上篇文章介绍了kong的基本概念，那么如何管理kong呢？本篇文章介绍konga的使用

**准备**

在开始本篇教程前，我们先做如下准备工作

1. 添加一个测试服务器

# docker run -itd --name example \

--network kongnet \

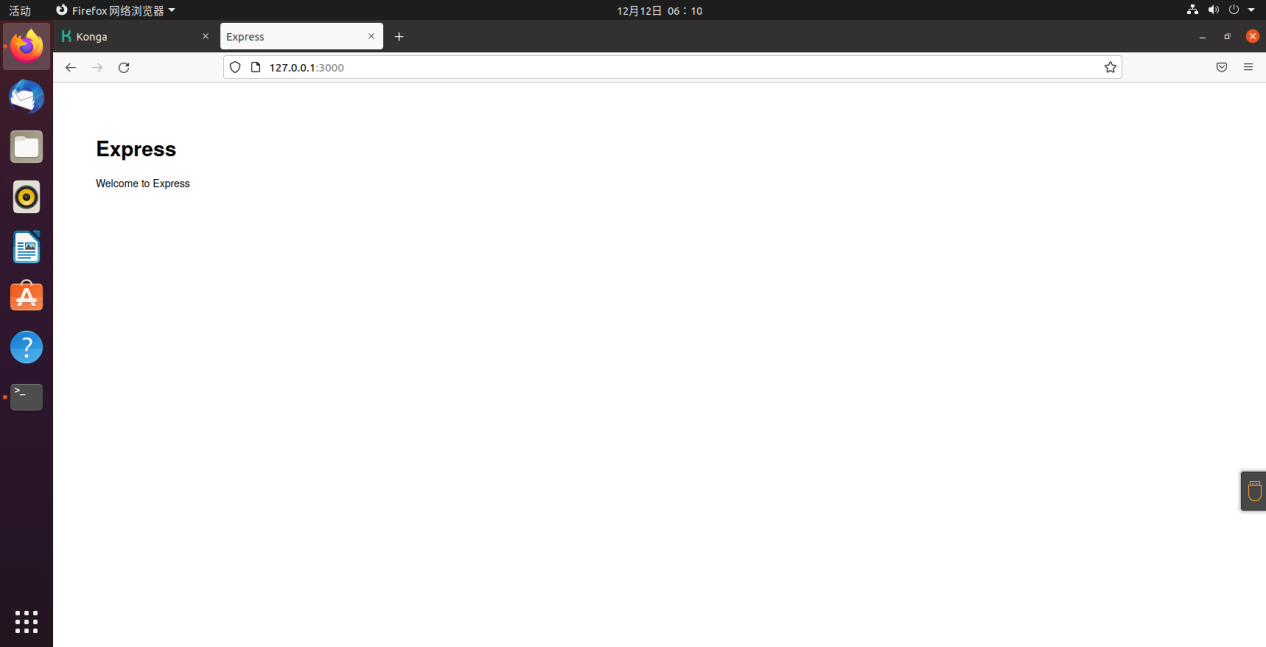
--restart=always \

-p 3000:3000 \

bitnami/node-example:0.0.1

该测试服务的名称叫做example，端口为3000

我们访问3000端口可以看到如下页面



**Konga**

konga是一个web站点（nodejs）,通过调用Kong Admin API管理kong

**创建数据库**

Konga需要一个数据库，执行如下命令创建

1. 进入容器：docker exec -it postgres /bin/bash
2. 进入postgres：psql -U postgres
3. 创建数据库：CREATE DATABASE konga OWNER kong;
4. 推出数据库：\q ，推出容器：exit

**安装konga**

1. 执行如下命令安装konga

# docker run -d --name konga \

--network kongnet \

--restart=always \

-p 1337:1337 \

-e "DB\_ADAPTER=postgres" \

-e "DB\_HOST=postgres" \

-e "DB\_PORT=5432" \

-e "DB\_USER=kong" \

-e "DB\_PASSWORD=kong123" \

-e "DB\_DATABASE=konga" \

-e "NODE\_ENV=development" \

docker.io/pantsel/konga:0.14.7

1. 说明：

-e "DB\_ADAPTER=postgres"：指定数据空类型为postgresql

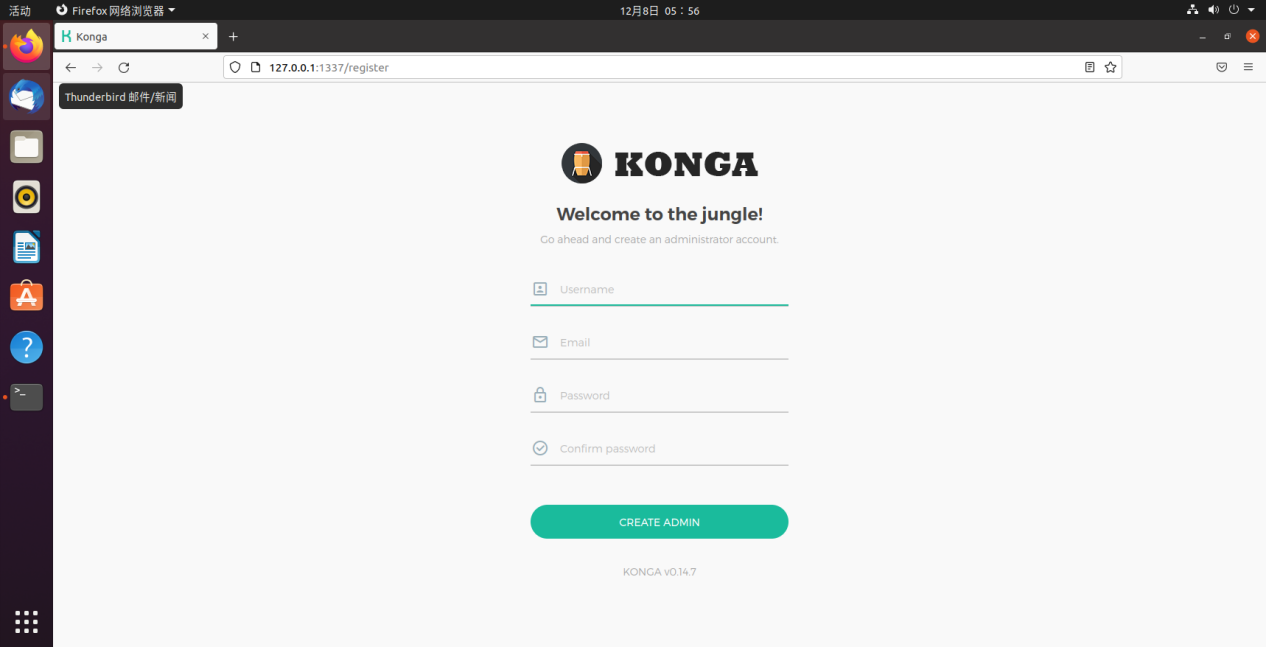
-e "DB\_HOST=postgres"：

指定数据库的主机（在kongnet中我们数据库服务器的名称就是postgres）

-e "DB\_DATABASE=konga"：指定数据库名称

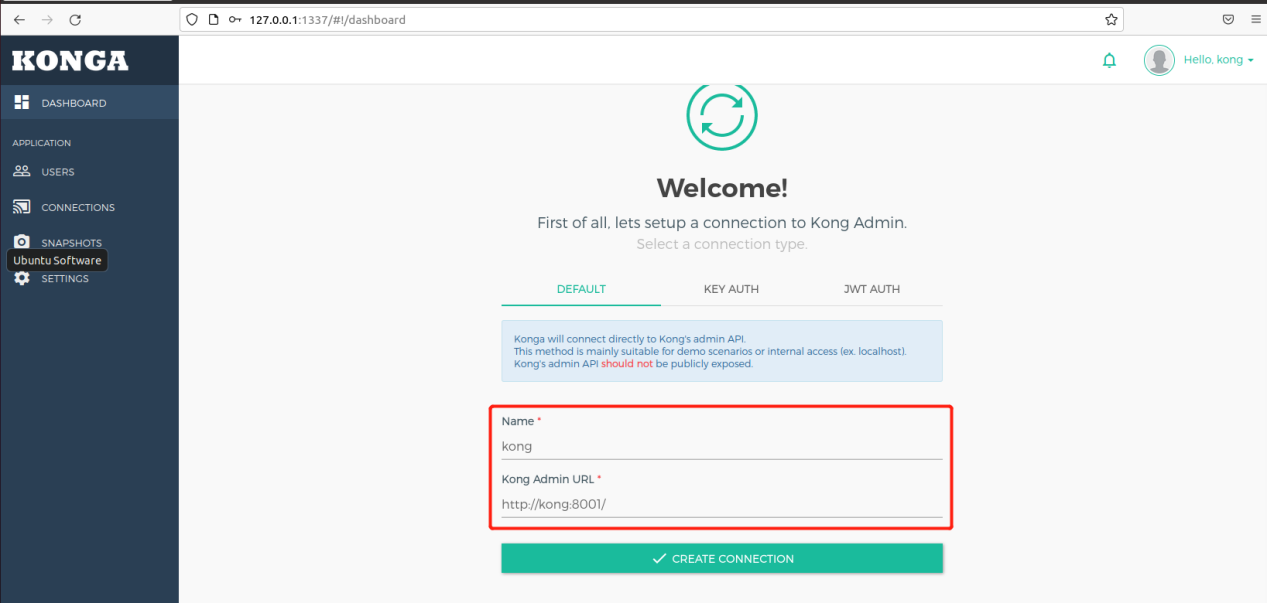
1. 查看konga：

我们访问1337端口即可看到konga的界面，安装指示，你可以创建一个管理员（我们之后使用的密码都是kong/kong123）

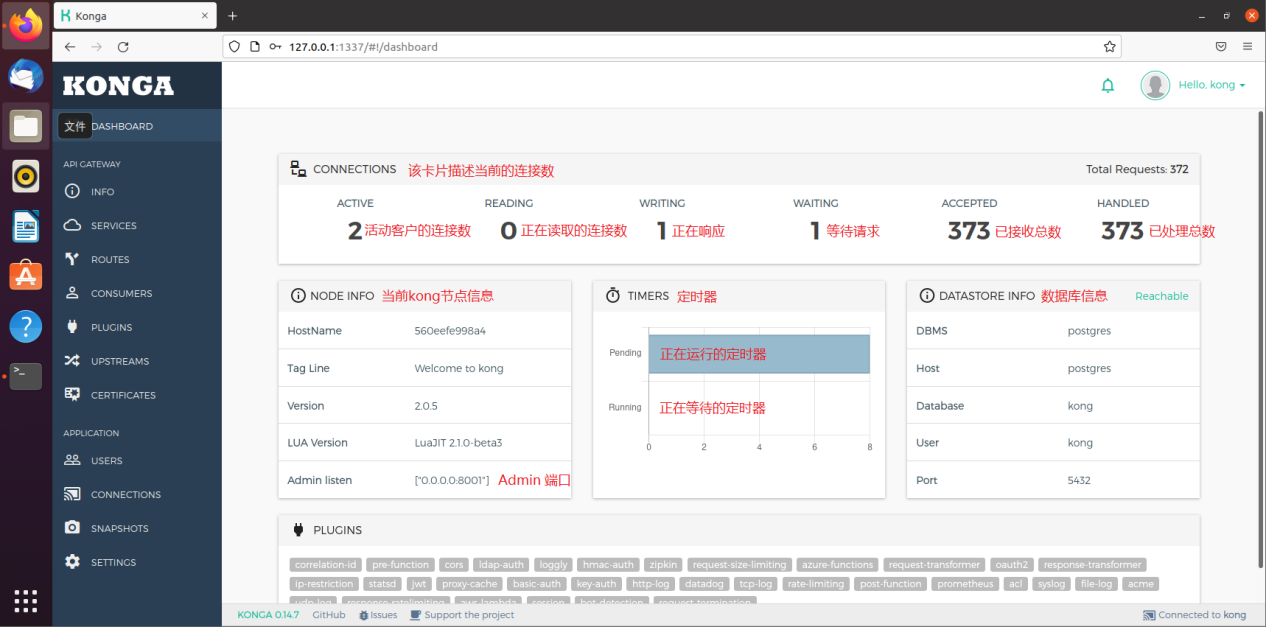


**将konga连接到kong**

进入到后台首页，我们填写kong的url即可（我们kong的容器叫做kong，在kongnet网段内使用http://kong:8001是可以访问到kong容器的），name随便填，

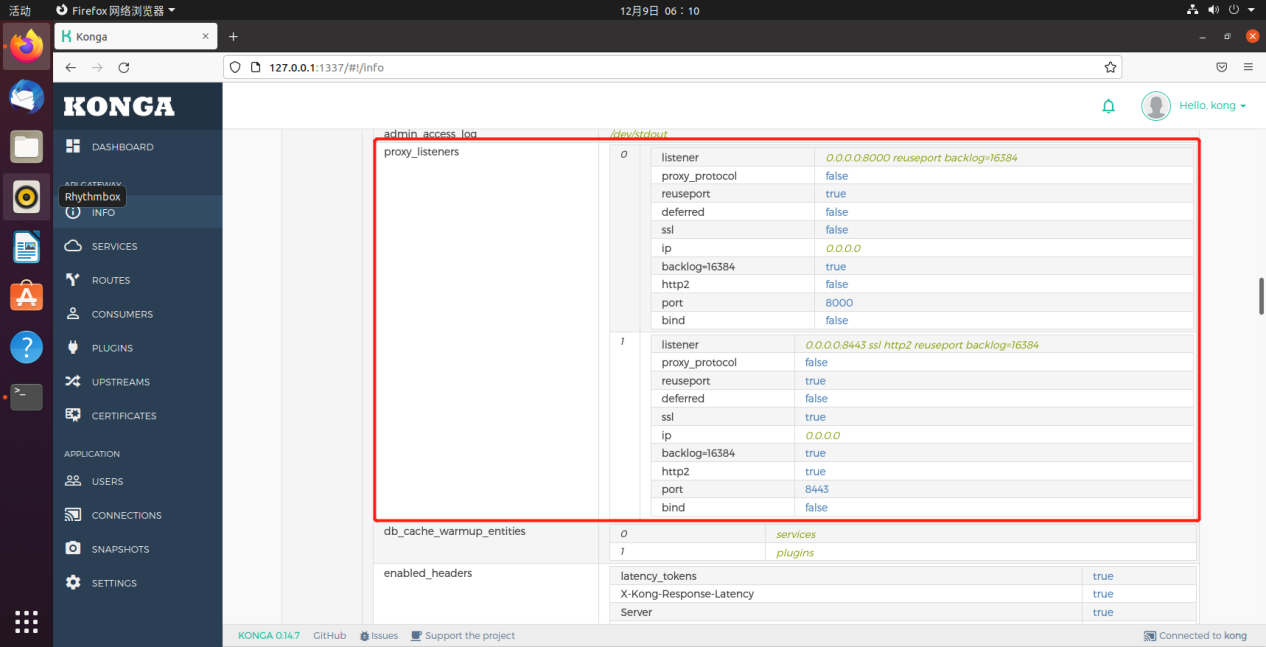


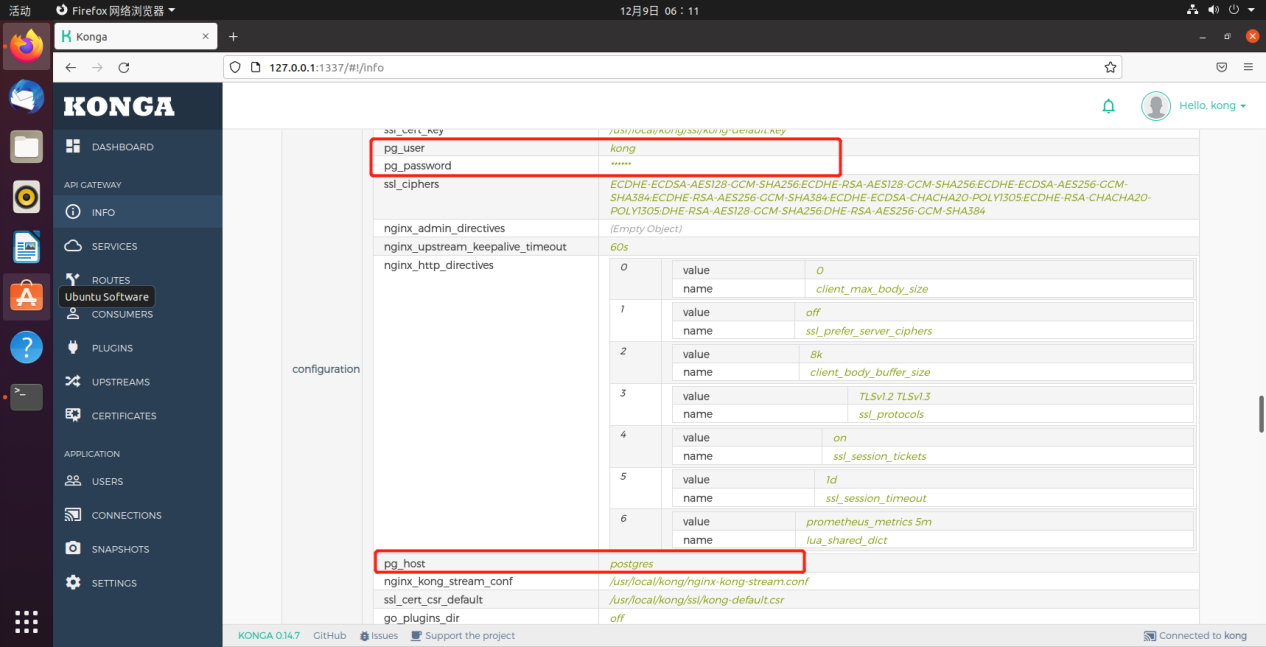
**konga仪表盘说明**



**INFO配置信息**

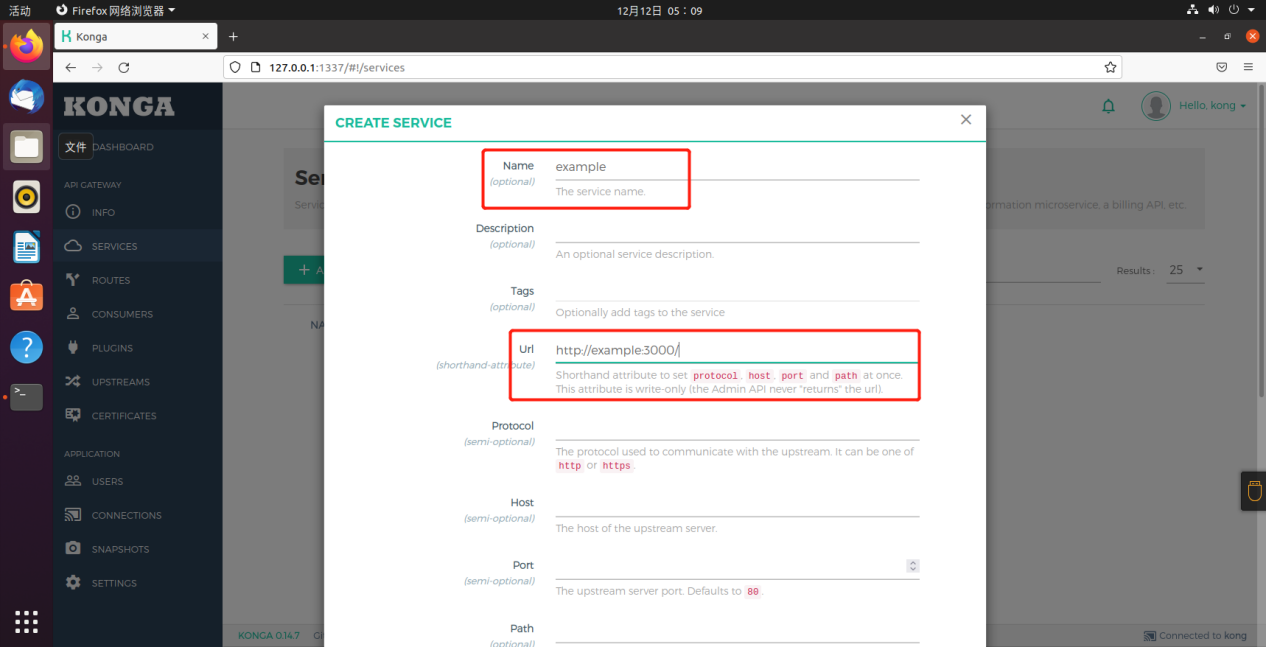
INFO展示了当前kong节点的所有配置信息，如下一些常见的配置信息：





**服务**

如下，我们可以创建一个服务



name：服务名称

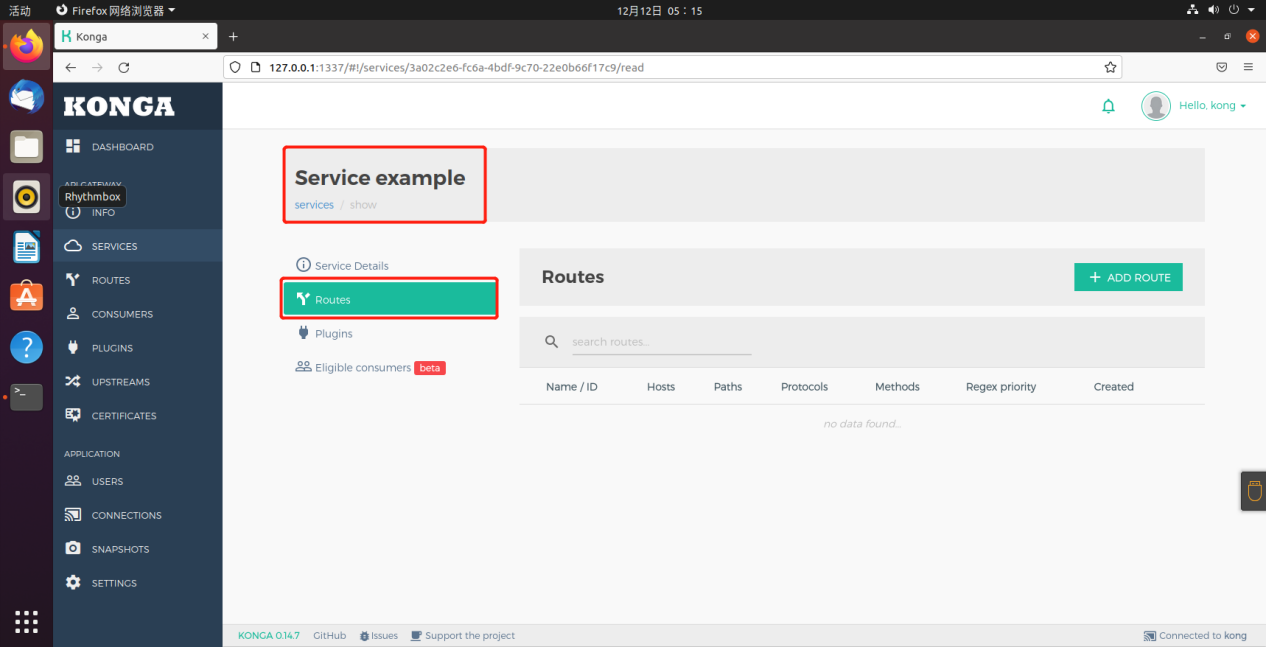
url：上游节点的url

一般情况下我们只需要填写这2个参数即可

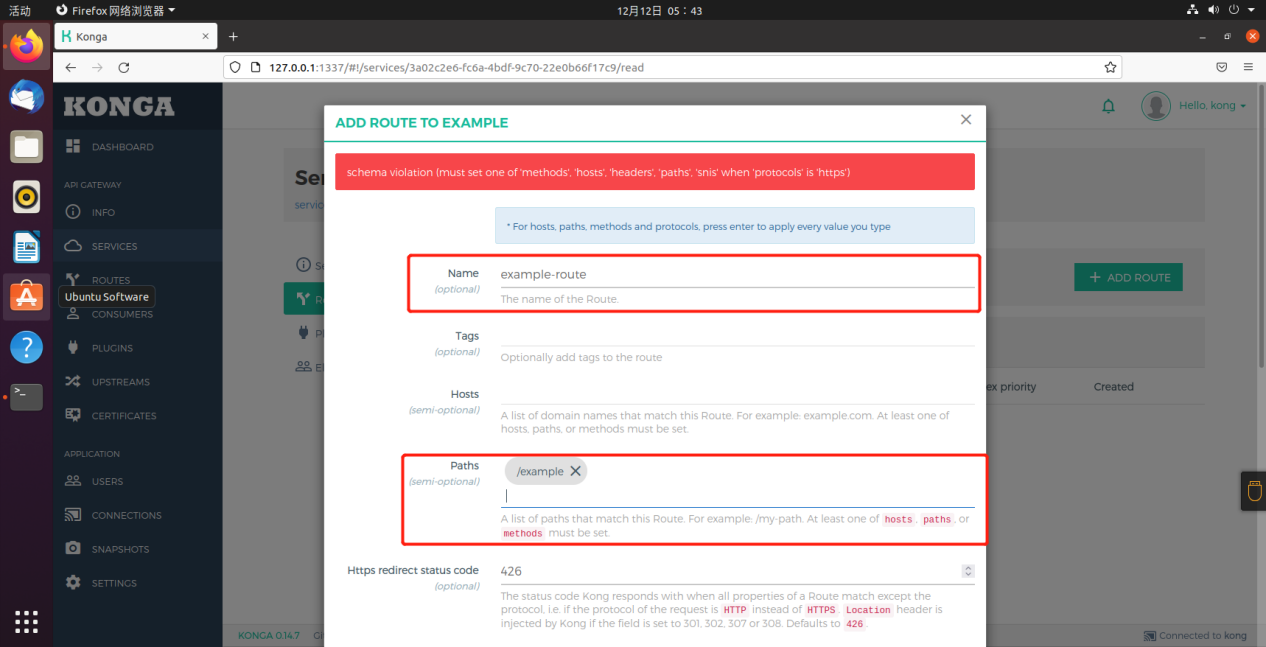
注：<http://example:3000/> 就是我们添加的服务器

**路由**

点击进入服务详情页面，我们可以为服务添加路由



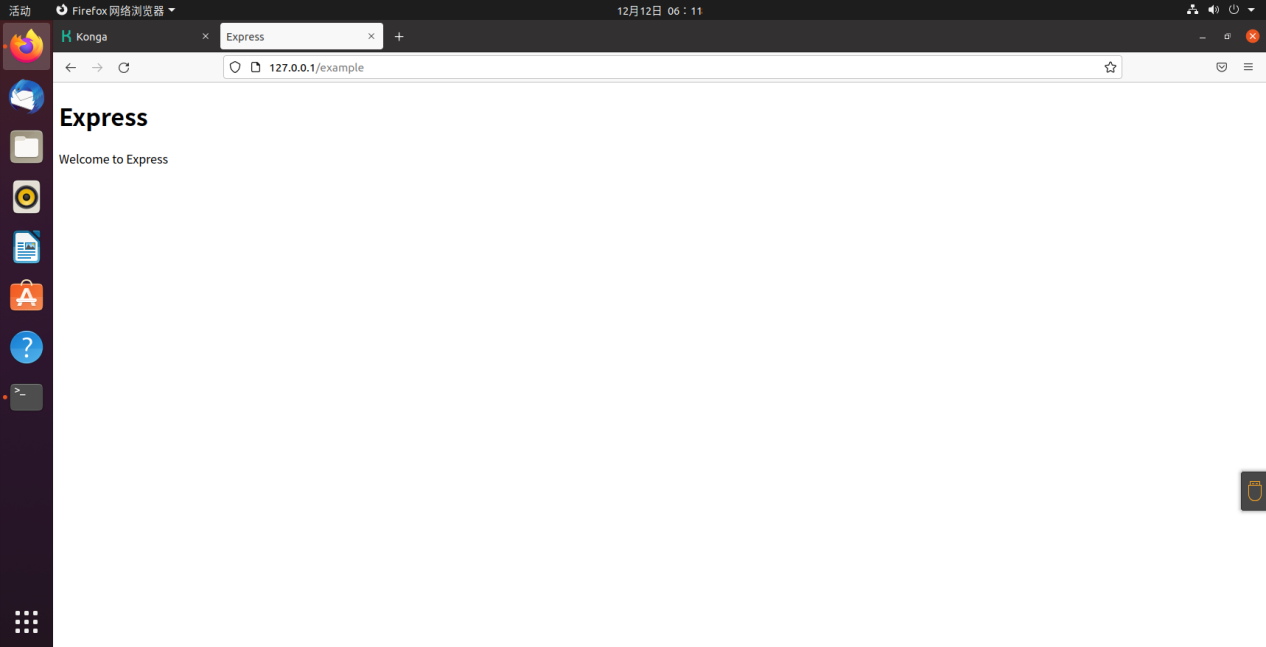
如下我们添加一个路由



该路由的名称叫做example-route，会将/example开头的url匹配的该服务

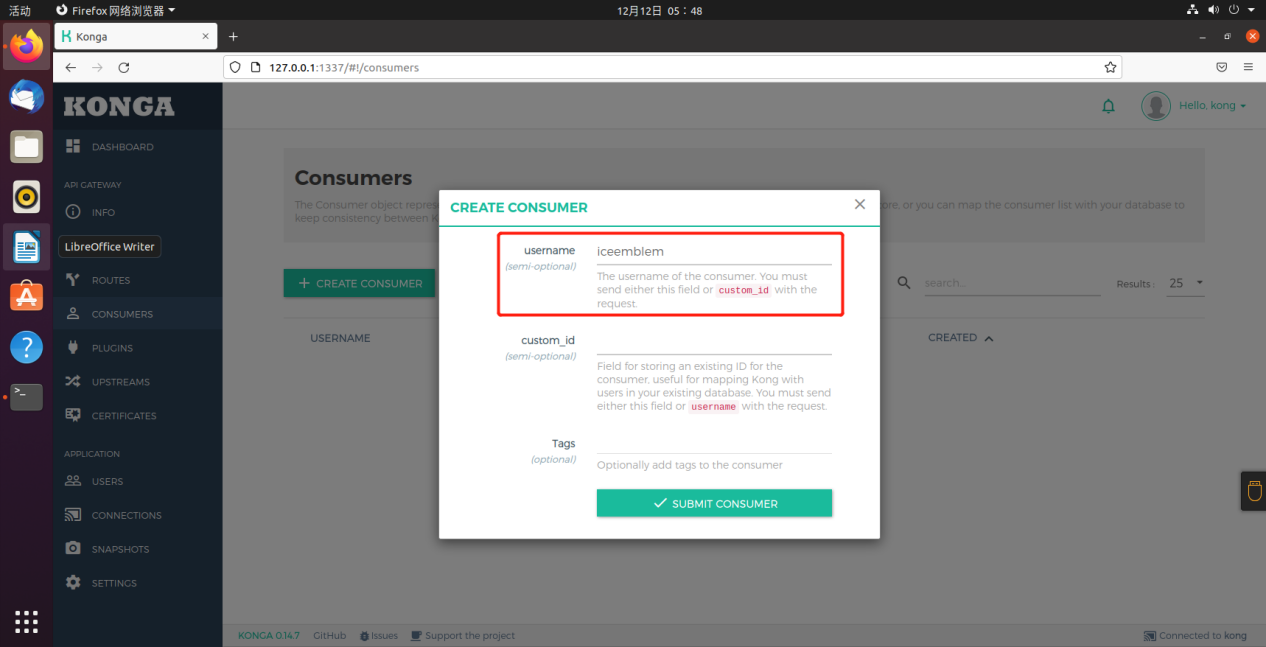
注：Paths要按一下回车键才会应用该值，被坑到了

测试，我们访问http://127.0.0.1/example就可以看到example服务器：



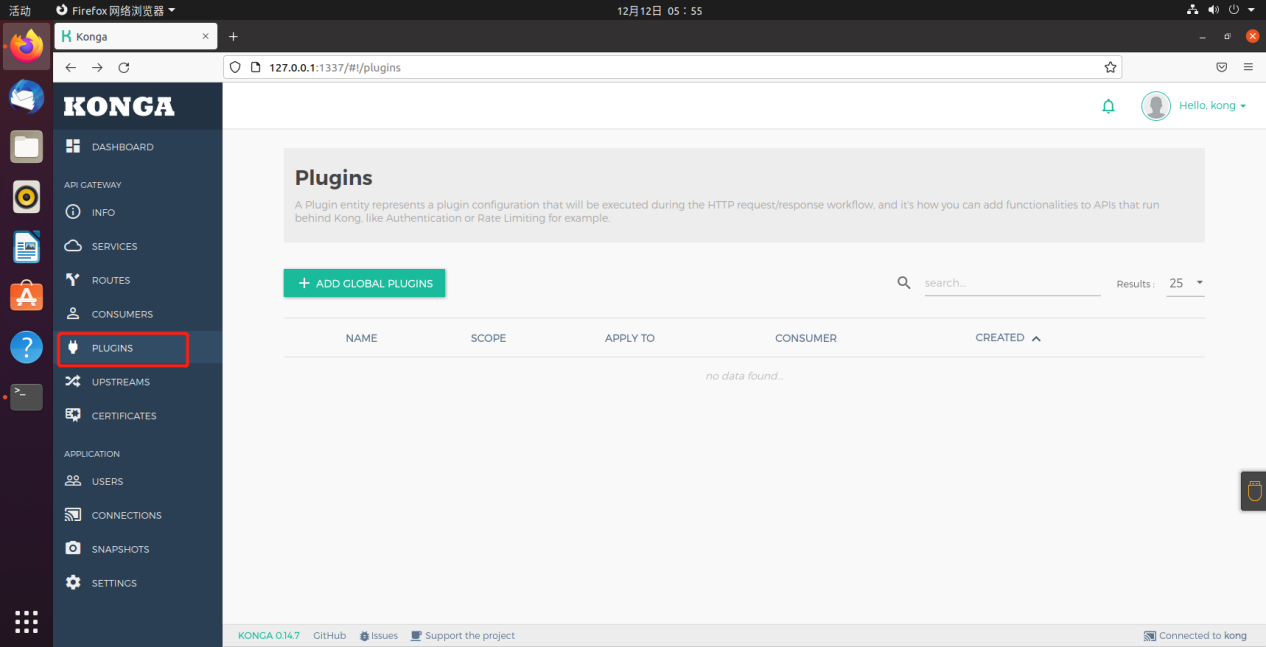
**消费者**

如下我们创建一个消费者



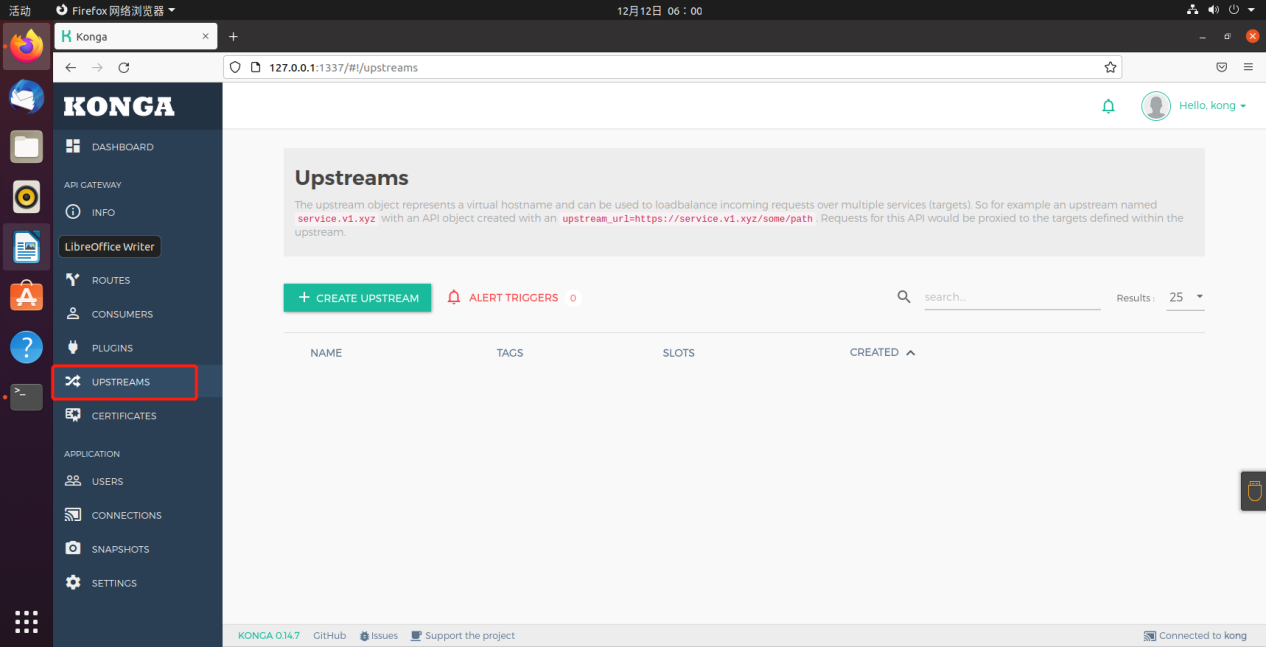
**插件**

PLUGINS显示了当前kong安装的插件

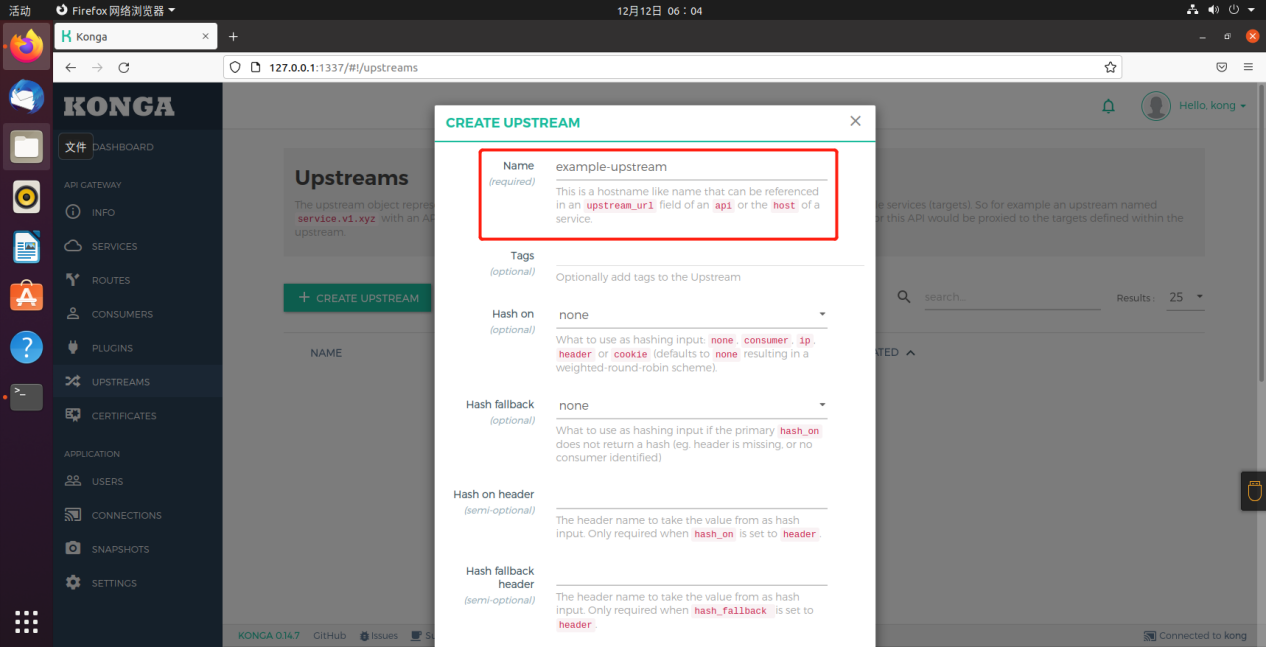


**上游**

UPSTREAMS用于管理当前的上游服务器

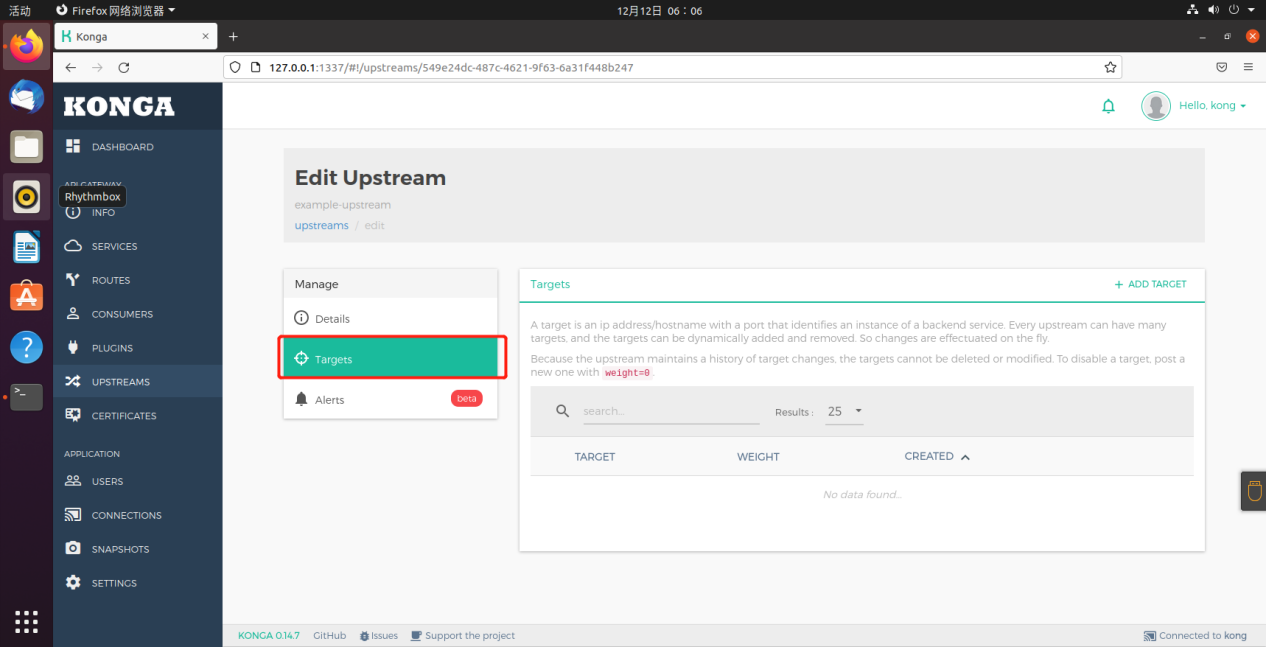


如下我们可以添加一个上游

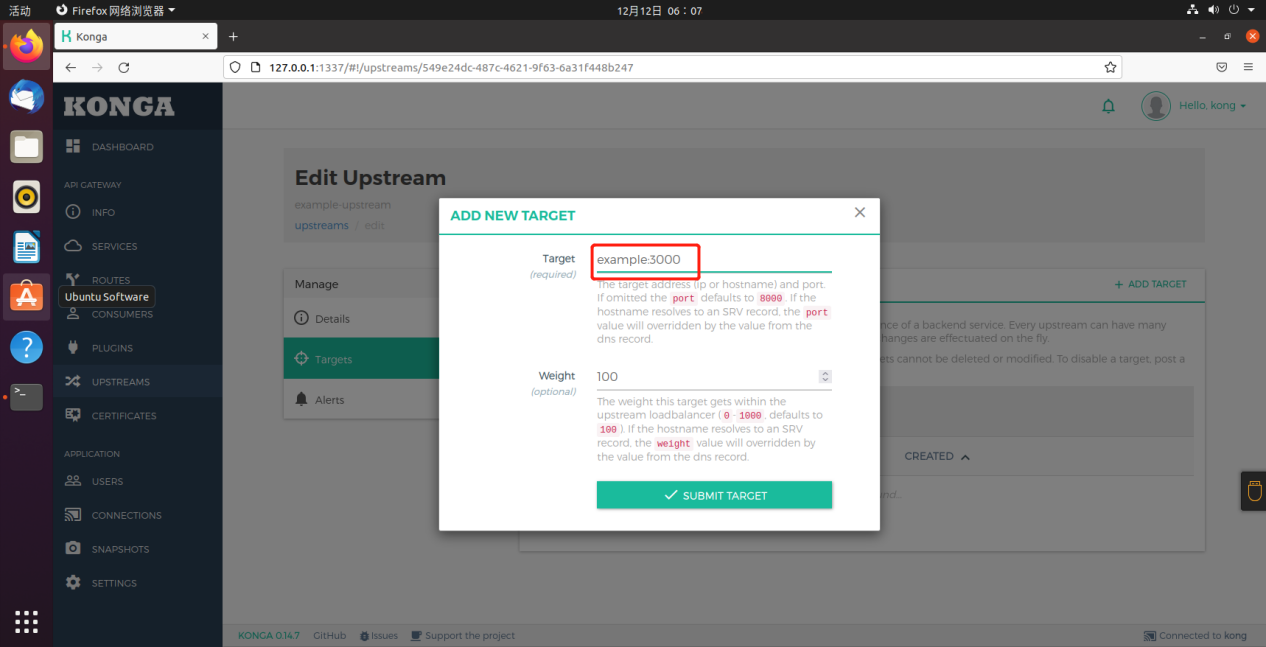


**目标节点**

进入上游详情里面，我们可以管理上游的目标节点

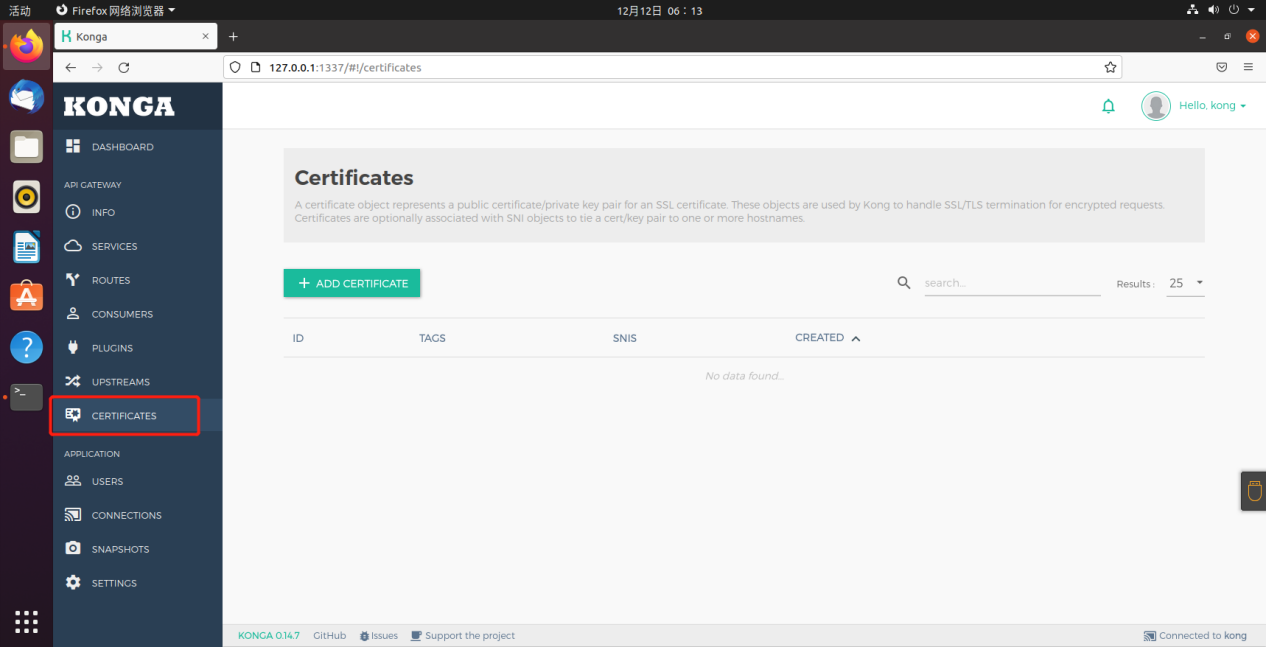


如下我们添加目标节点，其指向的就是我们的测试服务器

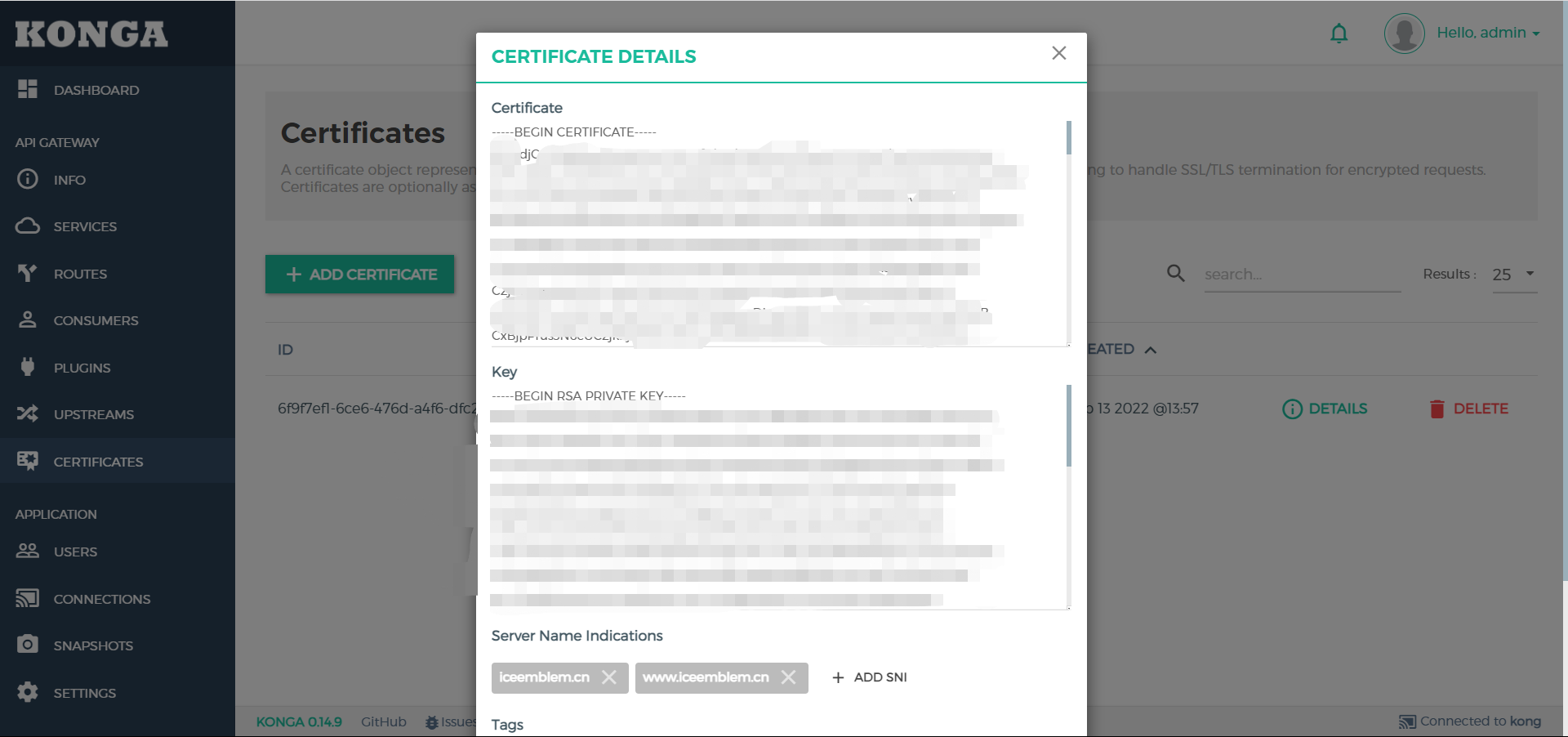


**证书**

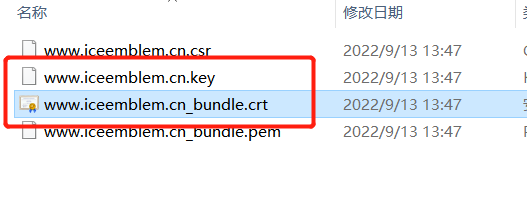
CERTIFICATES页面用于管理我们的SSL/TLS证书



如下添加正式



Certificate和Key对应我们的crt和key这2个文件的内容



Server Name Indications是我们正式绑定的域名

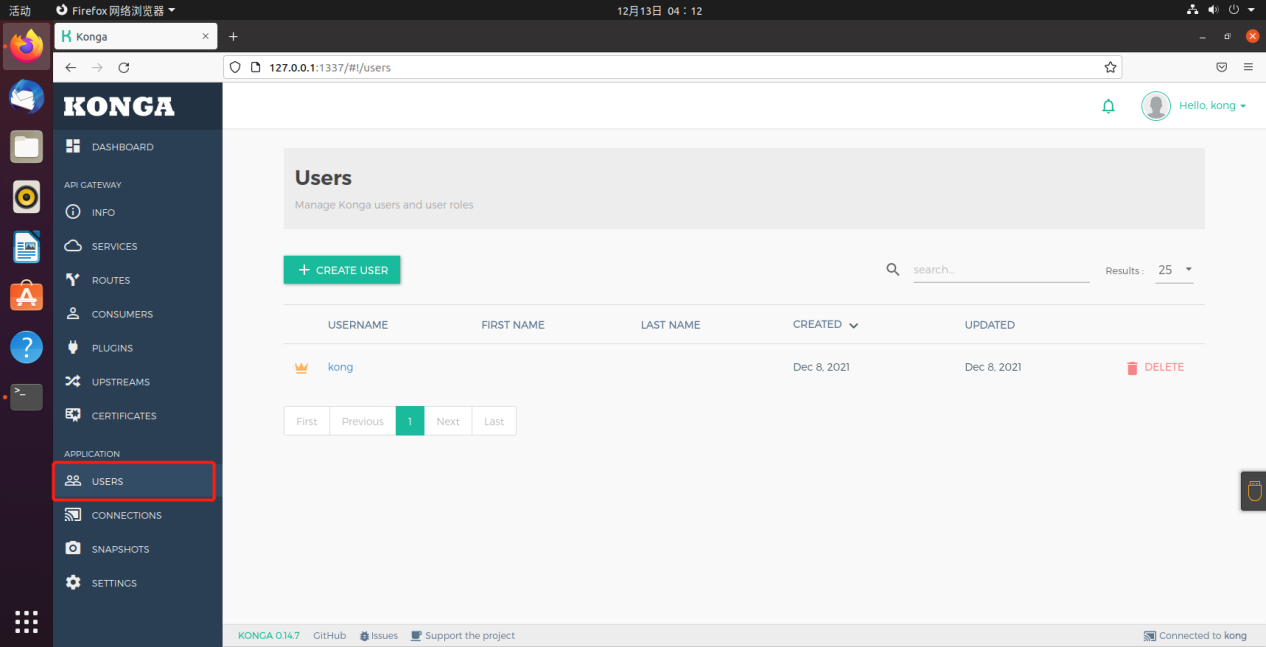
注：这里遇到了一个吭，他丫的从网上复制了一段docker-compose的代码，其环境变量如下，看似没问题实则有天大的坑，kong的http和https默认是8000和8443端口，不用写下面两行则运行正常，加上下面两行后https就失效了，原因应该是KONG\_PROXY\_LISTEN\_SSL不是一个有效的配置名，导致kong只暴露了8000端口

KONG\_PROXY\_LISTEN: 0.0.0.0:8000

KONG\_PROXY\_LISTEN\_SSL: 0.0.0.0:8443

**用户**

USERS页面管理kong的用户，可以添加kong管理员或普通用户等



**快照管理**

SNAPSHOTS页面用于管理kong的快照

如下我们添加一个任务，每周1进行一次快照

