一 kong 部署

1.1 部署 kong 的依赖数据库 postgres

```
docker run -d --name kong-database --net=host -v
/data/kong-database/data:/var/lib/postgresql/data -e
"POSTGRES_USER=kong" -e "POSTGRES_DB=kong" postgres:9.5
```

1.2 初始化 kong 数据库

```
docker run --rm \
    --link kong-database:kong-database \
    -e "KONG_DATABASE=postgres" \
    -e "KONG_PG_HOST=kong-database" \
    -e "KONG_CASSANDRA_CONTACT_POINTS=kong-database" \
    172.16.59.153/aiaas/aikong:1.0.3 kong migrations up
```

1.3 启动 kong

```
docker run -d --name kong  --net=host \
    -e "KONG_PG_HOST=10.1.87.70" \
    -e "KONG_PLUGINS=bundled,xfyun-hmac-ws" \
       -e "KONG_LOG_LEVEL=error" \
    -e "KONG_ADMIN_LISTEN=127.0.0.1:8001" \
    -e "KONG_ADMIN_LISTEN_SSL=127.0.0.1:8444" \
    -e "KONG_PROXY_LISTEN=0.0.0.0:80" \
    -e "KONG_PROXY_LISTEN_SSL=0.0.0.0:443" \
    172.16.59.153/aiaas/aikong:1.0.3
```

1.4 部署 kong 管理控制台 konga

```
docker run -d  --net=host -e "TOKEN_SECRET=123456"  -e
"NODE_ENV=production" --name konga  pantsel/konga:0.13.0
```

二 kong 配置 (使用 konga 控制台)

准备工作：CONNECTIONS->NEW CONNECTION 将 kong 的 admin api 地址添加进来（http:127.0.0.1:8001）

2.1 创建 serveice

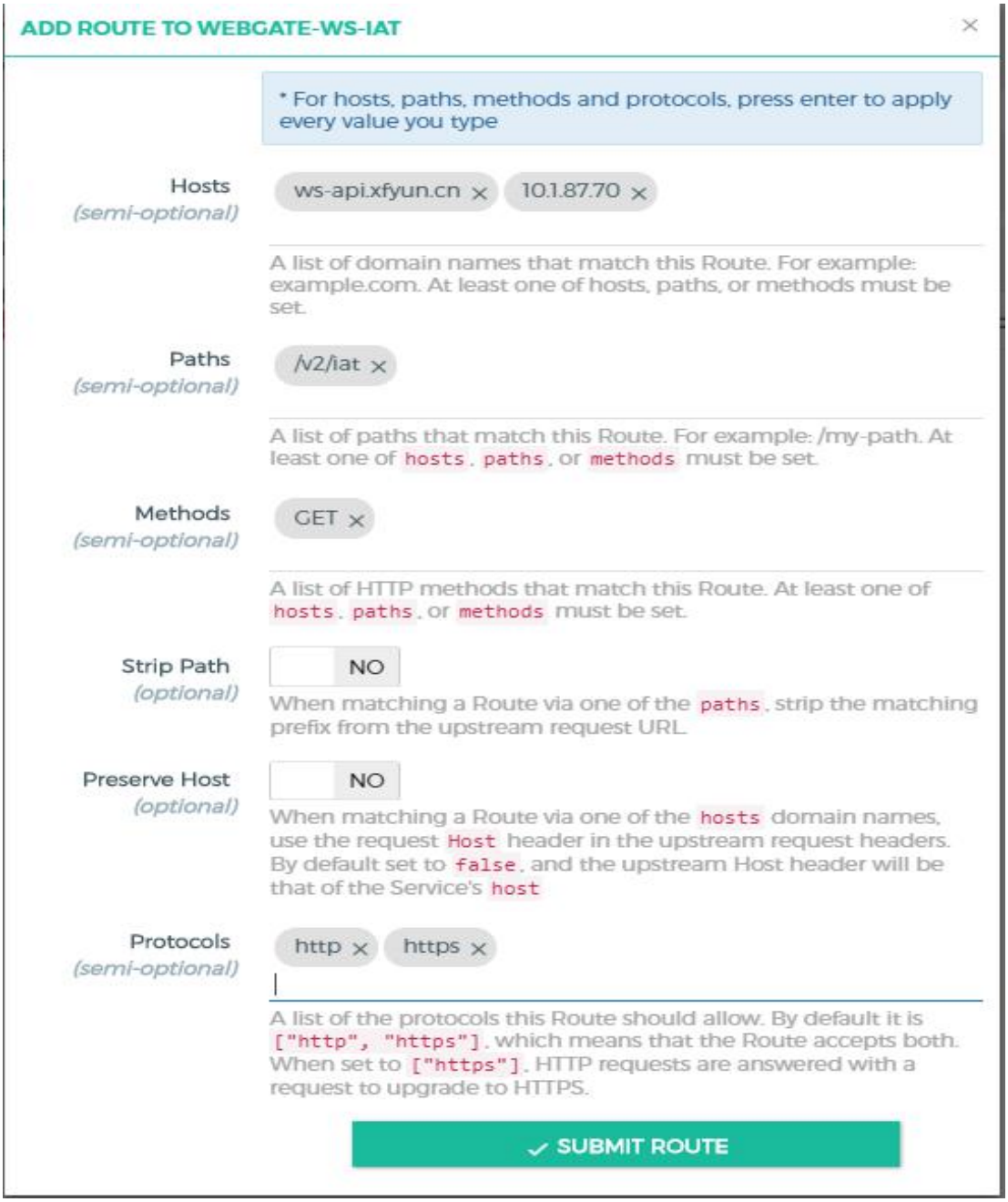
SERVICES->ADD NEW SERVICE

必填配置：

Host：上游服务地址，可以是个具体的 IP 地址，也可以是个 upstream-name
Port：上游服务端口号

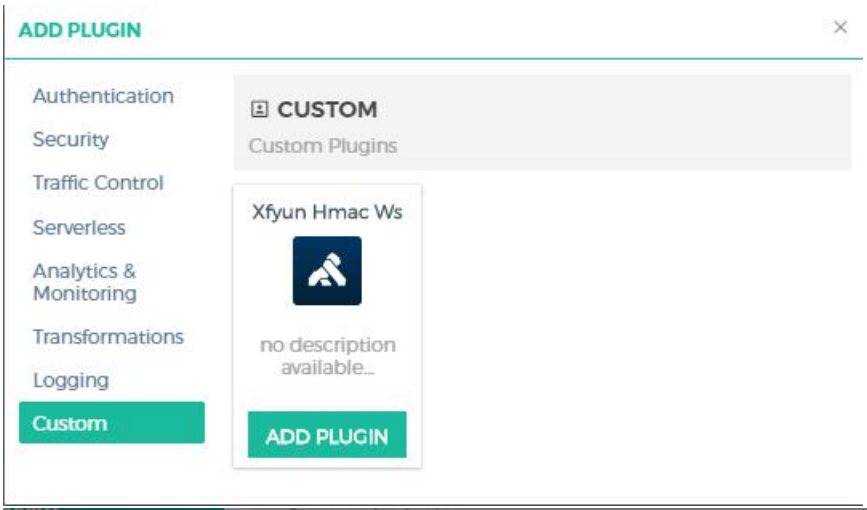
2.2 创建 Routes

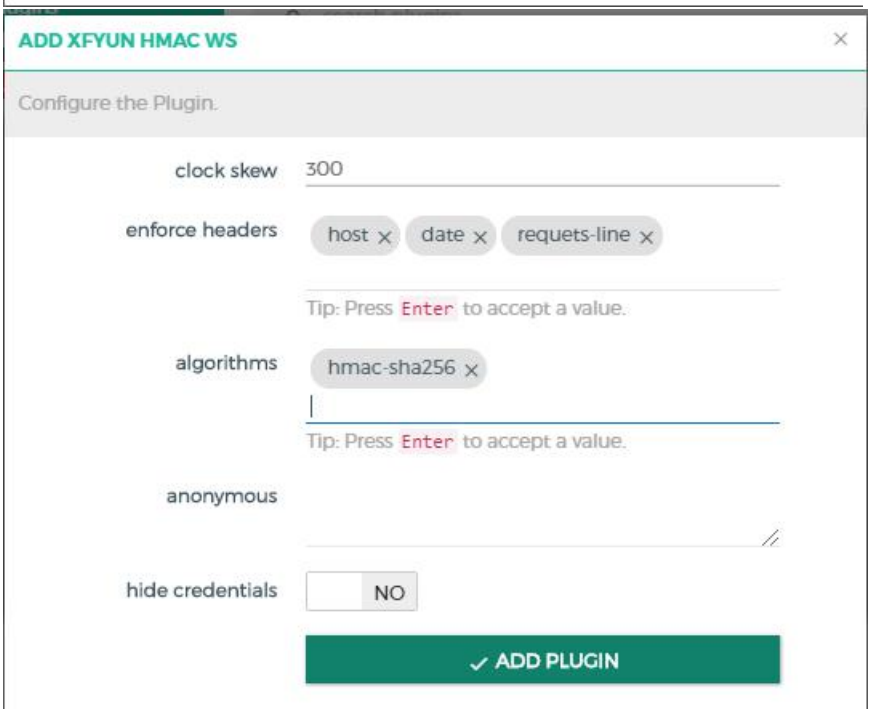

Service webgate-ws-iat
services / show
Service Details
Routes
Plugins
Eligible consumers
beta
Routes
+ ADD ROUTE
search routes...
Hosts
Paths
Protocols
Methods
Regex priority
Created


ADD ROUTE TO WEBGATE-WS-IAT
* For hosts, paths, methods and protocols, press enter to apply every value you type
Hosts
(semi-optional)
ws-api.xfyun.cn
10.1.87.70
A list of domain names that match this Route. For example: example.com. At least one of hosts, paths, or methods must be set.
Paths
(semi-optional)
/v2/iat
A list of paths that match this Route. For example: /my-path. At least one of hosts, paths, or methods must be set.
Methods
(semi-optional)
GET
A list of HTTP methods that match this Route. At least one of hosts, paths, or methods must be set.
Strip Path
(optional)
NO
When matching a Route via one of the paths, strip the matching prefix from the upstream request URL.
Preserve Host
(optional)
NO
When matching a Route via one of the hosts domain names, use the request Host header in the upstream request headers. By default set to false, and the upstream Host header will be that of the Service's host
Protocols
(semi-optional)
http
https
A list of the protocols this Route should allow. By default it is ["http", "https"], which means that the Route accepts both. When set to ["https"], HTTP requests are answered with a request to upgrade to HTTPS.
SUBMIT ROUTE

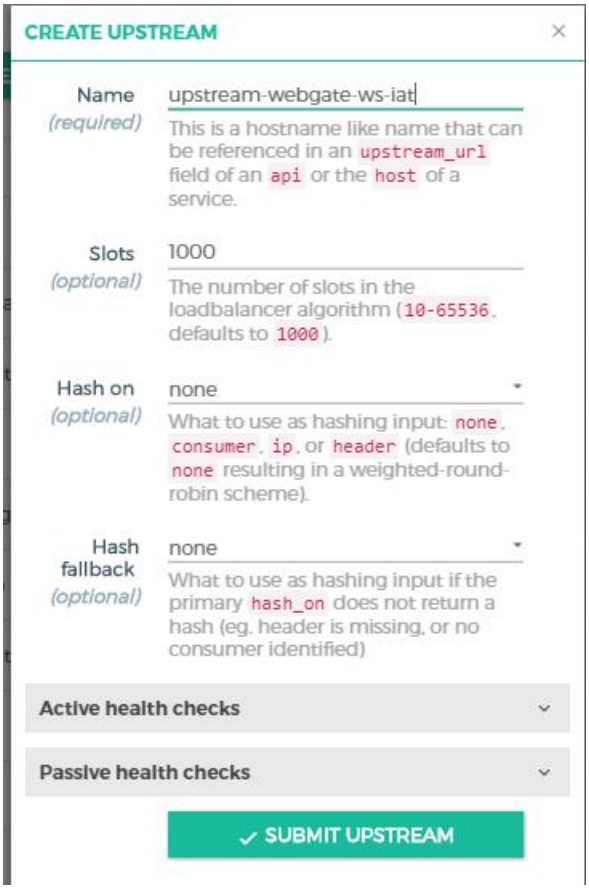
2.3 配置鉴权插件

Websocket 接口使用 xfyun-hmac-ws 插件，http 接口使用 hmac-auth 插件





2.4 创建 upstream ，upstream name 和 第一步创建 service 中填的 Host 中的值保持一致，



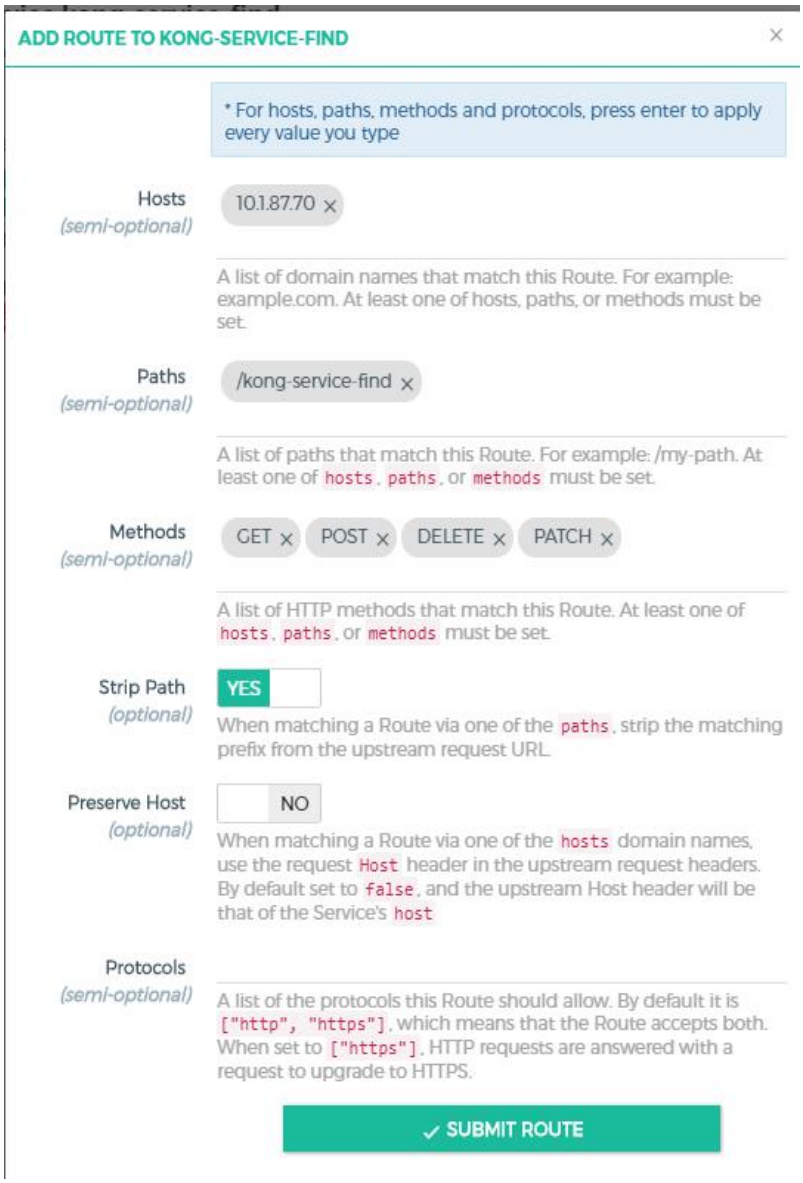
Name:upstream name
Slots：哈希算法中哈希槽的数量
Hash on:用于计算 hash 的字段
Hash fallback：当没有取到 hash 字段时用于计算 hash 的字段

经过以上四步，一个基本可用的服务即可配置出来

三 kong 服务发现配置，使上游服务启动时能够自动注册 upstream。
3.1 创建 service kong-service-find
Name： kong-service-find
Host：127.0.0.1 # kong admin api 监听地址
Port: 8001 #kong admin api 监听端口

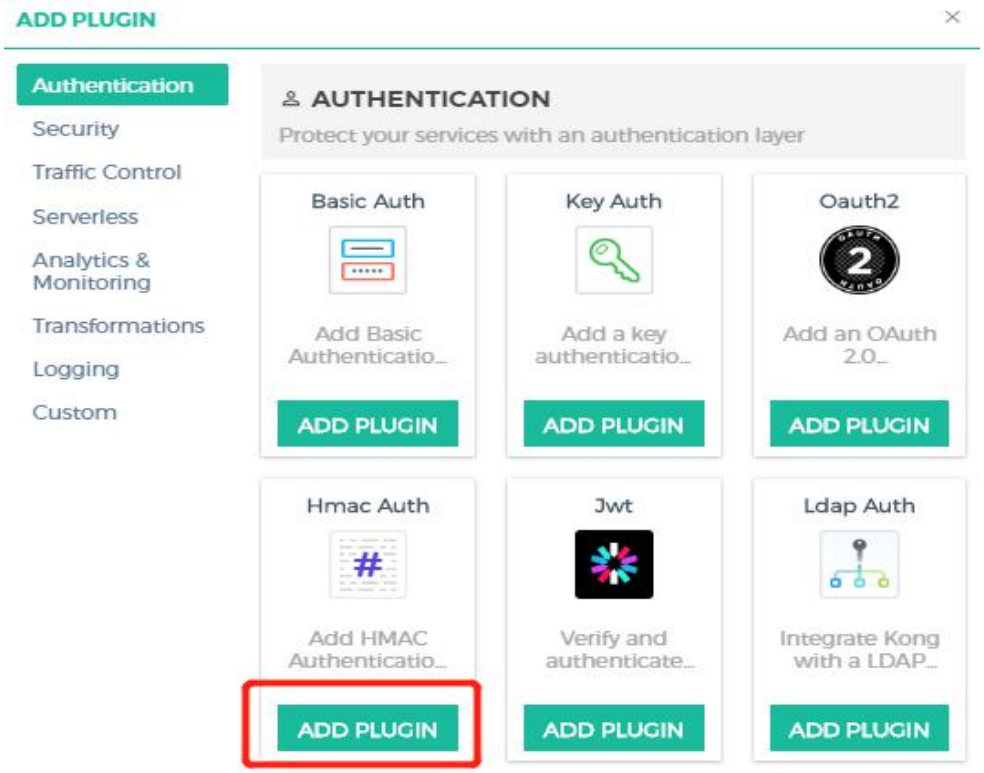
Service 的创建方法参考 2.1

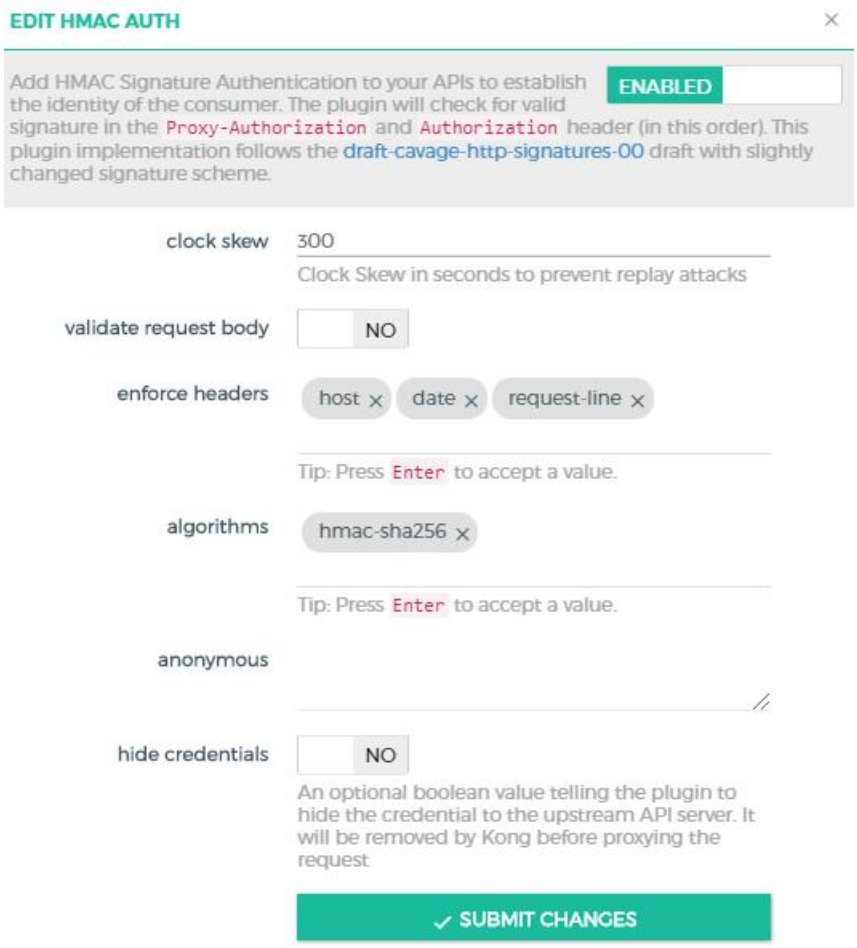
3.2 为 kong-service-find 创建路由

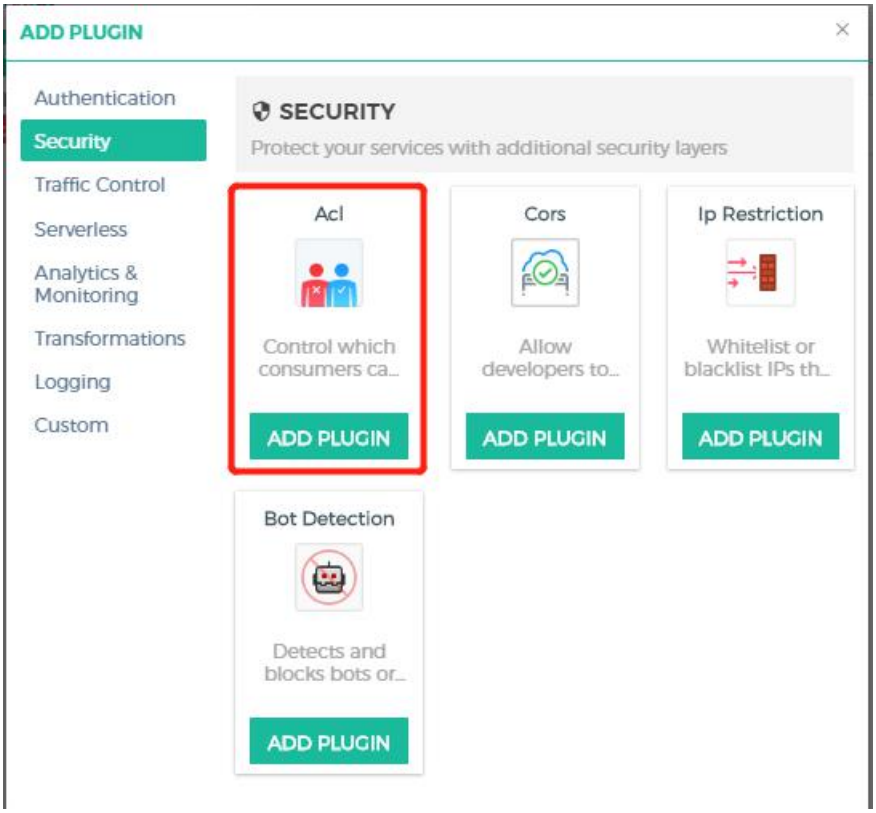


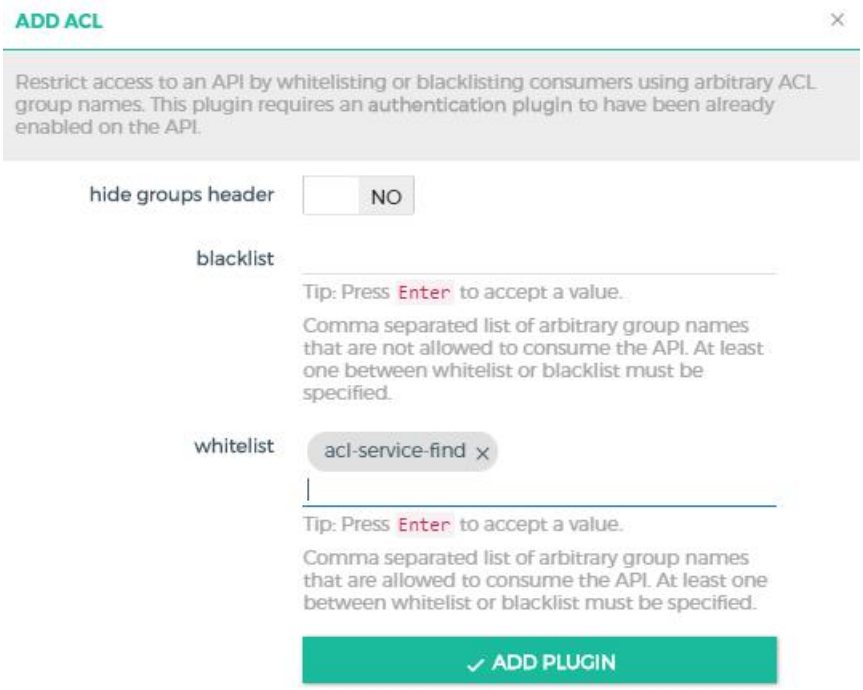
需要开启 strip path，如上，开启后上游服务可用通过 curl -X POST http://10.1.87.70:8000/kong-service-find/upstreams/upstream-webgate-ws-iat/targets -d “target=10.1.87.70:8082”
向 kong 注册服务。

3.3 服务发现路由放在 kong 上相当于暴露在公网上，需要添加 hmac-auth 插件并且添加 acl 控制。让只有指定的用户可以访问该接口


ADD PLUGIN
Authentication
Security
Traffic Control
Serverless
Analytics & Monitoring
Transformations
Logging
Custom
AUTHENTICATION
Protect your services with an authentication layer
Basic Auth
Add Basic Authenticatio...
ADD PLUGIN
Key Auth
Add a key authenticatio...
ADD PLUGIN
Oauth2
Add an OAuth 2.0...
ADD PLUGIN
Hmac Auth
Add HMAC Authenticatio...
ADD PLUGIN
Jwt
Verify and authenticate...
ADD PLUGIN
Ldap Auth
Integrate Kong with a LDAP...
ADD PLUGIN

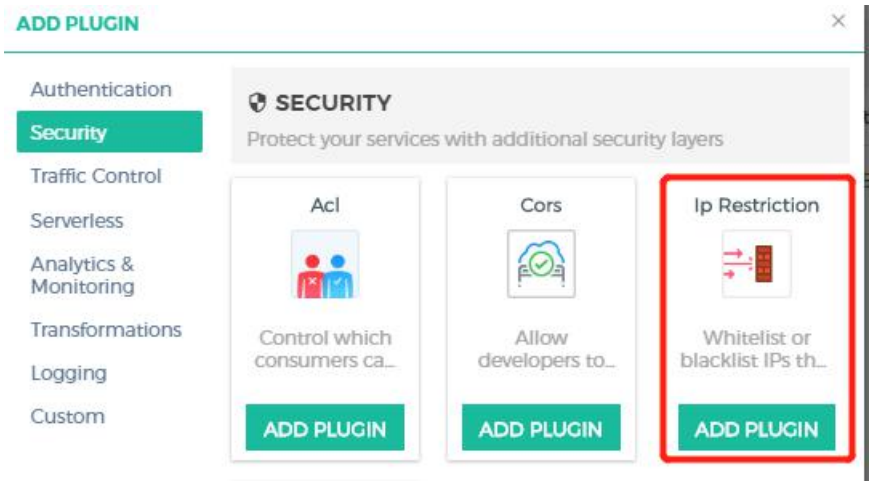

EDIT HMAC AUTH
Add HMAC Signature Authentication to your APIs to establish the identity of the consumer. The plugin will check for valid signature in the Proxy-Authorization and Authorization header (in this order). This plugin implementation follows the draft-cavage-http-signatures-00 draft with slightly changed signature scheme.
ENABLED
clock skew
300
Clock Skew in seconds to prevent replay attacks
validate request body
NO
enforce headers
host
date
request-line
Tip: Press Enter to accept a value.
algorithms
hmac-sha256
Tip: Press Enter to accept a value.
anonymous
hide credentials
NO
An optional boolean value telling the plugin to hide the credential to the upstream API server. It will be removed by Kong before proxying the request
SUBMIT CHANGES

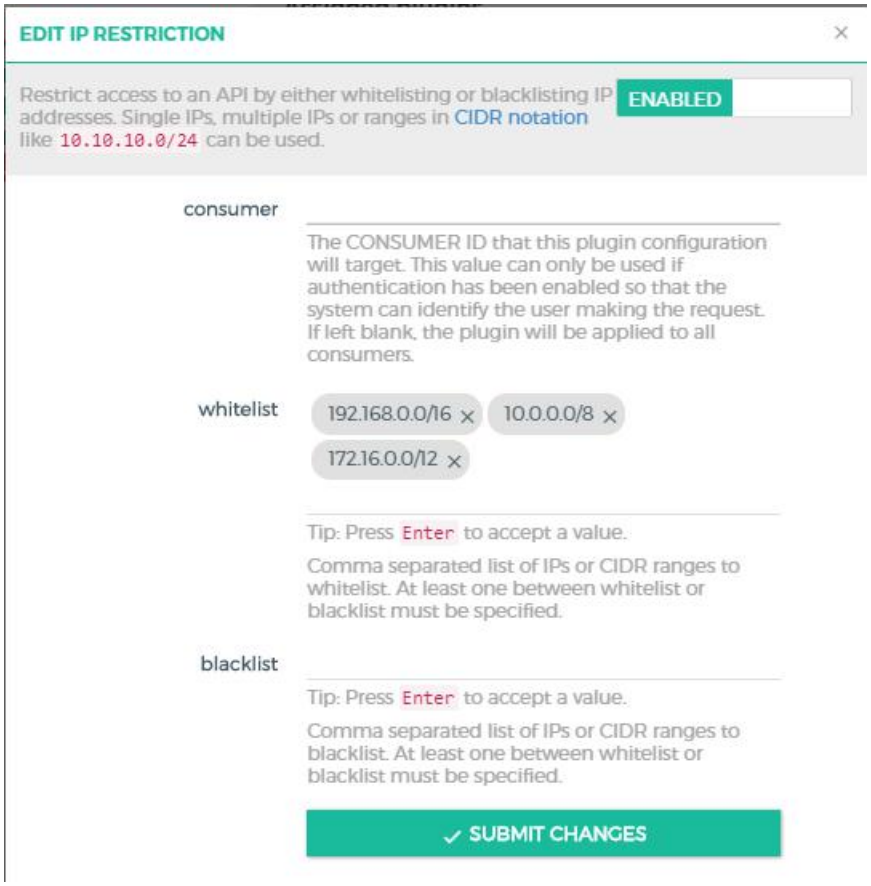




添加了 hmac-auth 后上游服务需要认证登录才能访问该服务，（后面涉及到用户的插件都需要开启这个插件或者 xfyun-hamc-ws 插件才能生效）
添加了 acl whitelist 后只有隶属于改 whitelist 组的用户才有权限访问该服务。用户和用户组的添加参考下一节

为了更加安全起见，还可以为服务配置 ip 白名单，使该接口只能通过内网访问

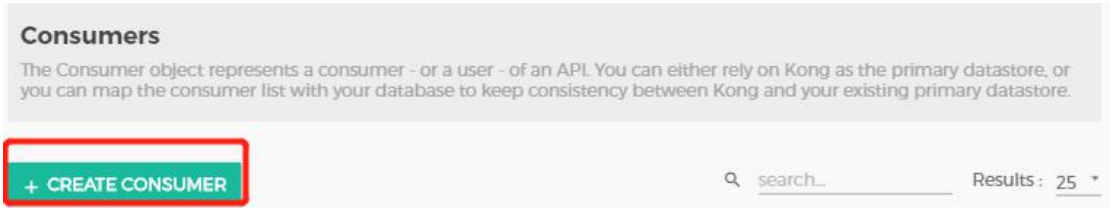


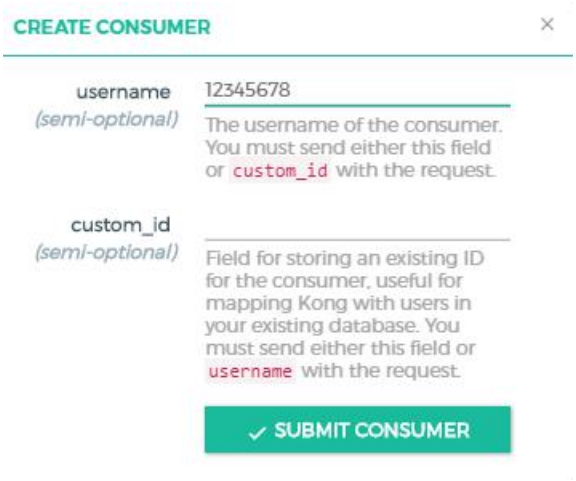


四 用户的创建

4.1 创建用户(App_id)

CONSUMERS-> CREATE CONSUMER

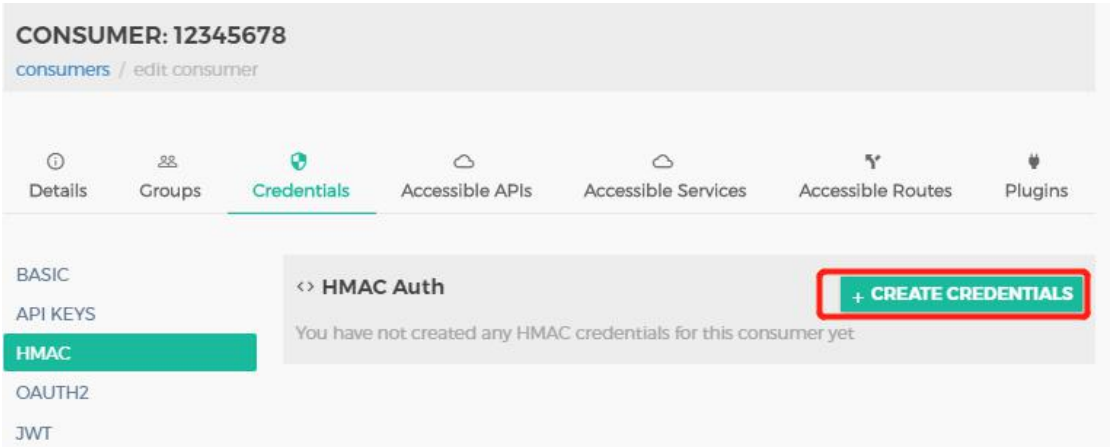


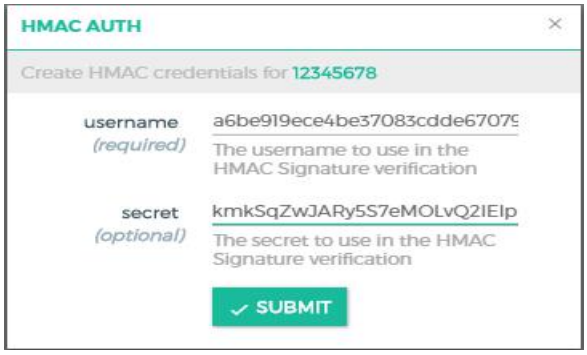


Username: 用户名，即 appid
Custom_id: 用来关联外部的用户 id，可以为空

4.2 创建 api_key 和 secret

api_key 和 apisecret 用于鉴权，配置了鉴权插件（hmac-auth，xfyun-hmac-auth）的服务需要带上 api_key 和 apisecret 构建鉴权的参数才能访问服务。

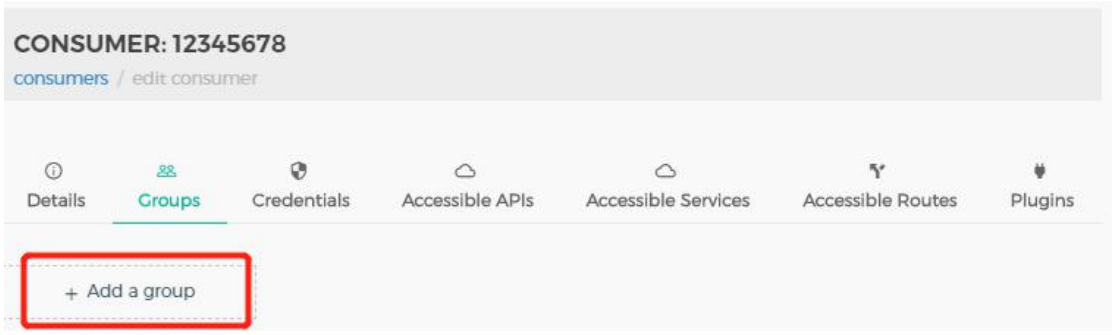


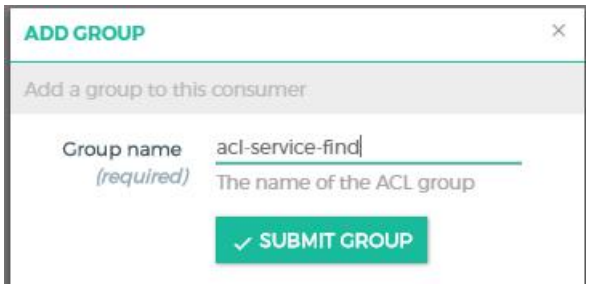


Username: 即 api_key
Secret: 即 api_sercret

4.3 添加用户到 acl 组

对于添加了 acl 插件的服务，只有在 acl 组中的用户才能访问到改服务，如第三章的服务发现服务，为用户添加到 acl 组后，该用户就有了访问对应的 acl 插件控制的服务的权限。





如上图所示，用户 12345678 添加进了服务发现 acl 组，使用 12345678 这个用户的 apikey 和 apisercret 就可以访问到服务发现服务了。