1. 前期准备工作
   1. mysql 5.6+
   2. java1.8+
   3. 机器: 192.168.31.52 pro环境：configService，adminService

192.168.190.129 dev环境：configService，adminService

192.168.190.128 portal

1. 数据库脚本
   1. apolloportaldb:是前端页面的库，自始至终只有一个库， 无论多少套环境，脚本所在位置/apollo/scripts/sql/apolloportaldb.sql
   2. apolloconfigdb:是每一套环境单独的库，有多少套环境就有多少个库（dev,uat,pro,正常三套）/apollo/scripts/sql/ apolloconfigdb.sql
2. 打包
   1. 修改配置L：/apollo/scripts/build.bat

3.1.2 为dev环境打包

#设置config\_dev库的数据库连接信息

rem apollo config db info

set apollo\_config\_db\_url="jdbc:mysql://192.168.31.52:3306/apolloconfigdb\_dev?characterEncoding=utf8"

set apollo\_config\_db\_username="root"

set apollo\_config\_db\_password="Root123.."

#设置portal库的数据库连接信息

rem apollo portal db info

set apollo\_portal\_db\_url="jdbc:mysql://192.168.31.52:3306/apolloportaldb?characterEncoding=utf8"

set apollo\_portal\_db\_username="root"

set apollo\_portal\_db\_password="Root123.."

#设置apollo\_meta的url，其实就是configservice的url,因为configservice自身就有eureka注册中心的作用，而apollo\_meta

#就是eureka注册中心

rem meta server url, different environments should have different meta server addresses

set dev\_meta="http://192.168.190.129:8080"

set pro\_meta=<http://192.168.31.52:8080>

3.1.3 执行build.bat,生成configService，adminService，portal对应的包

3.1.4 登录 192.168.190.128机器

#切换到root用户

su root

#创建/opt/logs文件夹

mkdir –p /opt/logs

#将/opt/logs文件的权限交给dev1

chown –R dev1 /opt/logs

#进入/usr/local文件夹下

cd /usr/local

#创建 apollo/portal文件夹

mkdir -p apollo/portal

#将apollo，portal文件的权限交给dev1

chown –R dev1 ./apollo/portal

cd ./apollo/portal

#切换到dev1用户

su dev1

#将portal打出来的包apollo-portal-0.9.1-github.zip放到此处,并解压

unzip apollo-portal-0.9.1-github.zip

3.1.5 登录 192.168.190.129机器

#切换到root用户

su root

#创建/opt/logs文件夹

mkdir –p /opt/logs

#将/opt/logs文件的权限交给dev1

chown –R dev1 /opt/logs

#进入/usr/local文件夹下

cd /usr/local

#创建 apollo/configService文件夹

mkdir -p apollo/configService

#将apollo/configService文件的权限交给dev1

chown –R dev1 ./apollo/configService

#创建 apollo/adminService文件夹

mkdir -p apollo/adminService

#将apollo/adminService文件的权限交给dev1

chown –R dev1 ./apollo/adminService

cd ./apollo/configService

#切换到dev1用户

su dev1

#将configService打出来的包apollo-configService-0.9.1-github.zip放到此处,并解压

unzip apollo-configService-0.9.1-github.zip

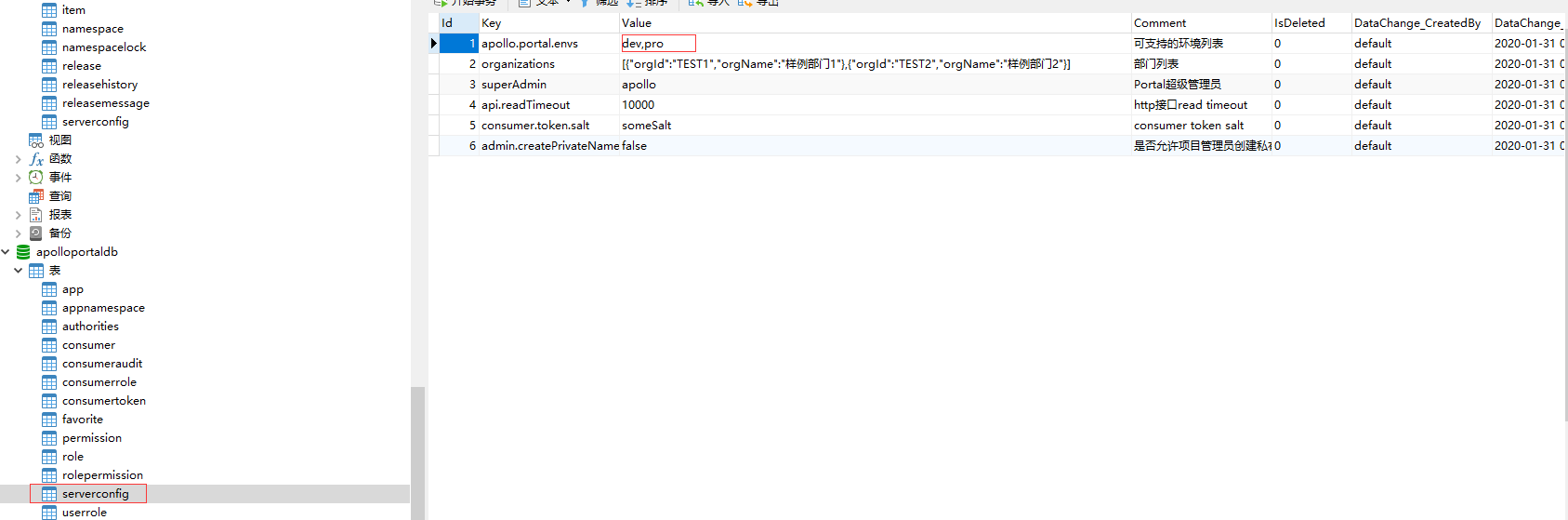
cd ./apollo/adminService

#将adminService打出来的包apollo-adminService -0.9.1-github.zip放到此处,并解压

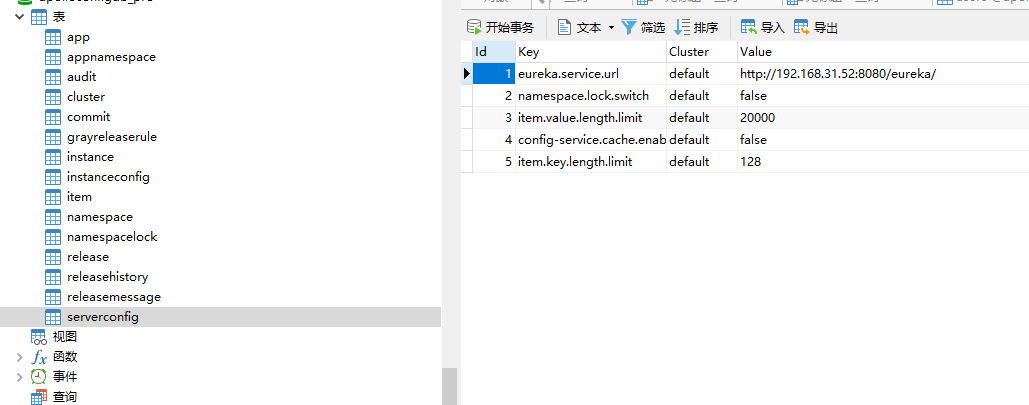
unzip apollo- adminService -0.9.1-github.zip

3.1.6 修改数据库配置

修改portal库中的serviceconfig表中的apollo.portal.envs,有几套环境写几套（dev，uat，pro）



修改configdb中的serviceconfig中的eureka.service.url对应的value改成该套环境对应的configservice



3.1.7 启动应用，启动顺序config-🡪admin🡪portal

登录 192.168.190.129机器

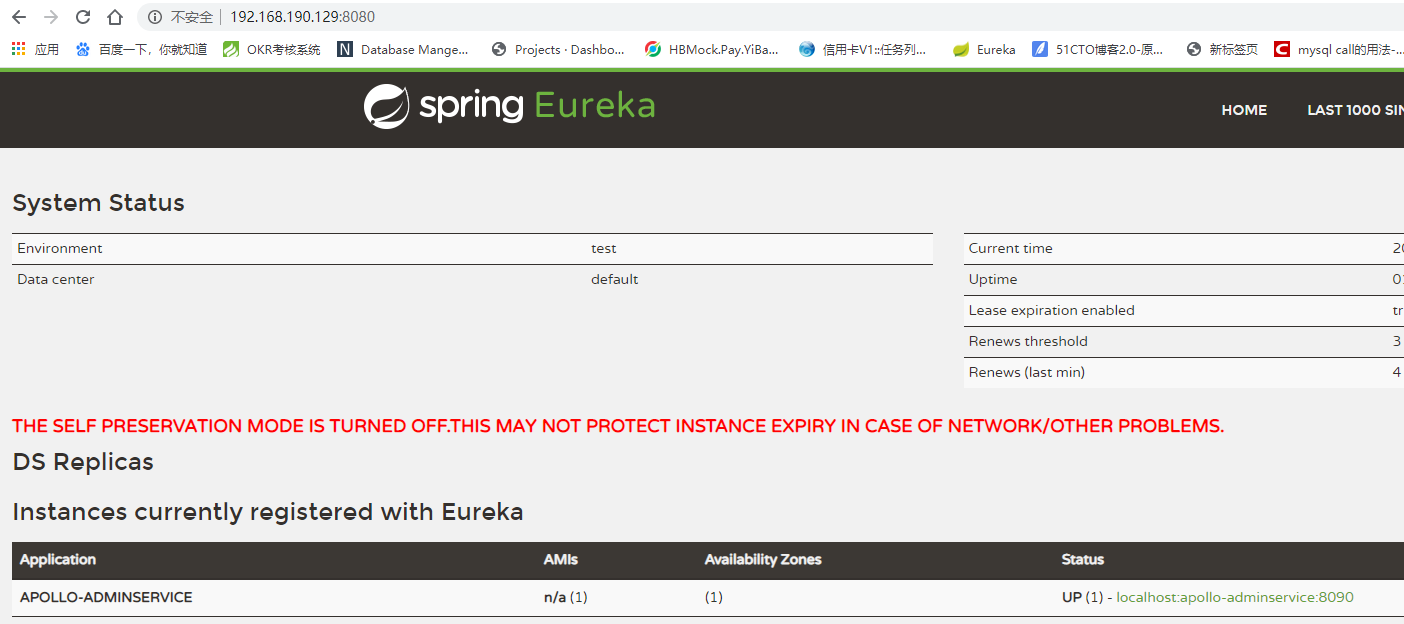
##启动configservice

/usr/local/apollo/configService/script/start.sh

##打开8080端口

firewall-cmd --zone=public --add-port=8080/tcp

验证服务是否正常



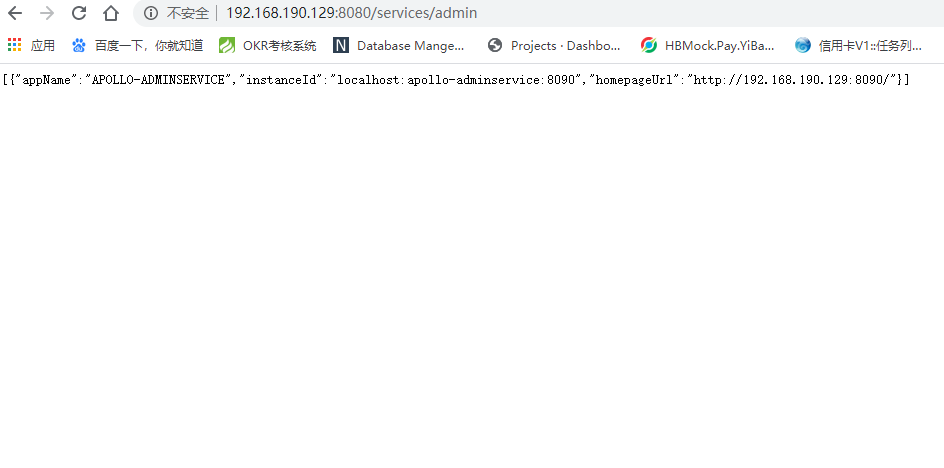
##启动adminservice

/usr/local/apollo/adminService/script/start.sh

##打开8090端口

firewall-cmd --zone=public --add-port=8090/tcp

验证服务是否正常



登录 192.168.190.128机器

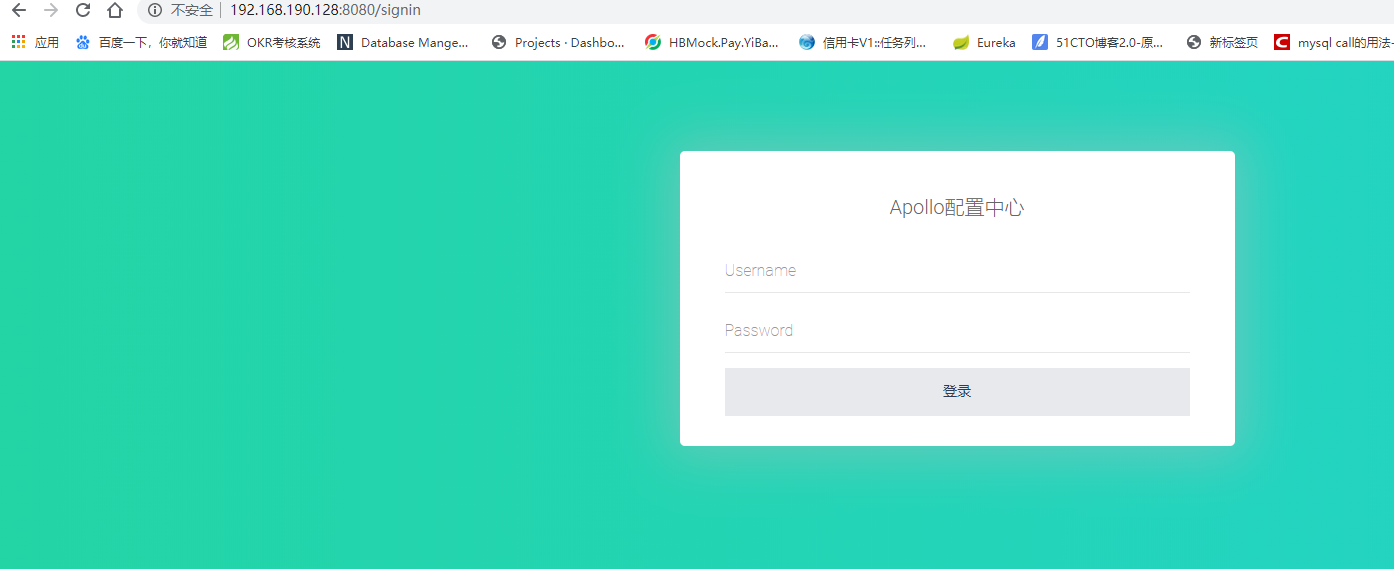
##启动portal

/usr/local/apollo/portal/script/start.sh

##打开8080端口

firewall-cmd --zone=public --add-port=8080/tcp

验证服务



初始化用户名/密码:apollo/admin

创建项目。

项目启动测试：

配置启动参数：-Dapollo.meta=http://192.168.190.129:8080 -Dapp.id=sc-client1-ms