

# 1 使用说明

---

能开平台接入只需要修改 原先的 请求url,并在消息头中加入 服务相关的信息即可,不影响原先的请求消息体。  
假设 注册的服务信息如下

```
appCode testApp  
apiCode testApi  
version 1.0.0  
能开的地址为 192.168.0.1
```

## 2 接入服务

---

网管支持的上游接入服务为 Restful风格的 或者是 http/https + json /text/xml/text

### 2.1 restful 风格接入

[关于rest风格的api介绍](#)

简单就是用 get post put delete 来表示资源的增删改查, uri 中使用占位符来表示具体的资源

有如下的rest 服务需要接入

```
http://api.example.com/user-management/users/{id}/accounts
```

#### 2.1.1 新建服务

创建服务

\* 服务名称

\* 服务编码

\* 协议  RESTFUL  HTTP  HTTPS

超时时间(毫秒)

资源路径  ⓘ

\* 生效时间:  ×

\* 失效时间:  ×

描述

## 2.1.2 版本配置

\* 版本号

\* 版本名称

\* 服务类型  单服务  编排服务

\* 访问路径

\* 访问方式  GET  POST  PUT  DELETE

\* 路由策略  随机  轮询  最小时延

\* 涉敏类型  不涉敏  涉敏  脱敏  高风险(用作展示与涉敏级别一样)

操作名称  ⓘ

超时时间(毫秒)

重试次数

请求报文大小

版本描述

### 2.1.3 节点配置

### 创建节点分组 ✕

节点分组名称

\*节点分组编码

\*协议  RESTFUL  HTTP  CSF  Eureka  SOCKET

\*节点获取方式  静态  动态

\*节点集合  ⓘ

\*客户端路径   
ⓘ 客户端路径不能为空

- 接入协议是 http

```
# 配置客户端路径  
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSrvAccessHandler
```

- 接入协议是 https

```
# 配置客户端路径  
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSslSrvAccessHandler
```

## 2.1.4 请求网关的路径

待确认

```
http://192.168.0.1:21100/rest
```

## 2.2 http/https+json 接入

有如下服务接入

```
POST /dev/oh/qryApp HTTP/1.1
Host: localhost:8080
Content-Type: application/json
Content-Length: 46

{
  "appCode": "app_16291992646709486308"
}
```

## 2.2.1 新建服务

新建服务

\* 服务名称

\* 服务编码

\* 协议

报文格式

超时时间(毫秒)

资源路径  ⓘ

\* 生效时间:  ×

\* 失效时间:  ×

描述

## 2.2.2 版本配置

* 版本号	<input type="text"/>
* 版本名称	<input type="text"/>
* 服务类型	<input checked="" type="radio"/> 单服务 <input type="radio"/> 编排服务
* 访问路径	<input type="text" value="/dev/oh/qryApp"/>
* 访问方式	<input type="radio"/> GET <input checked="" type="radio"/> POST <input type="radio"/> PUT <input type="radio"/> DELETE
* 路由策略	<input type="radio"/> 随机 <input checked="" type="radio"/> 轮询 <input type="radio"/> 最小时延
* 涉敏类型	<input checked="" type="radio"/> 不涉敏 <input type="radio"/> 涉敏 <input type="radio"/> 脱敏 <input type="radio"/> 高风险(用作展示与涉敏级别一样)
操作名称	<input type="text"/>
超时时间(毫秒)	<input type="text" value="60000"/>
重试次数	<input type="text" value="0"/>
请求报文大小	<input type="text"/>
版本描述	<input type="text"/>

### 2.2.3 节点配置

节点分组名称

\*节点分组编码

\*协议 RESTFUL HTTP CSF Eureka SOCKET

报文格式 JSON XML TEXT FORM

\*节点获取方式 静态 动态

\*节点集合  ⓘ

\*客户端路径

客户端路径不能为空

取消 提交

- 接入协议是 http

# 配置客户端路径

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSrvAccessHandler
```

- 接入协议是 https

# 配置客户端路径

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSslSrvAccessHandler
```

## 2.2.4 请求网关的路径

- http调用

```
http://192.168.0.1:21100/http
```

- https 调用

```
https://192.168.0.1:8443/https
```

## 2.3 http/https+formdata 接入

假设有如下 formdata 服务接入

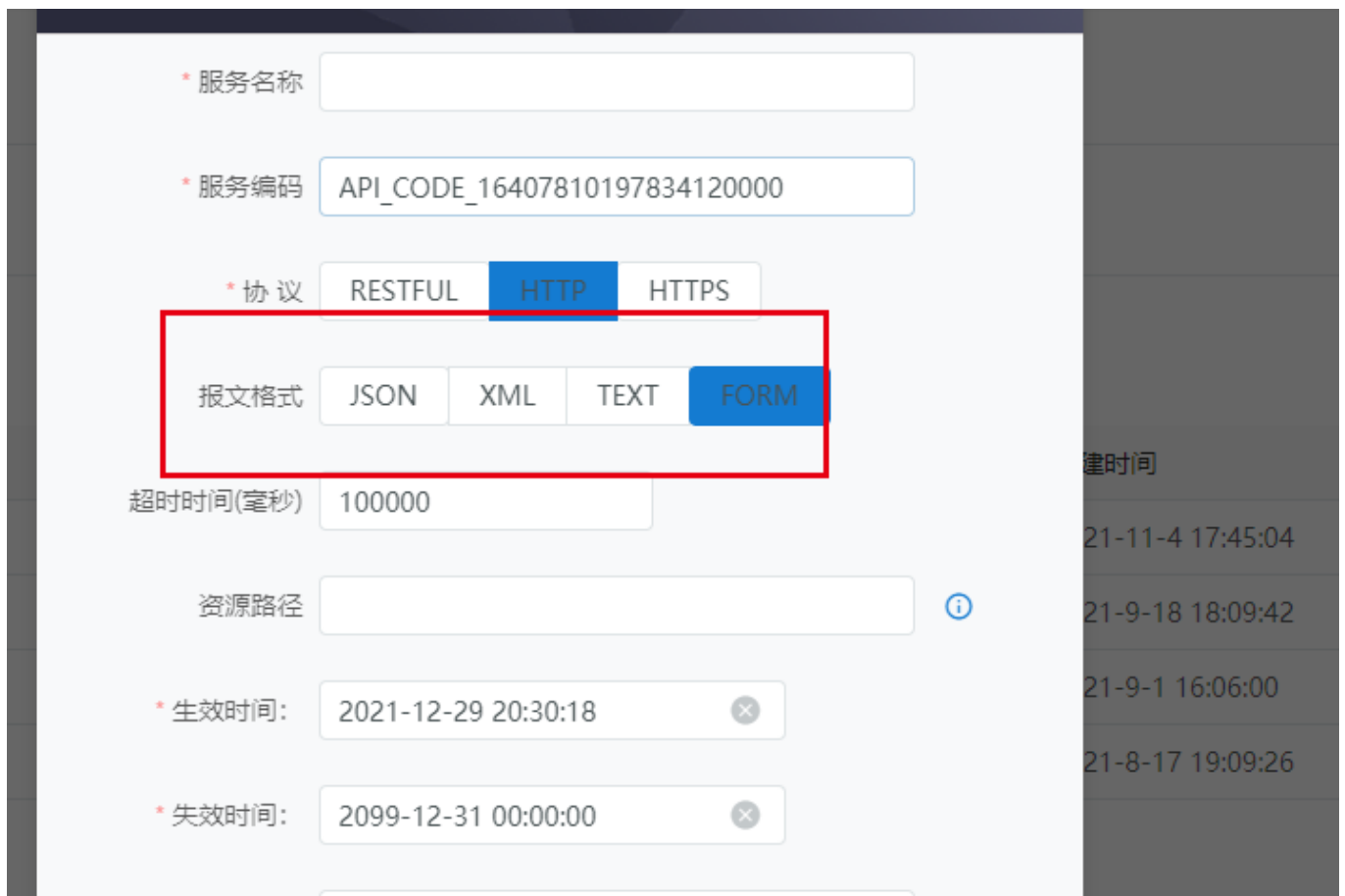
```
POST /test2 HTTP/1.1
Host: localhost:8083
Content-Length: 208
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW

----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="name"

aaa
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="id"

2
----WebKitFormBoundary7MA4YWxkTrZu0gW
```

### 2.3.1 新建服务



The screenshot shows a configuration form for a new service. The '协议' (Protocol) is set to 'HTTP' and '报文格式' (Message Format) is set to 'FORM'. A red box highlights these two settings. Other fields include '服务名称' (Service Name), '服务编码' (Service Code: API\_CODE\_16407810197834120000), '超时时间(毫秒)' (Timeout in milliseconds: 100000), '资源路径' (Resource Path), '生效时间' (Valid Time: 2021-12-29 20:30:18), and '失效时间' (Expired Time: 2099-12-31 00:00:00). On the right, a table shows a list of services with their creation times.

创建时间
21-11-4 17:45:04
21-9-18 18:09:42
21-9-1 16:06:00
21-8-17 19:09:26

### 2.3.2 版本配置

\* 版本号

\* 版本名称

\* 服务类型  单服务  编排服务

\* 访问路径

\* 访问方式  GET  POST  PUT  DELETE

\* 路由策略  随机  轮询  最小时延

\* 涉敏类型  不涉敏  涉敏  脱敏  高风险(用作展示与涉敏级别一样)

操作名称

超时时间(毫秒)

重试次数

### 2.3.3 节点配置

节点分组名称

\*节点分组编码

\*协议

报文格式

\*节点获取方式

\*节点集合  ⓘ

\*客户端路径

- 接入协议是 http

```
# 配置客户端路径
```

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSrvAccessHandler
```

- 接入协议是 https

```
# 配置客户端路径
```

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.HttpSslSrvAccessHandler
```

## 2.3.4 请求网关的路径

- http调用

```
http://192.168.0.1:21100/http
```

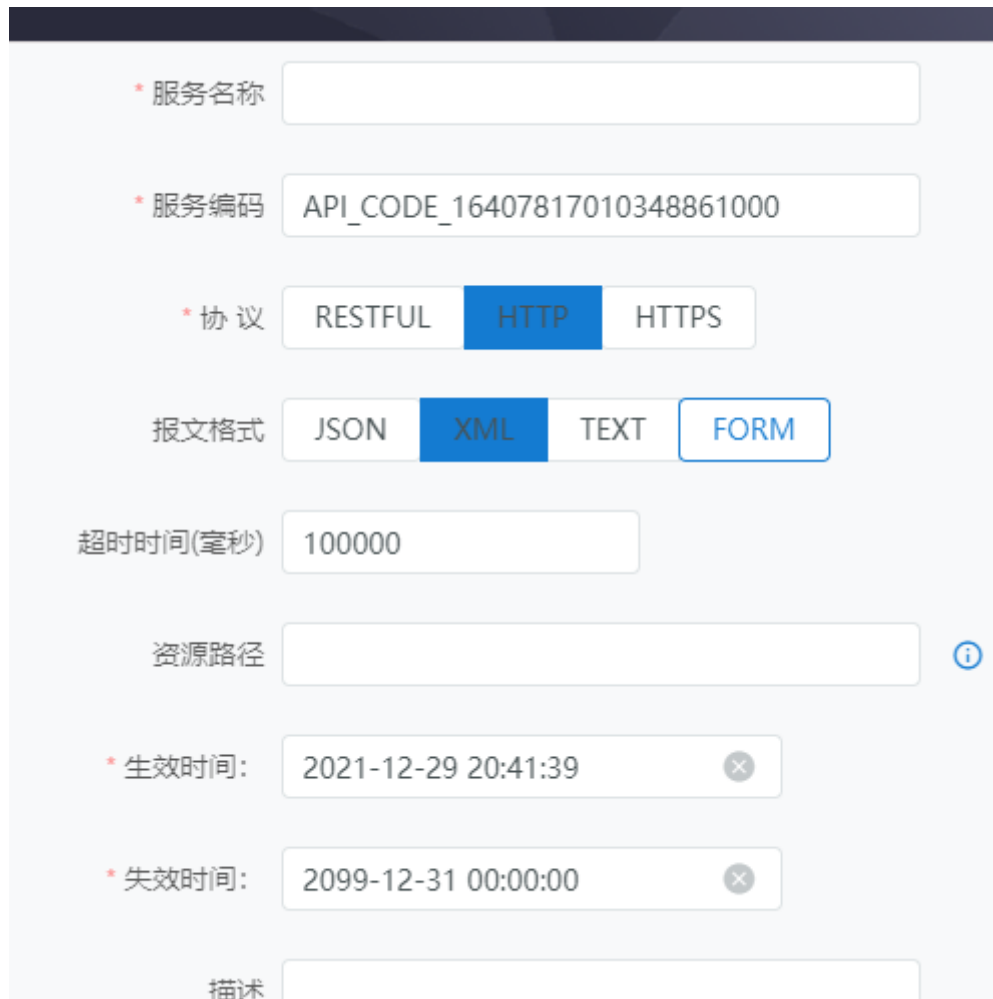
- https 调用

```
https://192.168.0.1:8443/https
```

## 2.4 http/https+xml 接入

soap 服务接入，参考其他已经实现的接入方式

### 2.4.1 新建服务



The screenshot shows a configuration form for a new service. The form includes the following fields and options:

- \* 服务名称**: A text input field.
- \* 服务编码**: A text input field containing the value "API\_CODE\_16407817010348861000".
- \* 协议**: A group of three buttons: "RESTFUL", "HTTP" (selected), and "HTTPS".
- 报文格式**: A group of four buttons: "JSON", "XML" (selected), "TEXT", and "FORM".
- 超时时间(毫秒)**: A text input field containing the value "100000".
- 资源路径**: A text input field with an information icon (i) to its right.
- \* 生效时间:**: A date and time picker field showing "2021-12-29 20:41:39" with a clear (x) button.
- \* 失效时间:**: A date and time picker field showing "2099-12-31 00:00:00" with a clear (x) button.
- 描述**: A text input field at the bottom.

### 2.4.2 版本配置

### 2.4.3 节点配置

节点分组名称

\*节点分组编码

\*协议

报文格式

\*节点获取方式

\*节点集合  ⓘ  
ⓘ 节点集合不能为空!

\*客户端路径   
ⓘ 客户端路径不能为空

# 配置客户端路径

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.WebServiceAccessHandler
```

## 2.4.4 请求网关的路径

待确认

```
http://192.168.0.1:21100/soap
```

## 2.5 http/https+socket 接入(比如短信接口)

```
POST /http HTTP/1.1
Host: 192.168.1.10:21100
appCode: app_16291992646709486308
apiCode: API_CODE_16291985439783020000
apiVersion: 2.0.0
format: text
Content-Type: text/plain
Content-Length: 9

{"1":"2"}
```

## 2.5.1 新建服务

\* 服务名称

\* 服务编码

\* 协议  RESTFUL  HTTP  HTTPS

报文格式  JSON  XML  TEXT  FORM

超时时间(毫秒)

资源路径  ⓘ

\* 生效时间:  ×

\* 失效时间:  ×

## 2.5.2 版本配置

略过

## 2.5.3 节点配置

创建节点分组 ×

节点分组名称

\* 节点分组编码

\* 协议  RESTFUL  HTTP  CSF  Eureka  SOCKET

\* 节点获取方式  静态  动态

\* 节点集合  ⓘ

\* 客户端路径

```
# 配置客户端路径
```

```
com.asiainfo.aifgw.srvorigin.process.serviceorigin.impl.SocketSendAccessHandler
```

## 2.5.4 请求网关的路径

- http调用

```
http://192.168.0.1:21100/http
```

- https 调用

```
https://192.168.0.1:8443/https
```

## 3 请求头描述

请求头	是否必填	类型	备注
<u>appCode</u>	是	String	应用编码
<u>apiCode</u>	是	String	服务编码
<u>apiVersion</u>	是	String	服务版本号
<u>format</u>	是	String	请求格式, 如果是 json 请求则为 <u>http+json</u>
<u>sign</u>	否	String	消息签名, 在开启签名校验时必须填
<u>accessToken</u>	否	String	接入认证信息, 在开启接入认证开关时必须填

## 4 场景举例

- 4.1 下游接入服务是 http/https + json , 通过 http 的方式请求能开平台

```
# 请求报文
```

```
curl --location --request POST 'http://192.168.0.1:21100/http' \  
--header 'appCode: testApp' \  
--header 'apiCode: testApi' \  
--header 'apiVersion: 1.0.0' \  
--header 'format: json' \  
--data-raw '{
```

```
"startTime":"2021-09-01 12:21:22",  
"endTime":"2021-09-30 12:21:22"  
}'
```

- 4.2 下游接入服务是 http/https + form , 通过 https 的方式请求能开平台

此处的form 指的是 multipart/form-data;

```
# 请求报文
```

```
curl --location --request POST 'http://192.168.0.1:8443/https' \  
--header 'appCode: testApp' \  
--header 'apiCode: testApi' \  
--header 'apiVersion: 1.0.0' \  
--header 'format: form' \  
--form 'userId="1"' \  
--form 'name="2"'
```