

租个房上线环境及演练

运行环境

环境介绍

操作系统采用linux centos7

服务	IP	账号	备注	责任人
MySQL	192.168.100.240:3306	root/root	关系型数据库，5.7版本。	hash
Redis	192.168.100.241:6379	密码： redis123456	分布式缓存中间件，版本5.0.3	hash
Kafka	192.168.100.249:9092		消息中间件，版本2.1.0	hash
Elasticsearch	192.168.100.120:9300 192.168.100.120:9200		搜索引擎，版本6.4，附带对应的ik分词器插件	hash
Kibana	192.168.100.120:5601		ElasticSearch仪表盘，版本6.4	hash
Zookeeper	192.168.100.249:2181		分布式协调服务，版本3.4.13	hash
DubboAdmin	192.168.100.249:7001	root/root	Dubbo服务治理工具，版本2.0.2 incubator-dubbo-ops	hash
Nginx	192.168.100.120:80 192.168.100.121:80		用来做web应用的反向代理服务，版本1.16.0	hash
LVS/Keepalived	192.168.100.240（主） 192.168.100.241（从）	主从的虚拟IP为 192.168.100.200	LVS搭建Nginx集群，Keepalived用来保证两个LVS主从高可用	hash
rental-web	192.168.100.120:8088 192.168.100.121:8088	myAdmin/admin hash/hash	租个房web应用，浏览器提供服务，作为Dubbo服务消费者	hash
rental-uc	192.168.100.121:20880 192.168.100.122:20880		租个房用户中心 Dubbo服务提供则	hash
rental-house	192.168.100.121:8085:20885 192.168.100.122:8085:20885		租个房房源Dubbo服务提供者。8085 hessian协议服务；20885 dubb协议服务。	hash

注意对应的端口号畅通

配置

配置文件按照环境进行配备，开发、生产各一套。生产具体配置参考上面的环境表格，自己搭建一套环境，参考项目中的 application-online.properties 文件，进行配置。

打包

pom文件调整

各环境配置文件配备

日志文件配置调整

进入项目根目录，通过maven命令进行打包。

```
mvn clean package -DskipTests=true
```

运行

启动用户中心

```
nohup java -jar rental-uc-provider-0.0.1.jar --dubbo.protocol.port=20880 --  
dubbo.protocol.host=192.168.100.121 --spring.profiles.active=online > rental-  
uc.log 2>&1 &
```

启动房源服务

```
nohup java -jar rental-house-provider-0.0.1.jar --server.port=8085 --  
dubbo.protocols.dubbo.port=20885 --dubbo.protocols.dubbo.host=192.168.100.121 --  
dubbo.protocols.hessian.host=192.168.100.121 --spring.profiles.active=online >  
rental-house.log 2>&1 &
```

启动web应用

```
nohup java -jar rental-web-0.0.1.jar --server.port=8088 --  
spring.profiles.active=online > rental-web.log 2>&1 &
```

启动可能碰到的问题

mysql连接报错

```
The last packet sent successfully to the server was 0 milliseconds ago. The  
driver has not received any packets from the server.
```

表示rental系统向mysql成功发送了数据包，但是没有收到mysql对应的响应数据包。造成这个原因的可能是网络造成的，比如有两个网卡在运行。另外也可能是数据库的版本兼容的问题。

kafka升级到2.1.0报错

```
Topic(s) [house_build] is/are not present and missingTopicsFatal is true
```

高版本Kafka需要先定义topic，那么我们可以在项目启动的时候，通过api来建立对应的topic即可。

```
// 参考这个配置类来建立topic
edu.dongnao.rental.house.provider.config.KafkaConfig
```

故障演练

单机故障

Web应用单机故障

可以通过Nginx来进行反向代理，搭建集群服务

Nginx故障

单机故障

LVS 搭建Nginx集群

```
# 192.168.100.240 192.168.100.241上，执行lvs直接路由设置
sudo sh lvs_dr.sh start 192.168.100.200 80 "192.168.100.120 192.168.100.121"
ens192:0
# 停止虚拟IP
sudo sh lvs_dr.sh stop ens192:0

# 192.168.100.120 192.168.100.121上 执行Nginx服务虚拟地址
sudo sh lvs_rs.sh start 192.168.100.200 10:0
# 停止
sudo sh lvs_rs.sh stop 10:0
```

Keepalived 保证LVS/Nginx高可用

```
# Nginx主
sudo /usr/sbin/keepalived -f /etc/keepalived/rental-keepalived-lvs-master.conf
# Nginx备
sudo /usr/sbin/keepalived -f /etc/keepalived/rental-keepalived-lvs-backup.conf
```

服务雪崩

能不能进行一个后端服务访问的控制

Nginx限流配置介绍

Dubbo服务治理

Dubbo服务故障如何处理？利用DubboAdmin进行操作

服务资源隔离

动态配置

访问控制

黑白名单

访问控制

服务版本迭代

优雅停机

Dubbo提供的优雅停机机制

web容器优雅停机，通过servlet注销方法实现

灰度发布

服务器级别：DubboAdmin，禁用服务方式，基于IP地址

用户级别：通过自定义的配置文件来配置部分用户，DubboAdmin动态配置，指定参数名