<groupId>io.seata</groupId>

<artifactId>seata-spring-boot-starter</artifactId> </dependency>

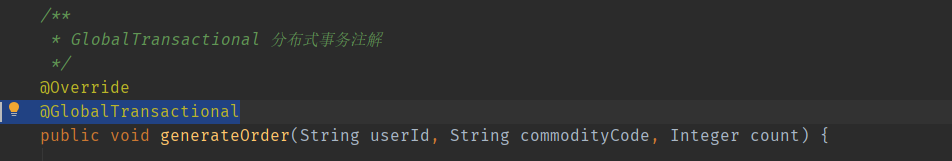
<dependency>

<groupId>io.seata</groupId>

<artifactId>seata-all</artifactId>

</dependency>

5、在需要开启全局事务的Controller 或者Service 类方法上加 @GlobalTransactional 注解



即可使用全局事务

# 二、注解说明

1、@GlobalTransactional

@GlobalTransactional注解用来开启全局事务，用在整个分布式事务发起方的业务方法上

2、@GlobalLock

@GlobalLock的作用是将当前方法对数据的修改加入Seata全局事务机制中，它对数据资源的操作需要先查询全局锁，如果存在其他 Seata 全局事务正在修改，则该方法也需等待，也就是说，如果想要Seata 全局事务执行期间，数据库不会被其他事务修改，则该方法需要强制添加 GlobalLock 注解，来将其纳入 Seata 分布式事务的管理范围。使用该注解的优点是可以有效确保数据准确，防止脏数据的出现；

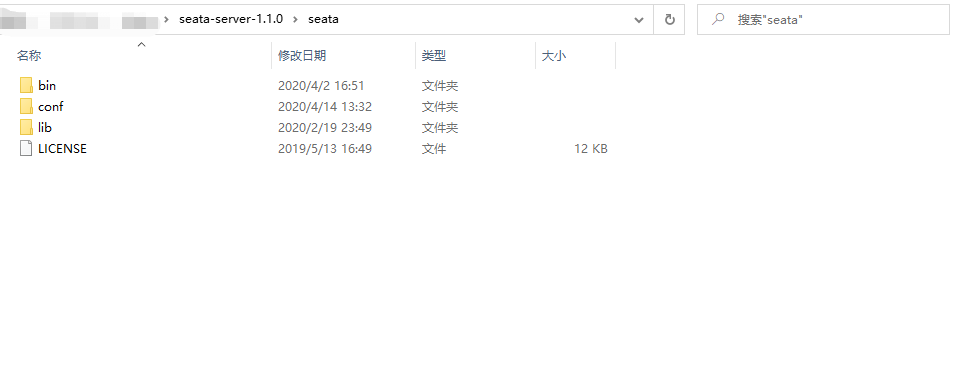
优点：可以有效确保数据准确，防止脏数据的出现

缺点：效率会受到影响

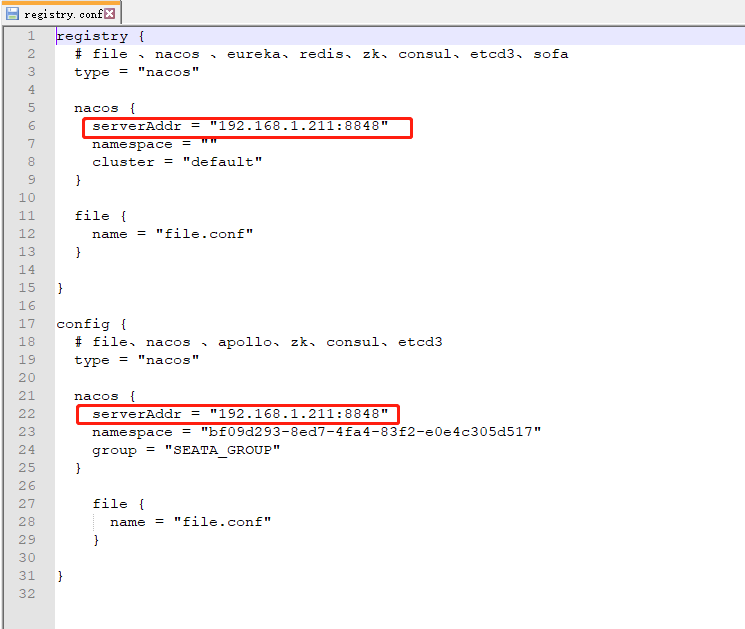
# 三、TC高可用部署说明

Seata 的TC可以通过在Nacos注册中心注册多个达到高可用的目的。

1、解压seata-server-1.1.0.zip，解压后目录如下



2、进入/conf 目录



修改registry.conf文件内容：

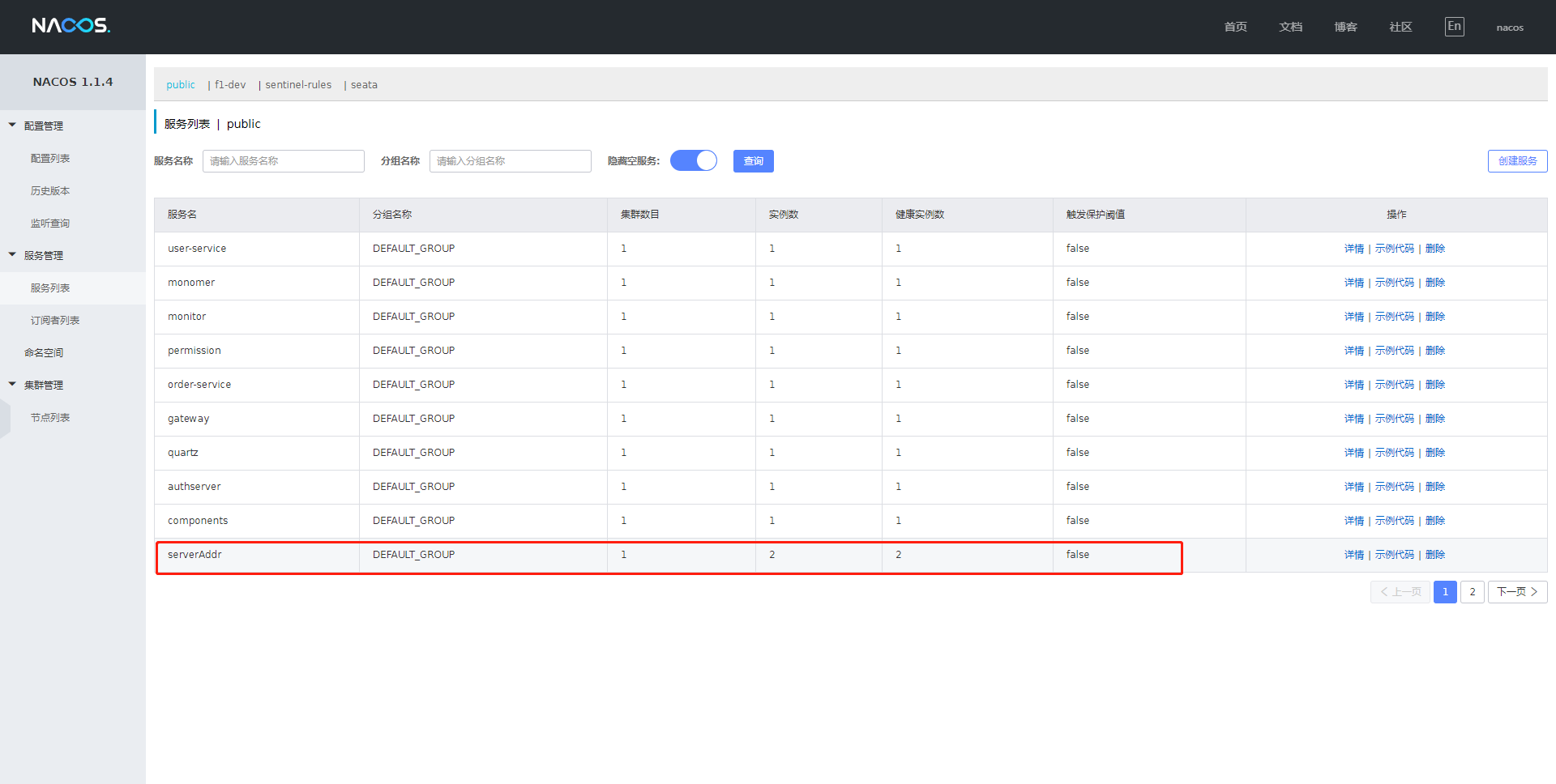
registry模块设置的是TC所要注册到的注册中心，如果需要高可用部署，需要将多个TC 配置 registry模块下的Nacos地址一致。Config模块设置的是TC 读取配置的地址。

Linux运行命令:

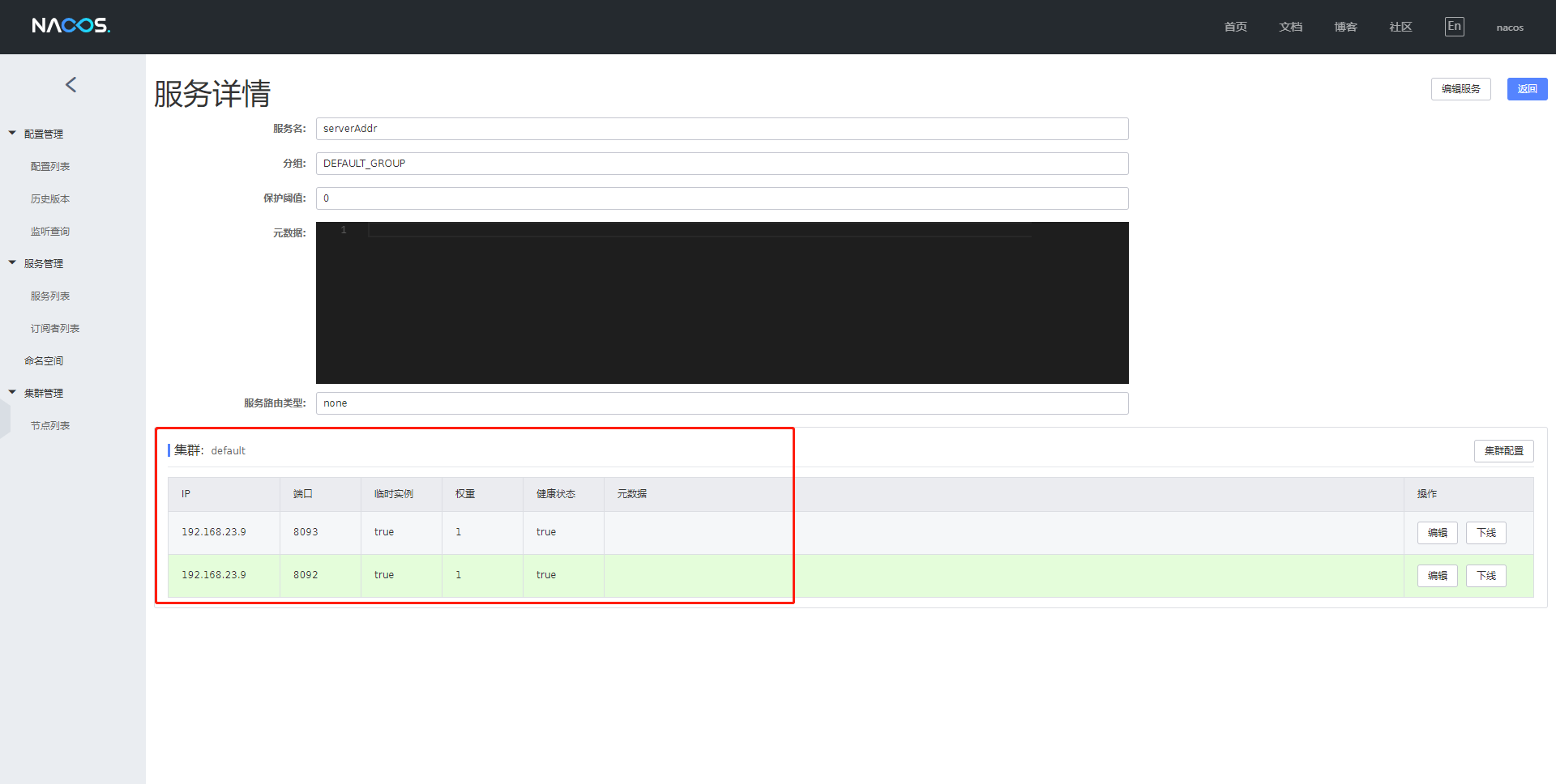
nohup sh seata-server.sh -p 8092 &

Windows运行命令：

seata-server.bat -p 8092



可以看到 serverAddr 服务的健康实例数变成了2



这样就完成了TC的高可用