

# 1. 部署前准备

前面的服务, 咱们都是在单机上部署的, 接下来我们使用多台服务器, 演示下分布式服务部署.

在部署之前, 我们先整理下需要部署的内容:

- 1. MySQL
- 2. Nacos
- 3. 网关服务
- 4. 产品服务
- 5. 订单服务

由于服务器受限, 我们把MySQL, Nacos, 网关服务部署为单例的, 产品服务和订单服务部署为多例的.

## 环境要求

机器个数:

1-N台均可.

如果为1台, 参考前面课程, 所有的项目均部署在一台机器即可.

如果为多台, 把上面5个服务, 根据机器性能分摊, 分开部署即可.

为更好的学习, 建议3台以上

机器环境:

Linux环境(CentOS, Ubuntu均可)

## 本课程机器情况说明:

机器个数: 4台

服务部署分配如下

服务器	计划部署内容
服务器1	MySQL, Nacos
服务器2	产品服务实例1, 订单服务实例1
服务器3	产品服务实例2, 订单服务实例2

## 2. 部署操作

### 2.1 MySQL安装

#### 1. MySQL 安装

参考 [📖 MySQL安装](#)

#### 2. 数据初始化

参考 [📖 2. 环境和项目搭建](#) 中的[数据准备](#)章节

#### 3. 对其他服务器授权

MySQL默认情况下, 只允许本地连接, 即localhost, 如果其他服务器需要连接到MySQL, 需要MySQL对这个服务器授权

语法格式:

```
1 grant 权限 on 数据库对象 to 用户
```

#### 1. 使用下面SQL, 创建用户, 并授权

```
1 -- 创建用户bite, 并设置密码, 此步可省略
2 CREATE USER 'bite'@'%' IDENTIFIED BY 'BITE@yyds.666';
3 -- 对bite用户授权
4 -- *.* 表示所有库的所有表, 也可以指定库和表
5 -- %表示IP, %表示允许所有IP访问, 也可以指定IP
6 GRANT ALL ON *.* TO 'bite'@'%';
7 -- 让修改生效
8 FLUSH PRIVILEGES;
```

#### 2. 修改bind-address

修改文件路径: /etc/mysql/mysql.conf.d/mysqld.cnf

把bind-address = 127.0.0.1改为 bind-address = 0.0.0.0

```
1 #bind-address                = 127.0.0.1
2 bind-address                  = 0.0.0.0
```

### 3. 重启MySQL服务器

```
1 sudo systemctl restart mysql
```

### 4. 开放3306端口号

### 5. 测试授权结果

使用CMD客户端, 连接服务器MySQL, 如果可以正确连接, 则授权成功

```
1 #把110.41.51.65改成自己服务器的IP -u改成设置的账号名 -p后是对应的密码
2 mysql -h110.41.51.65 -P3306 -ubite -pBITE@yyds.666
```

```
C:\Users\lucf>mysql -h110.41.51.65 -P3306 -ubite -pBITE@yyds.666
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

## 2.2 Nacos 安装

参考 [目5. 注册中心的其他实现-Nacos](#) 中的**Nacos安装**章节

安装前需要安装JDK

## 2.3 产品服务, 订单服务部署

两台服务器操作步骤一样

### 1. 安装JDK.

参考 [目JDK17安装](#)

### 2. 登录服务器, 上传Jar包

```
1 [root@VM-0-17-centos spring-cloud]# ls
2 order-service-1.0-SNAPSHOT.jar  product-service-1.0-SNAPSHOT.jar
```

### 3. 启动服务

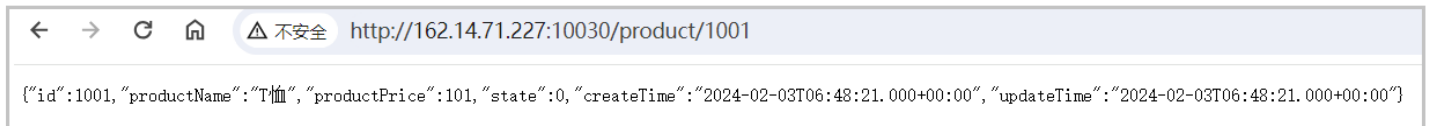
```
1 #后台启动product-service, 并设置输出日志到logs/order.log
2 nohup java -jar product-service-1.0-SNAPSHOT.jar --
  spring.cloud.nacos.discovery.ip=162.14.71.227 >logs/product.log &
3
4 #后台启动order-service, 并设置输出日志到logs/order.log
5 nohup java -jar order-service-1.0-SNAPSHOT.jar --
  spring.cloud.nacos.discovery.ip=162.14.71.227 >logs/order.log &
```

Nacos 默认注册地址为内网IP, 如果各个服务器不在一个局域网时, 根据内网IP去访问时, 会出现网络不通, 接口调用失败, 需要通过`spring.cloud.nacos.discovery.ip` 设置注册的地址

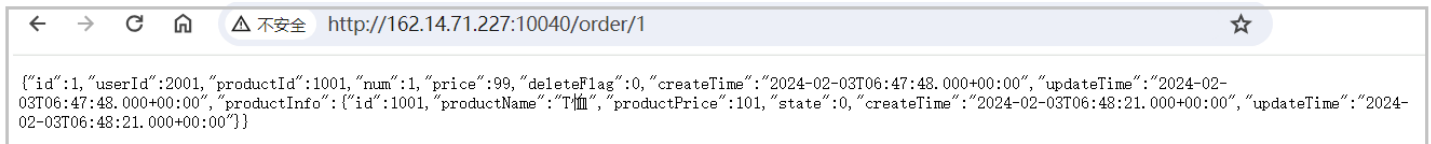
### 4. 开放对应端口号

### 5. 测试

产品服务: <http://162.14.71.227:10030/product/1001>



订单服务: <http://162.14.71.227:10040/order/1>



## 2.4 网关服务部署

### 1. 安装JDK.

参考 [📖JDK17安装](#)

### 2. 登录服务器, 上传Jar包

```
1 [root@VM-0-17-centos spring-cloud]# ls
2 gateway-1.0-SNAPSHOT.jar  logs
```

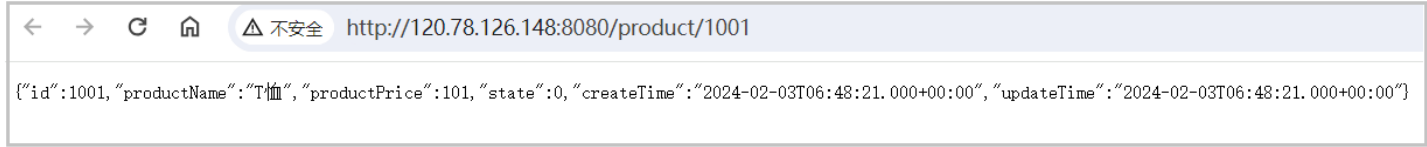
### 3. 启动服务

```
1 #后台启动gateway, 并设置输出日志到logs/gateway.log
2 nohup java -jar gateway-1.0-SNAPSHOT.jar --server.port=8080 >logs/gateway.log &
```

3. 开放对应端口号

4. 测试

产品服务: <http://120.78.126.148:8080/product/1001>



订单服务: <http://120.78.126.148:8080/order/1>

