SOFAStack

API 网关

操作指南

产品版本：AntStack Plus 1.11.0

文档版本：20221009

蚂蚁集团
ANT GROUP

法律声明

蚂蚁集团版权所有©2022，并保留一切权利。

未经蚂蚁集团事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

商标声明

蚂蚁集团 ANT GROUP及其他蚂蚁集团相关的商标均为蚂蚁集团所有。本文档涉及的第三方的注册商标，依法由权利人所有。

免责声明

由于产品版本升级、调整或其他原因，本文档内容有可能变更。蚂蚁集团保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在蚂蚁集团授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过蚂蚁集团授权渠道下载、获取最新版的用户文档。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.发布 API 07
1.1. API 管理 07
1.1.1. API 属性说明 07
1.1.2. 创建 API 12
1.1.3. API 版本管理 23
1.1.4. API 流量治理 25
1.1.5. 绑定/解绑授权对象 26
1.1.6. API 调试 28
1.1.7. 导入导出 API 29
1.1.8. 查看/生成 API 文档 30
1.2. API 分组 31
1.2.1. 创建 API 分组 31
1.2.2. 管理 API 分组 33
1.3. 系统集群 35
1.3.1. 创建系统集群 35
1.3.2. 管理系统集群 42
1.4. 路由规则 44
1.4.1. 创建路由规则 44
1.4.2. 管理路由规则 46
1.5. 数据模型 49
1.5.1. 创建数据模型 49
1.5.2. 导入数据模型 50
1.5.3. 管理数据模型 54
1.5.4. SDK 生成 56
1.6. 外部授权 API 59
1.6.1. 使用外部授权 API 59

1.6.2. 管理外部授权 API 64
1.7. 授权管理 66
1.7.1. 创建授权对象 66
1.7.2. 管理授权对象 67
1.7.3. 绑定/解绑 API 71
1.8. 参数映射 71
1.8.1. 创建参数映射 71
1.8.2. 配置参数映射模板 72
1.8.3. 管理参数映射 78
1.9. IP 策略管理 80
1.9.1. 创建 IP 策略 80
1.9.2. 管理 IP 策略 82
1.9.3. 绑定/解绑 API 84
1.10. CORS 配置 86
1.10.1. 创建 CORS 规则 86
1.10.2. 管理 CORS 规则 88
1.10.3. 绑定/解绑 API 91
1.11. 接口编排 92
1.11.1. 创建接口编排 92
1.12. 全局设置 99
1.12.1. 跨域设置 99
1.12.2. 注册中心设置 101
2.订阅 API 105
2.1. 创建应用 105
2.2. 数据加密 105
2.3. 管理应用 110
3.最佳实践 113
3.1. 双网关配置 113

3.2. 代码生成 117
3.2.1. 总览 118
3.2.2. SOFARPC 生成 Proto 文件 120

1.发布 API

1.1. API 管理

1.1.1. API 属性说明

本文将列出在 API 网关控制台创建并发布 API 时，您需要配置的相关属性。

API 定义

基本信息

属性	必填项	说明
API 名称	是	• 用于识别 API。 • 支持中文、英文、数字、下划线（_）、连接符（-），且只能以英文和中文开头。32 个字符以内。 • 同一个 工作空间下 API 名称不能相同。
API 分组	是	• 用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。 • 同一个环境下，API 分组名称不能相同。 • API 分组名称：支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。
API 描述	否	用于描述 API 的作用等，64 个字符以内。
应用	是	• 应用：表示 API 发布后支持 web 浏览器、H5 容器、小程序、JS、业务系统的调用。 • mPaas移动应用：表示 API 发布后支持 mPaaS 移动客户端（iOS、Android）的调用。 • 应用以及 mPaas 移动应用：表示 API 发布后支持 web 浏览器、H5 容器、小程序、JS、业务系统的调用，同时也支持 mPaaS 移动客户端（iOS、Android）的调用。
OperationType	是	• 针对 mPaaS 移动应用设置的 API 服务标识，用于定位要访问的资源。 • 支持英文字母、数字、英文句号（.），200 字符以内。 • 需按照 `com.xxx.opt1` 的格式输入。

请求路径	是	• 表示请求的资源的 URL，通过请求路径可以定位到要请求的资源。 • 支持英文字母、数字、下划线（_）、连接符（-），200 字符以内。 • 以斜杠（/）开头，不能以斜杠（/）结尾，格式示例：`/getUserInfo`。 • 支持占位符模式 `/home/{id}`，如果需要占位符透传控制面，API 的后端请求路径需要为空。
HTTP 请求方法	是	表明要对给定的 HTTP 资源执行的操作： • GET：检索服务器中的数据。 • POST：将数据发送到服务器进行处理。 • PUT：更新服务器中存储的数据。 • DELETE：删除服务器中的数据。
路径匹配规则	是	• 绝对匹配：调用时完全匹配以上填写的路径，例如：`/path`。 • 前缀匹配：以请求路径为前缀，可匹配多个前缀一样的不同路径，例如：填写 `/path` 的匹配效果为 `/path/*`。
安全认证	是	• 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。 • 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
接口名称	是	• 客户端 RPC 服务或者 DUBBO 服务需要订阅的接口名称。 • API 接口名称，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头。 • 格式示例：`com.alipay.testapp.facade.ConfigFacade:1.0@DEFAULT`。
方法名	是	• 客户端 RPC 服务或者 DUBBO 服务需要订阅的接口名称。 • API 服务方法名，支持英文字母、数字组成，以字母作为开头。 • 格式示例：`getUserID`。
报文类型	是	表示请求和响应中的媒体类型信息，用来告诉服务端如何处理请求的数据，以及告诉客户端（一般是浏览器）如何解析响应的数据。
报文编码	是	表示客户端接受文本内容的字符集样式。

实体参数	否	客户端 DUBBO 服务接口方法使用的实体参数全路径，例如：Ljava/lang/String，自定义参数实体时注意以分号（；）结尾。
返回参数	否	客户端 DUBBO 服务接口方法返回参数全路径，例如：Ljava/lang/String，自定义参数实体时注意以分号（；）结尾。
DUBBO 分组	否	客户端 DUBBO 服务接口订阅时的服务分组。

请求参数

请求参数表示客户端向网关发起请求时要配置的参数。

属性	必填项	说明
参数名	否	参数的唯一标识。
参数位置	否	• **Header**：自定义 API 请求头。例如 `X-MyHeader: Value` 。 • **Path**：参数为请求 URL 路径的一部分。例如 `/users/{id}` 。 • **Query**：参数在请求 URL 末尾的问号 `?` 之后以 `name=value` 形式出现。参数之间以 `&` 分隔。例如 `/users?role=admin&id=12345` 。
类型	否	• String • Int • Long • Double • Float • Boolean
默认值	否	在 API 请求的参数值为空时使用的默认值。 默认值类型必须和参数类型一致。
描述	否	参数的业务描述。

请求 Body 类型	否	请求 Body 类型表示客户端向网关发起请求时要配置的参数，可以依赖已经创建数据模型。 API 在使用编排功能时必须选择数据模型才能实现参数转换以及透传，在 API 分组中若想要生成 SDK 必须在请求 Body 或者响应 Body 选择数据模型才能成功。 • Long • Interger • Float • Double • Boolean • Byte • Short • String • List：可以选择基础类型或者数据模型作为参数模板。 • Object：需要选择数据模型作为参数模板。 • Map：可以选择基础类型或者数据模型作为参数模板。

响应参数

属性	必填项	说明
成功响应示例	否	为 API 订阅者提供请求成功时的返回数据参考。
失败响应示例	否	为 API 订阅者提供请求失败时的返回数据参考。
错误码	否	HTTP 默认错误代码。
错误信息	否	自定义错误提示信息。
描述	否	错误码的业务描述。

响应body类型	否	响应的参数，可以依赖数据模型 API 在使用编排功能时必须选择数据模型才能实现参数转换以及透传，在 API 分组中若想要生成 SDK 必须在请求 Body 或者响应 Body 选择数据模型才能成功。 • Long • Interger • Float • Double • Boolean • Byte • Short • String • List：可以选择基础类型或者数据模型作为参数模板。 • Object：需要选择数据模型作为参数模板。 • Map：可以选择基础类型或者数据模型作为参数模板。

后端配置

属性	必填项	说明
后端服务类型	是	表示网关接收到请求后转发给的后端服务类型： • Mock：表示如果接口后端还没有提供，使用 Mock 用于模拟一个后端服务。 • 系统集群：表示网关接收到前端请求后转发到真实业务系统的集群。
后端协议类型	是	表示网关接收到请求后转发给的后端服务使用的通信协议类型： • HTTP • SOFARPC • SOFAREST • DUBBO • DUBBOX • TRIPLE • SOFA_TR
请求路径	否	服务端的实际请求路径，不填默认使用 API 定义中的请求路径。
超时时间	是	API 请求超时时间，单位为毫秒（ms），默认 3000 毫秒。

路由策略	是	表示当网关接收到语法后使用的路由策略： • 根据 Header 路由：根据 Header 里带的参数路由 • 根据 LDC 路由：使用 LDC 单元化路由。 • 根据权重路由：数值范围为 0-100，所有规则加起来为 100。 • 根据请求路径路由：直接转发。
接口名称	是	• 服务端 RPC 服务、SOFAREST、DUBBO 服务发布的接口名称。 • API 接口名称，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头。 • 格式示例：`com.alipay.testapp.facade.ConfigFacade:1.0@DEFAULT` 。
方法名	是	• 服务的 RPC 服务或者 DUBBO 服务需要订阅的接口方法名称。 • API 服务方法名，支持英文字母、数字组成，以字母作为开头。 • 格式示例：`getUserID` 。
实体参数	否	服务端 DUBBO 服务接口方法使用的实体参数全路径，例如：Ljava/lang/String，自定义参数实体时注意以分号（；）结尾。
返回参数	否	服务端 DUBBO 服务接口方法返回参数全路径，例如：Ljava/lang/String，自定义参数实体时注意以分号（；）结尾。
DUBBO 分组	否	服务端 DUBBO 服务接口订阅时的服务分组。
是否跨域	是	是否开启跨域网关开关，若开启只能选择开启跨域的路由规则进行路由到入口网关。
入口网关-分组ID/分组自定义域名	否	开启跨域后，可以自定义入口网关-分组ID/分组自定义域名用作在入口网关进行匹配分组。
是否pub至注册中心	否	开启后，API 发布会直接 Pub 到注册中心，默认只会 Pub 到 SOFARegistry，关闭则不会主动 Pub 到 SOFA 注册中心。
是否灰度	否	是否开启全链路灰度，依赖 DRS，若开启会根据 instance_id 从 DRS 注册中心匹配路由规则。

1.1.2. 创建 API

在 API 网关中，API 发布者将应用的服务接口注册并对外发布后才可以被外部客户端调用。本文介绍如何快速在控制台创建并发布一个 API 服务。

创建 HTTP API

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > API 管理**。
2. 在 API 列表页，单击列表右上方的 **创建 API**。
3. 在新页面中，选择 **HTTP API** 类型，单击 **创建**。
4. 在 **定义 API** 步骤中，您需要配置以下信息：
   - **API 分组**：必填，用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。

     > **说明**
     >
     > 您可以选择从下拉菜单中选择已有分组，也可以单击 **创建分组** 新建一个 API 分组。详见 创建 API 分组。

   - **API 名称**：必填，用于识别 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。

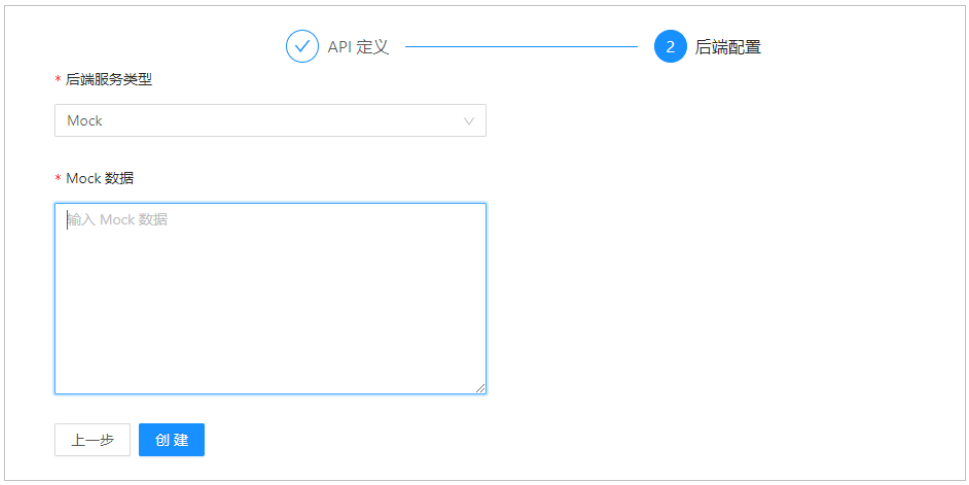
     > **说明**
     >
     > 同一个 API 工作空间下，API 名称不能相同。

   - **描述**：选填，用于描述 API 的作用等，64 个字符以内。
   - **API 授权应用类型**：必选，指定可以订阅并调用该 API 的应用类型。
     - mPaaS 移动应用：表示 API 发布后支持 mPaaS 移动客户端（iOS、Android）的调用。
     - 应用以及 mPaaS 移动应用：表示 API 发布后支持 Web 浏览器、H5 容器、小程序、JS、业务系统的调用，同时也支持 mPaaS 移动客户端（iOS、Android）的调用。
     - 应用：表示 API 发布后支持 web 浏览器、H5容器、小程序、JS、业务系统的调用。
   - **OperationType**：必填，针对 mPaaS 移动应用设置的 API 服务标识，用于定位要访问的资源。
     - 支持英文字母、数字、.，200 字符以内。
     - 需按照 `com.xxx.opt1` 的格式输入。
   - **请求路径**：必填，针对应用设置的请求资源的 URL，通过请求路径可以定位到要请求的资源。
     - 支持英文字母、数字、下划线（_）、连接符（-），200 字符以内。
     - 以斜杠 / 开头，不能以斜杠 / 结尾，格式示例： `/getUserInfo` 。
   - **路径匹配规则**：如订阅并调用该 API 的应用类型为 **应用** 时，必填。
     - 绝对匹配：表示只有请求的路径和配置的路径完全一样时才能匹配成功。
     - 前缀匹配：如希望带有某个前缀的请求全部匹配到某一个 API，则可以使用前缀匹配。
   - **方法**：必填，表明要对给定的 HTTP 资源执行的操作：
     - **GET**：检索服务器中的数据。

- POST：将数据发送到服务器进行处理。
- PUT：更新服务器中存储的数据。
- DELETE：删除服务器中的数据。

- **安全认证**：必选，根据实际需要自行选择。
  - 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。
  - 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
- **请求参数**：可选，详见 API 属性说明 > 请求参数。
- **响应参数**：可选，详见 API 属性说明 > 响应参数。

5. 单击 **下一步**，进入 **后端配置** 页面，您需要选择 **后端配置类型** 并完成后端配置信息
   - **Mock**：表示如果接口后端还没有提供，使用 Mock 模拟一个后端服务。
     - Mock 数据：必填，输入 Mock 的后端服务数据，JSON 格式。



   - **系统集群**：表示网关接收到前端请求后转发到真实业务系统的集群。

- **协议类型**：必选，表示网关接收到请求后转发给的后端服务使用的通信协议类型。协议不同，需要配置的信息也有所不同。
  - HTTP

    请求路径：必填，API 请求所指向的资源 URL。以斜杠 / 开头，支持字母、数字、下划线（_）、连接符（-），200字符以内。
  - SOFARPC
    - 接口名称：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
    - 服务标识：选填，支持指定服务的 uniqueId。
    - 方法名：支持英文字母、数字组成，以小写字母作为开头，格式示例：`getUserID`。
  - SOFAREST
    - 接口名称：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
    - 服务标识：选填，支持指定服务的 uniqueId。
    - 请求路径：必填，API 请求所指向的资源 URL。以斜杠 / 开头，支持字母、数字、下划线（_）、连接符（-），200 字符以内。，格式示例：`/home/{id}`。
  - DUBBO/DUBBOX
    - 接口名称：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
    - 实体参数：参数实体指的是参数的全路径，例如：Ljava/lang/String；自定义参数实体时注意以 `;` 结尾，以 `L` 开头。
    - 服务版本：dubbo接口服务发布的版本号,可以参考dubbo发布服务xml示例
    - DUBBO分组：dubbo接口服务发布的分组，可以参考dubbo发布服务xml示例
    - 方法名：支持英文字母、数字组成，以小写字母作为开头，格式示例：`getUserID`。

    dubbo 服务发布 xml 实例

    ```
    <beanid="testService"class="com.dap.provider.TestServiceImpl"/>
    <dubbo:serviceref="testService" interface="com.dap.api.IService"
    group="dubbo分组" version="服务版本"timeout="5000"/>
    ```

- **超时时间**：必填，API 请求超时时间，单位为毫秒（ms）。
- **路由策略**：必选，表示当网关接收到请求后使用的路由策略。详见 创建路由规则。
  - 根据 Header 路由：根据 Header 里带的参数路由。
  - 根据权重路由：数值范围为 0-100，所有规则加起来为 100。
  - 根据请求路径/接口路由：即直接转发。

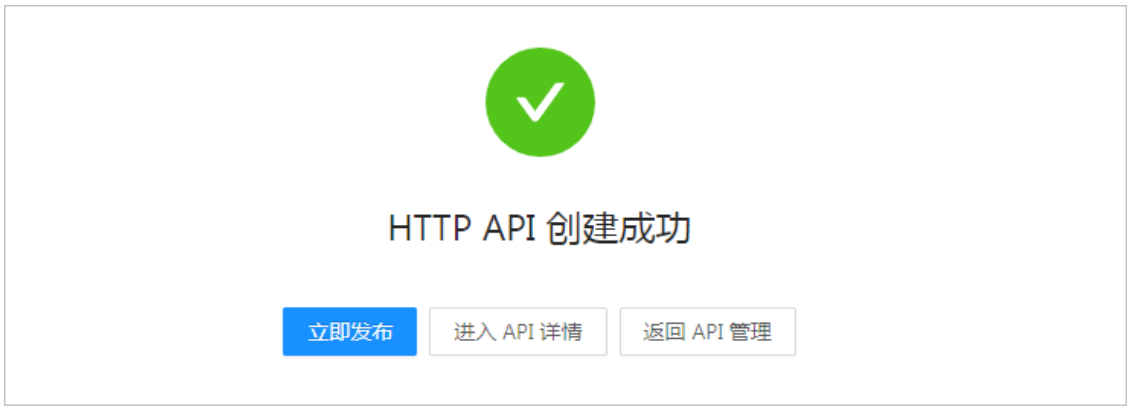
- 根据 LDC 路由：即 LDC 单元化路由转发，仅支持协议类型为 SOFARPC 或 SOFAREST 时可选。

> **说明**
>
> 使用单元化路由转发前，需要在单元化应用服务中创建统一接入实例，具体内容请参见 创建统一接入实例。

- **系统集群**：必选，选择当前环境内已配置好的系统集群。您也可以单击 **创建系统集群** 直接新建一个系统集群。详见 创建系统集群。
- **响应参数映射自定义配置**：可选，支持从下拉菜单中选择现有的响应参数映射规则，或单击 **新建参数映射规则** 直接创建一个响应参数映射规则。更详细的创建步骤，参见 创建参数映射。
- **请求参数映射配置**：可选，支持选择 **参数名映射** 或 **自定义配置**。选择 **自定义配置**，即可使用现有的请求参数映射规则。

6. 单击 **创建** 即可。

HTTP API 创建完成后，您可以单击 **立即发布** 发布该 API，单击 **进入 API 详情** 查看 API 详细信息，或单击 **返回 API 管理** 回到 API 列表页。



创建 SOFARPC API（协议转换）

1. 在 **API 发布 > API 管理** 页面，单击 API 列表右上方的 **创建 API**。
2. 在新页面中，选择 **SOFARPC API（协议转换）** 类型，单击 **创建**。
3. 在 **定义 API** 步骤中，您需要配置以下信息：
   - **API 分组**：必选，用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。
   - **API 名称**：必填，用于识别 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。

     > **说明**
     >
     > 同一个 API 分组下，API 名称不能相同。

   - **描述**：选填，用于描述 API 的作用等，64 个字符以内。
   - **接口名称**：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。

- **服务标识**：选填，支持通过 uniqueId 指定服务。
- **方法名**：支持英文字母、数字组成，以小写字母作为开头，格式示例：`getUserID`。
- **报文类型**：必选，选择 `application/json`。
- **安全认证**：必选，根据实际需要自行选择。
  - 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。
  - 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
- **是否pub至注册中心**：选填，开启后api发布会支持pub到注册中心，目前只支持sofa注册中心
- **请求参数**：可选，详见 API 属性说明 > 请求参数。

4. 单击 **下一步**，进行后端配置：
   - **无**：表示网关接收到前端请求后，不转发到业务集群。
   - **Mock**：表示如果接口后端还没有提供，使用 Mock 用于模拟一个后端服务。
     - Mock 数据：必填，输入 Mock 的后端服务数据，JSON 格式。
   - **系统集群**：表示网关接收到前端请求后转发到真实业务系统的集群。

- 协议类型：必选，表示网关接收到请求后转发给的后端服务使用的通信协议类型。
  - **HTTP**
    - 请求路径：API 请求所指向的资源 URL。以斜杠 / 开头，支持字母、数字、下划线（_）、连接符（-），200字符以内。
    - 超时时间：必填，API 请求超时时间，单位为毫秒（ms）。
    - 方法：HTTP请求方法：GET\POST\PUT\DELETE
    - 报文类型：http服务报文类型，目前网关支持application/json
    - 报文编码：http服务报文编码，目前网关支持UTF-8
    - 路由策略：必选，表示当网关接收到语法后使用的路由策略。详见 创建路由规则。
      - 根据 Header 路由：根据 Header 里带的参数路由。
      - 根据权重路由：数值范围为 0-100，所有规则加起来为 100。
      - 根据请求路径/接口路由：直接转发。
    - 系统集群：必选，选择当前环境内已配置好的系统集群，或单击 **创建系统集群** 直接新建一个系统集群。详见 创建系统集群。
  - **DUBBO/DUBBOX**
    - 接口名称：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
    - 实体参数：参数实体指的是参数的全路径：例如：Ljava/lang/String; 自定义参数实体时注意以；结尾以L开头
    - 服务版本：dubbo接口服务发布的版本号,可以参考dubbo发布服务xml示例
    - DUBBO分组：dubbo接口服务发布的分组，可以参考dubbo发布服务xml示例
    - 方法名：支持英文字母、数字组成，以小写字母作为开头，格式示例：`getUserID`。

    dubbo服务发布xml实例

5. 单击 **创建**。

SOFARPC API 创建完成后，您可以单击 **立即发布** 发布该 API，单击 **进入 API 详情** 查看 API 详细信息，或单击 **返回 API 管理** 回到 API 列表页。

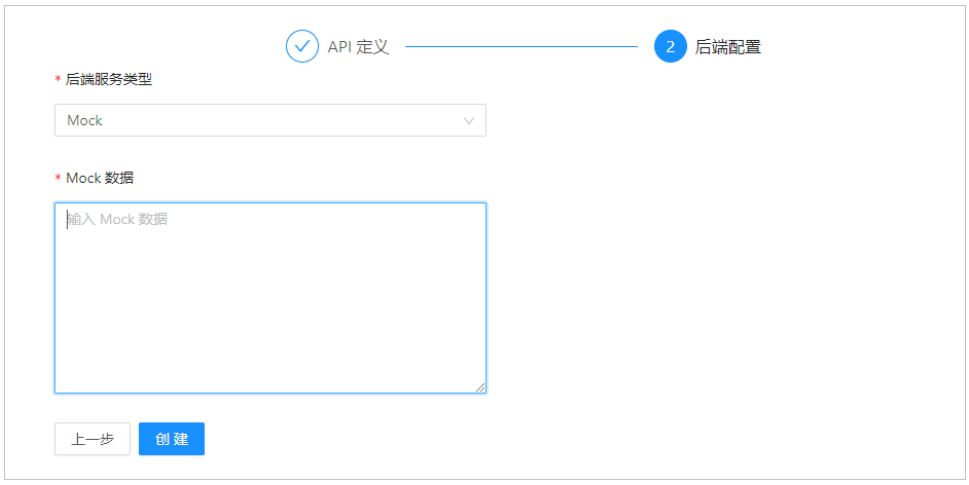
创建 DUBBO API

1. 在 **API 发布 > API 管理** 页面，单击 API 列表右上方的 **创建 API**。
2. 在新页面中，选择 **DUBBO API** 类型，单击 **创建**。
3. 在 **创建 DUBBO API** 页面，您需要配置以下信息：
   - **API 分组**：必选，用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。
   - **API 名称**：必填，用于识别 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。

> **说明**
>
> 同一个 API 分组下，API 名称不能相同。

- **描述**：选填，用于描述 API 的作用等，64 个字符以内。
- **接口名称**：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
- **实体参数**：选填，参数实体指的是dubbo客户端订阅接口服务参数的全路径：例如：Ljava/lang/String; 自定义参数实体时注意以；结尾，以L开头
- **返回参数**：选填，dubbo服务的返回参数全路径例如：Ljava/lang/String; 自定义参数实体时注意以；结尾，以L开头
- **DUBBO 分组**：选填，作为 客户端Dubbo服务分组，主要用于服务发现。
- **方法名**：支持英文字母、数字组成，以小写字母作为开头，格式示例：`getUserID`。
- **报文类型**：必选，目前仅支持 `application/json`。
- **报文编码**：必选，目前仅支持 `UTF-8`。
- **安全认证**：必选，根据实际需要自行选择。
  - 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。
  - 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
- **请求参数**：可选，详见 API 属性说明 > 请求参数。

4. 单击 **下一步**，进入 **后端配置** 页面，您需要选择 **后端配置类型** 并完成后端配置信息：
   - **Mock**：表示如果接口后端还没有提供，使用 Mock 模拟一个后端服务。
     - Mock 数据：必填，输入 Mock 的后端服务数据，JSON 格式。



   - **系统集群**：表示网关接收到前端请求后转发到真实业务系统的集群。

- **协议类型**：必选，表示网关接收到请求后转发给的后端服务使用的通信协议类型。协议不同，需要配置的信息也有所不同。
  - HTTP
    - **请求路径**：必填，API 请求所指向的资源 URL。以斜杠 / 开头，支持字母、数字、下划线（_）、连接符（-），200字符以内。
    - **方法**：必填，表明要对给定的 HTTP 资源执行的操作：
      - GET：检索服务器中的数据。
      - POST：将数据发送到服务器进行处理。
      - PUT：更新服务器中存储的数据。
      - DELETE：删除服务器中的数据。
    - **报文类型**：必选，目前仅支持 `application/json` 。
    - **报文编码**：必选，目前仅支持 `UTF-8` 。
  - SOFARPC
    - **接口名称**：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例： `com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT` 。
    - **服务标识**：选填，支持指定服务的 uniqueId。
    - **方法名**：支持英文字母、数字组成，以小写字母作为开头，格式示例： `getUserID` 。
- **超时时间**：必填，API 请求超时时间，单位为毫秒（ms）。
- **路由策略**：必选，表示当网关接收到请求后使用的路由策略。详见 创建路由规则。
  - 根据 Header 路由：根据 Header 里带的参数路由。
  - 根据权重路由：数值范围为 0-100，所有规则加起来为 100。
  - 根据请求路径/接口路由：即直接转发。
  - 根据 LDC 路由：即 LDC 单元化路由转发，仅支持协议类型为 SOFARPC 或 SOFAREST 时可选。

  > **说明**
  > 使用单元化路由转发前，需要在单元化应用服务中创建统一接入实例，具体内容请参见 创建统一接入实例。

- **系统集群**：必选，选择当前环境内已配置好的系统集群。您也可以单击 **创建系统集群** 直接新建一个系统集群。详见 创建系统集群。

5. 完成后单击 **创建**，即可完成一个 DUBBO API 的创建。

创建双网关跨云 API

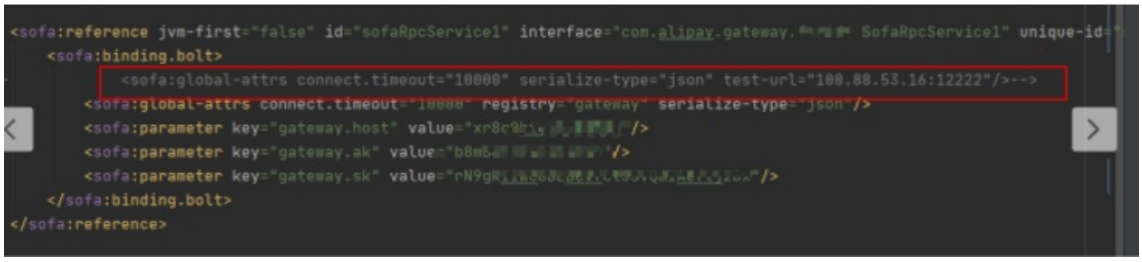
通过在两个网络隔离的区域各部署一个网关，可以实现两个原本网络隔离区域之间的网络互通。

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > API 管理**。
2. 在 API 列表页，单击列表右上方的 **创建 API**。

3. 在新页面中，选择 **双网关跨云 API** 类型，单击 **创建**。
4. 在 **API 定义** 步骤中，您需要配置以下信息：
   - **所属分组**：必选，用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。
   - **API 名称**：必填，用于识别 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。
   - **描述**：选填，用于描述 API 的作用等，64 个字符以内。
   - **出口网关协议类型**：必选，是调用端发起调用的协议类型。支持 HTTP 和 SOFARPC 两种类型。
   - **入口网关协议类型**：必选，是后端系统集群使用的协议类型。支持 HTTP 、SOFARPC 两种类型。

     > **说明**
     >
     > 入口网关协议类型种类个数由出口网关协议类型决定。
     >
     > - 出口网关协议类型为 HTTP 时，入口网关协议类型支持 HTTP、SOFARPC 和 SOFAREST 三种类型。
     > - 出口网关协议类型为 SOFARPC 时，入口网关协议类型支持 HTTP、SOFARPC 两种类型。

     当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，网关需要采取直接连接的模式，如下图：

     > **说明**
     >
     > `test-url` =网关地址:12222

   - **API 授权应用类型**：必选，指定可以订阅并调用该 API 的应用类型。
     - 应用以及 mPaaS 移动应用：（仅专有云）表示 API 发布后支持 Web 浏览器、H5 容器、小程序、JS、业务系统的调用，同时也支持 mPaaS 移动客户端（iOS、Android）的调用。
     - mPaaS 移动应用：（仅专有云）表示 API 发布后支持 mPaaS 移动客户端（iOS、Android）的调用。
     - 应用：表示 API 发布后支持 web 浏览器、H5容器、小程序、JS、业务系统的调用。
   - **OperationType**：当授权应用类型包含 mPaaS 移动应用时，必填，是针对 mPaaS 移动应用设置的 API 服务标识，用于定位要访问的资源。
     - 支持英文字母、数字、.，200 字符以内。
     - 需按照 `com.xxx.opt1` 的格式输入。
   - **请求路径**：必填，是针对应用设置的请求资源的 URL，通过请求路径可以定位到要请求的资源。

- 支持英文字母、数字、下划线（_）、连接符（-），200 字符以内。
- 以斜杠 / 开头，不能以斜杠 / 结尾，格式示例： `/getUserInfo` 。

- **路径匹配规则**：如订阅并调用该 API 的应用类型为 **应用** 时，必填。
  - 绝对匹配：表示只有请求的路径和配置的路径完全一样时才能匹配成功。
  - 前缀匹配：如希望带有某个前缀的请求全部匹配到某一个 API，则可以使用前缀匹配。
- **接口名称**：出口网关协议类型为 SOFARPC时，必填，是服务的接口类型，支持英文字母、数字、英文逗号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。
- **服务标识**：出口网关协议类型为 SOFARPC时，选填，支持英文字母、数字、英文逗号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。

> **说明**
>
> 若服务标识不为空，接口名称中优先使用服务标识作为 uniqueId。

- **方法名**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必填，服务方法名，以小写英文字母开头，格式示例：getUserID。
- **方法**：当出口网关类型为 HTTP 时，必填，表明要对给定的 HTTP 资源执行的操作：
  - **GET**：检索服务器中的数据。
  - **POST**：将数据发送到服务器进行处理。
  - **PUT**：更新服务器中存储的数据。
  - **DELETE**：删除服务器中的数据。
- **报文类型**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必选，仅支持 application/json。
- **报文编码**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必选，仅支持 UTF-8。
- **出口网关和入口网关**：必选，选择在 跨域设置中配置好的网关信息。

> **说明**
>
> 注册中心类型仅支持 NONE。

- **安全认证**：必选，根据实际需要自行选择。
  - 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。
  - 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
- **请求参数**：可选，详见 API 属性说明 > 请求参数。
- **响应参数**：可选，详见 API 属性说明 > 响应参数。

> **说明**
>
> 当入口网关类型和出口网关类型都为 SOFARPC 时，不需要进行后端配置。

5. 单击 **下一步**，进入 **后端配置** 页面，完成后端配置信息。

- **后端服务类型**：无法编辑，默认为系统集群。
- **协议类型**：无法编辑，为上一步中入口网关协议类型。
- **接口名称**：入口网关协议类型为 SOFARPC 或 SOFAREST 时，必填，是服务的接口类型，支持英文字母、数字、英文逗号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。
- **服务标识**：入口网关协议类型为 SOFARPC 或 SOFAREST 时，选填，支持英文字母、数字、英文逗号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。

  > **说明**
  >
  > 若服务标识不为空，接口名称中优先使用服务标识作为 uniqueId。

- **方法名**：当入口网关类型为 SOFARPC，必填，服务方法名，以小写英文字母开头，格式示例：getUserID。
- **请求路径**：请求到后端时显示的路径，必须以 / 开头，不能以 / 结尾。
- **超时时间**：必填，API 请求超时时间，单位为毫秒（ms）。
- **路由策略**：必选，表示当网关接收到请求后使用的路由策略。详见 创建路由规则。
  - 根据 Header 路由：根据 Header 里带的参数路由。
  - 根据权重路由：数值范围为 0-100，所有规则加起来为 100。
  - 根据请求路径/接口路由：即直接转发。
  - 根据 LDC 路由：即 LDC 单元化路由转发，仅支持协议类型为 SOFARPC 或 SOFAREST 时可选。

  > **说明**
  >
  > 使用单元化路由转发前，需要在单元化应用服务中创建统一接入实例，具体内容请参见 创建统一接入实例。

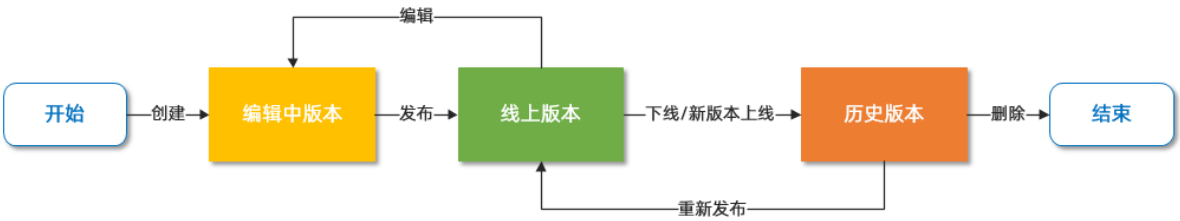
- **系统集群**：必选，选择当前环境内已配置好的系统集群，支持选择跨 VPC 集群。您也可以单击 **创建系统集群** 直接新建一个系统集群。详见 创建系统集群。
- **响应参数映射自定义配置**：可选，支持从下拉菜单中选择现有的响应参数映射规则，或单击 **新建参数映射规则** 直接创建一个响应参数映射规则。更详细的创建步骤，参见 创建参数映射。
- **请求参数映射配置**：可选，支持选择 **参数名映射** 或 **自定义配置**。选择 **自定义配置**，即可使用现有的请求参数映射规则。

6. 完成后，单击 **创建**，完成双网关跨云 API 的创建。

1.1.3. API 版本管理

API 只有在发布后才能被订阅方使用。创建后的 API 可以根据业务需求进行编辑、下线和编辑重发布等操作。新版本发布上线后，老版本自动成为历史版本被下线。

API 版本的生命周期如下：



版本说明：

- 每次发布一个版本时，系统为当前版本分配一个新的版本号。
- 每次保存草稿时，系统为未发布草稿分配一个新的版本号。
- 同一 API 只能存在一个编辑中版本。新保存的草稿自动覆盖未发布的已有草稿。

查看版本

在 **API 管理** 页面，单击目标 API 进入其详情页。在 **版本管理** 标签页下，您可以查看到当前线上版本、编辑中版本以及历史发布版本，如下图所示。

发布版本

您可以对线上版本与历史版本重新编辑再发布，也可以直接发布编辑中的版本。

重新编辑并发布线上版本

1. 在 **版本管理 > 线上版本** 中，单击 **操作** 列的 **重新编辑** 按钮。
2. 在编辑页面，修改 API 信息后，单击 **保存**。
3. 在 API 编辑的内容保存成功页面，单击 **立即发布**。
4. 在 **发布** 窗口，您可以选择输入具体版本原因，单击 **确定**，即可发布成功。

发布编辑中版本

1. 在 **版本管理 > 编辑中版本** 中，单击 **操作** 列的 **发布** 按钮。
2. 在 **发布** 窗口，输入具体版本原因，单击 **确定**，即可发布成功。

编辑重发布历史版本

1. 在 **版本管理 > 历史版本** 中，单击 **操作** 列的 **编辑重发布** 按钮。
2. 在编辑页面，修改 API 信息后，单击 **保存**。
3. 在 API 编辑的内容保存成功页面，单击 **立即发布**。
4. 在 **发布** 窗口，您可以选择输入具体版本原因，单击 **确定**，即可发布成功。

下线版本

您可以将处于已发布状态的 API 下线。下线后， API 状态切换为 **未发布**，无法被该 API 的订阅方应用继续调用。您可以通过以下任一方法下线 API 当前版本：

- 在 API 列表中选择要下线的 API，单击 **下线 > 确认**。
- 在 API 详情页面，单击右上角的 **下线** 按钮。
- 在 API 详情 > **版本管理** 页签中，单击 **线上版本** 右侧的 **下线** 按钮。

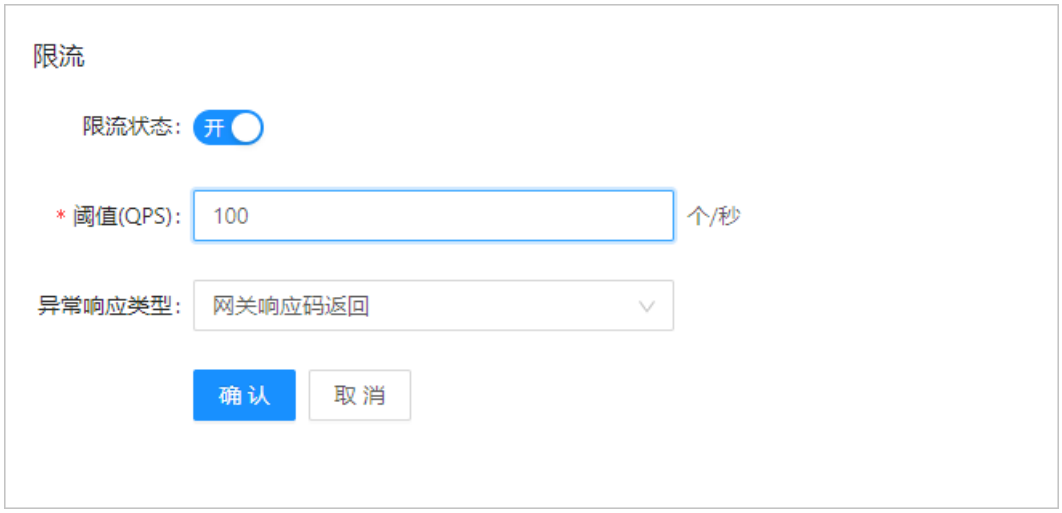
1.1.4. API 流量治理

在 API 详情页，您可以对该 API 进行流量治理，即配置限流与缓存。

限流

限流指对网关到后端集群的 QPS 进行限制。操作步骤如下：

1. 在 API 详情页，切换至 **流量治理** 标签页。
2. 单击 **限流** 右侧图标，开始编辑具体限流配置：
   - 限流状态：将状态切换为 **开** 状态。
   - 阈值(QPS)：指网关到后端集群的 QPS，框内的值只能是大于 0 的正整数。
   - 异常响应类型：支持选择 **返回数据** 与 **网关响应码返回** 两个选项。
3. 单击 **确认**。



缓存

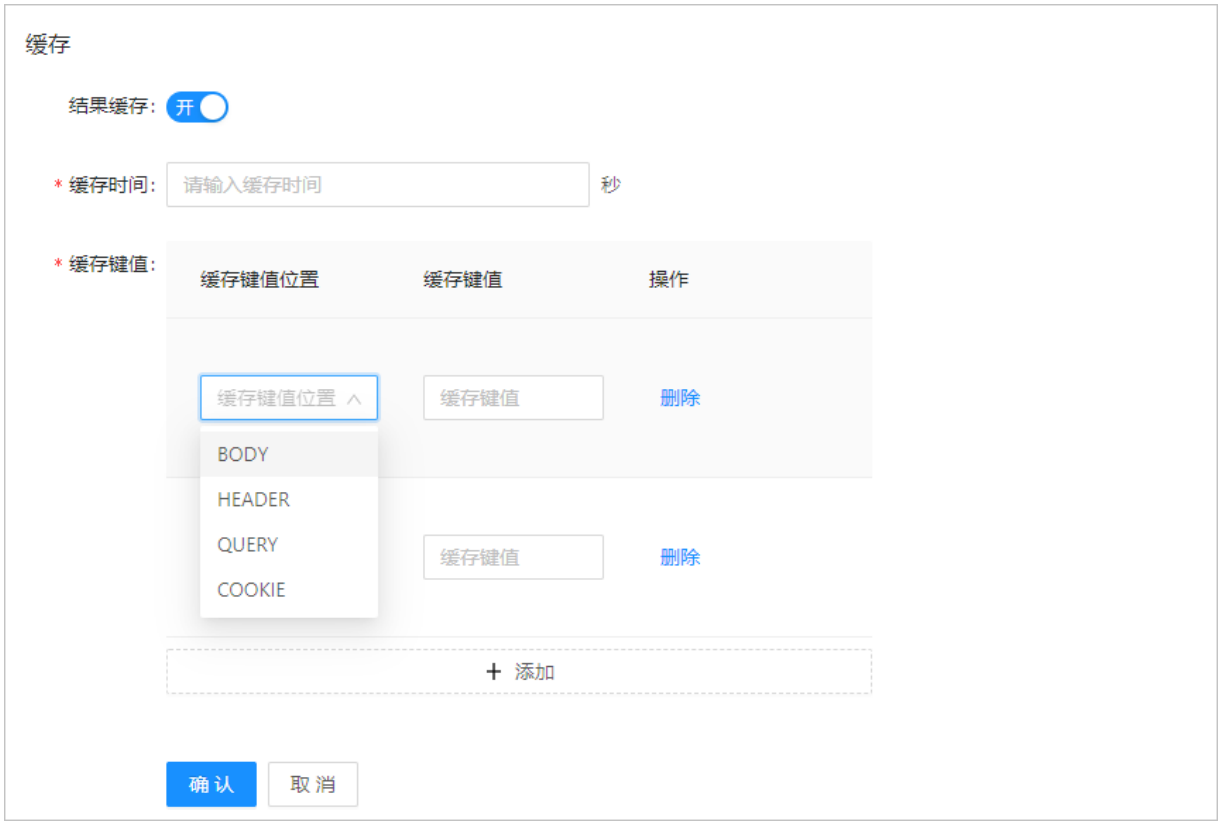
缓存是指网关将后端的内容缓存在网关上，降低对后端 Server 的压力。

> **说明**
>
> 关于 API 缓存配置的更多详细规则，参见 API 缓存配置。

操作步骤如下：

1. 在 API 详情页，切换至 **流量治理** 标签页。
2. 单击 **缓存** 右侧图标，开始编辑具体限流配置：
   - 结果缓存：将状态切换为 **开** 状态。
   - 缓存时间：将内容缓存在网关上的时间，单位为秒（s）。最长支持 5 分钟。
   - 缓存键值：需要配置缓存键值位置与具体缓存键值，支持从 HEADER、QUERY、COOKIE 和 BODY 中进行选择。

     > **说明**
     >
     > 如果选择 BODY，BODY 必须是一个 JSON 字符串，暂时不支持 pb 格式的 BODY。
3. 单击 **确认**。



1.1.5. 绑定/解绑授权对象

只有获得授权的订阅方应用可以调用 API。在 API 详情页的 **授权对象** 标签页下，您可以绑定授权对象、解除绑定以及限制应用访问 API 的次数。

绑定授权对象

1. 进入 **API 发布 > API 管理** 页面，找到待授权的 API，单击进入其详情页。
2. 在 **授权对象** 标签页下，单击 **绑定授权对象**，进入 **添加授权应用** 窗口。
   - 对于已添加为授权对象的订阅方应用，您可以直接在授权对象列表中搜索找到。
   - 对于暂未添加为授权对象的订阅方应用，您需要先为该应用创建一个授权对象。详见 创建授权对象。

3. 在授权对象的列表中，选择待授权对象，单击 **确定**，完成绑定。

解除绑定

在 API 详情页 > **授权对象** 标签页下，找到需要解除绑定的授权对象，单击其 **操作** 列的 **解除绑定** 按钮，并确定即可。

限流配置

您可以对 API 绑定的应用进行限流，即指定每秒该应用对该 API 可访问的次数。

1. 在 API 详情页 > **授权对象** 标签页下，找到需要限流的授权对象，单击其 **操作** 列的 **限流配置** 按钮。
2. 在新窗口中，进行限流配置后，单击 **确定**。
   - 限流开关：将其状态设置为 **开**。
   - 限流阈值：必填，即每秒该应用对该 API 可访问的次数，只能输入大于 0 的正整数。

- 异常响应类型：必选，支持 **返回数据** 与 **网关响应码返回**。

1.1.6. API 调试

对于已发布的 API，您可以在该 API 的详情页进行 API 调试。

> **说明**
>
> 目前仅支持调试 HTTP 类型的 API 服务。

操作步骤

1. 在 **API 发布 > API 管理** 页面，单击待调试的 API 名称，进入其详情页。
2. 目前只有HTTP支持api调试
3. 在 API 详情页，切换至 **API 调试** 标签。
4. 在 **请求数据** 区域，配置如下请求数据：
   - 请求响应头：配置请求参数名、参数位置以及具体的参数值。参见 API 属性说明 > 请求参数。
   - 请求body: POST请求时参数需要以json的形式填写
   - 身份认证：选择调试的目标授权对象。

5. 单击 **发送请求**

1.1.7. 导入导出 API

您可以在 API 网关控制台上以 JSON 文件的形式对 API 进行批量导入及导出。

> **说明**
>
> - 为减少操作耗时，建议一次性导入/导出 API 的数量不超过 100 个。
> - 导入 API 的 JSON 格式请参见模板 API Json 文件格式。

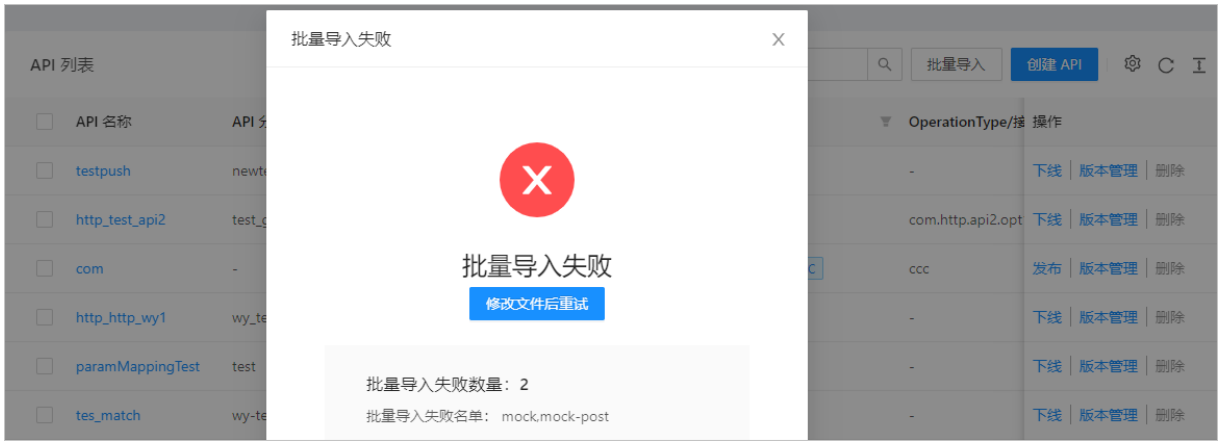
批量导出 API

1. 在 **API 发布 > API 管理** 页面，在 API 列表左侧勾选需要导出的 API。
2. 单击 API 列表右下角的 **批量导出** 按钮，即可将选择的 API 转换为 JSON 文件保存至本地。

批量导入 API

1. 在 **API 发布 > API 管理** 页面，单击 API 列表右上方的 **批量导入** 按钮。
2. 在弹出的窗口中，选择本地的 JSON 文件进行 API 导入。

如果 API 导入失败，您可以在弹出的窗口中查看到导入失败的 API 数量及名称，如下图所示。



导入失败，大多情况是因为导入的 API 数据与当前选择的租户、工作空间下的现有 API 数据重复冲突了。您可以修改本地文件中相应的 API 配置数据，继而尝试重新上传导入。

1.1.8. 查看/生成 API 文档

为了让用户更好的使用网关维护 API 接口，满足输出 API 文档的需求，网关支持针对 API 进行查看和下载文档的功能。

操作步骤

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布 > API 管理**，进入 API 管理页面。
3. 在 API 列表右侧单击 **查看文档**。

4. 在跳转页面可以看到该 API 的接口文档。

5. 您可以单击页面左上角 **点击下载**，将接口文档下载至本地。

> **说明**
>
> 目前仅支持的下载格式为 Word。

1.2. API 分组

1.2.1. 创建 API 分组

API 分组将网关上的 API 按照业务需求进行分组管理。同一分组下的 API 共享一个域名地址。

您可以通过以下任一方式创建 API 分组：

在分组列表中创建 API 分组

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **API 分组**，进入分组列表。
3. 单击 API 分组列表右上方的 **创建分组** 按钮。
4. 在弹出窗口中，配置 API 分组信息：
   - **分组名称**：必填，用于识别 API 分组。支持英文字母、中文、数字、下划线（_）、连字符（-），32 个字符以内。
   - **描述**：选填，用于描述 API 分组的作用等，64 个字符以内，可为空。
   - **是否启用自定义域名**：网关支持用户进行自定义域名，开启后，您可以输入相应的域名字段。

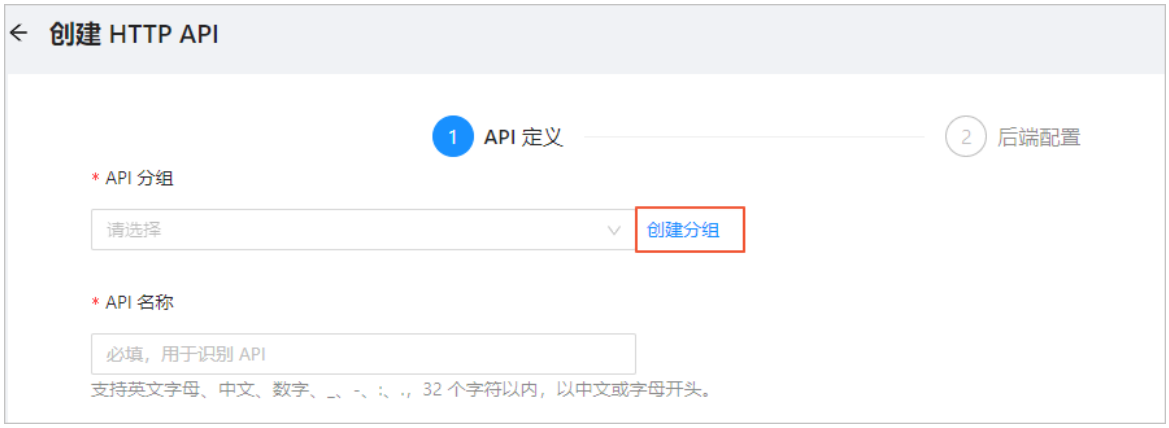
说明

- 域名不能为空，仅支持英文字母、数字、连字符（-）、英文句号（.），至少为5个字符，只能以英文开头，不能以数字、特殊字符结尾；使用连字符（-）时必须为 xx.xx-xx.xx 格式；支持使用 xx.xx.xx 格式，每个英文句号（.）后至少需要两个字符。
- 最多只能添加两个自定义域名。

5. 完成后单击 **确定**，完成 API 分组的创建。

在创建 API 时创建 API 分组

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **API 管理**。
3. 在 API 列表右上方，单击 **创建 API**。
4. 选择您需要的 API 类型后，单击 **创建**，进入 API 创建页面。
5. 在 **API 分组** 栏，单击右侧的 **创建分组** 按钮后，输入分组信息，单击 **确定** 即可。



1.2.2. 管理 API 分组

在 **API 分组** 页面，您可以根据业务需要，查看并管理所有的 API 分组。

查看 API 分组

在 API 分组列表页，您可以查看该分组的部分概览信息，包括分组名称、分组 ID、该分组下的 API 数量、分组描述信息、分组的创建人以及创建时间，如下图所示。

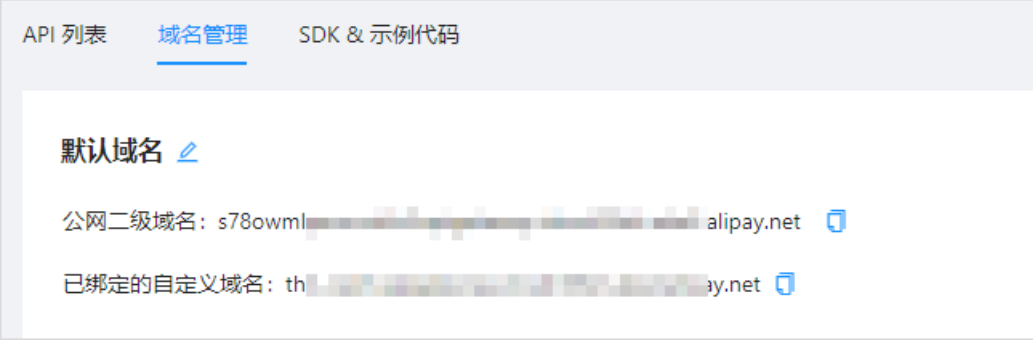
查看 API 分组详情

在分组列表中，您还可以单击分组名称，进入该分组的详情页，查看分组的详细信息，主要包括以下信息：

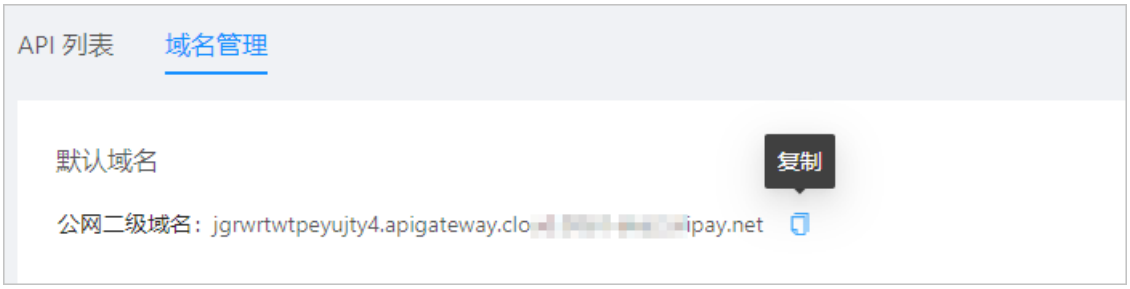
- **基本信息**：展示了分组名称、分组 ID、创建人、创建时间与描述。
- **API 列表**：
  - 列出了该 API 分组下所有的 API 以及 API 的概览信息，比如 API 名称，前端协议类型、OperationType/接口名称、请求路径、方法、状态、后端服务类型等。
  - 在此列表，您还可以对各个 API 进行下线、发布、删除等操作。

- 在列表右上方，单击 **创建 API**，即可直接在此 API 分组下快速创建一个 API。

- **域名管理**：显示了客户端使用 HTTP 协议访问网关时使用的默认二级域名，和已绑定的自定义域名。
  - 公网二级域名由 **分组 ID + 网关的二级域名** 组成。
  - 已绑定的自定义域名由 **用户自定义域名+网关的二级域名** 组成。



- **域名管理**：显示了客户端使用 HTTP 协议访问网关时使用的默认二级域名，由 `分组 ID + 网关的二级域名` 组成。



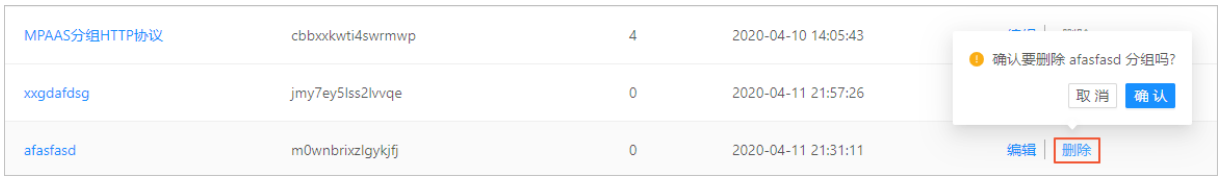
编辑 API 分组

> **说明**
>
> 当该分组 开启分组授权 ，只有获得授权的成员和管理员能够进行编辑。
>
> 您可以通过以下任一方式编辑 API 分组：
>
> - 在 API 分组列表中，找到目标分组，单击右侧 **操作** 列中的 **编辑** 按钮。
> - 在 API 分组的详情页，单击右上角的 **编辑** 按钮。

删除 API 分组

对于无用的 API 分组，您可以将其删除。只需在 API 分组列表中，找到目标分组，单击右侧 **操作** 列中的 **删除** 按钮并确认即可。

> **说明**
>
> - 仅支持删除 API 数量为 0 的 API 分组。
> - 当该分组 开启分组授权 ，只有获得授权的成员和管理员能够进行删除操作。



1.3. 系统集群

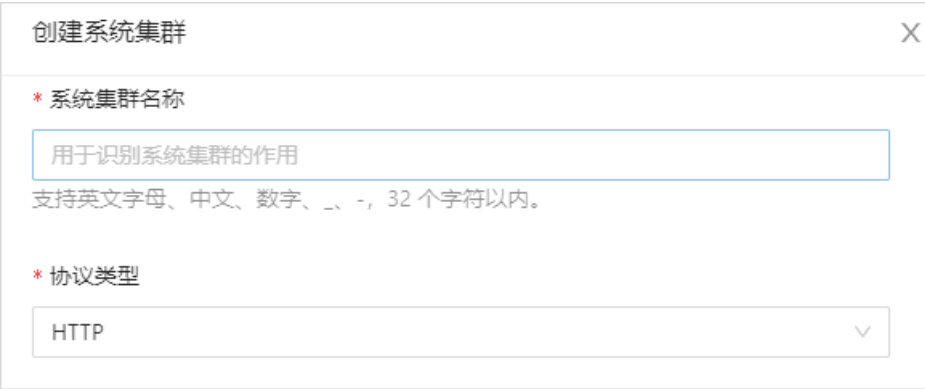
1.3.1. 创建系统集群

系统集群表示 API 网关接收到前端请求后转发到真实业务系统的集群。本文介绍如何在 API 网关控制台快速创建系统集群。

目前 API 网关支持创建 HTTP、SOFARPC、SOFAREST、DUBBO 、 TRIPLE 和 SOFA_TR 六种协议类型的系统集群。

HTTP 协议类型

1. 进入 API 网关控制台页面，左侧导航栏选择 **API 发布 > 系统集群**。
2. 在系统集群列表页面，单击右上方的 **创建系统集群**。
3. 在新弹出窗口中，配置基础信息：



   - **系统集群名称**：必填，用于识别系统集群，支持英文字母、中文、数字、下划线（_）、连字符（-），32 个字符以内。
   - **协议类型**：选择 **HTTP**，超文本传输协议。
4. 选择地址配置方式。

   手动配置：手动配置系统集群的 IP 地址或域名。

   - **IP地址/域名**：指定 IP 地址/域名，格式为 `地址：端口` 。

- IP 地址或域名：
  - IP 地址格式为 `（1~200）.（0~200）.（0~200）.（0~200）` 。
  - 域名可以包含字母、数字或者半角的连字符 （-），总共不超过 200 个字符。
  - 必须以 `http://` 或 `https://` 开头。
- 端口：HTTP 默认端口为 80，SOFARPC/MRPC/SOFAREST 默认端口为 12200。端口号范围为 1-65535。

注册中心：选择已经在网关上创建好的注册中心。

- **注册中心类型**：选择注册中心类型，可选 SOFA（默认）、SOFA（自定义）、NACOS。
- **注册中心名称**：根据所选注册中心类型选择对应的注册中心。

> 说明
>
> - SOFA（默认）表示 SOFA Registry，即蚂蚁默认的注册中心，选择该类型后无需再选择注册中心。
> - SOFA（自定义）表示您可以选择在 全局设置中创建的 SOFA 类型注册中心。

- **负载均衡策略**：指定负载均衡策略，**支持** 随机 或 **轮询**。

5. 完成其他字段配置。
   - **认证方式**：需要选择认证方式，即网关向后端系统集群发送请求时是否加签。可以选择 **无** 或 **密钥**。

     > 说明
     >
     > - 当您创建 SOFARPC 类型的 API，并使用 HTTP 协议将请求转发到后端，即 SOFARPC 转 HTTP 时，此时服务端不支持加签功能。
     > - 网关现已经支持创建自定义密钥，当认证方式选择 **密钥** 时，您可以自定义 AccessKey 和 SecretKey。

   - **描述**：选填，用于描述系统集群的作用等，64 个字符以内，可为空。
6. 单击 **确定**，完成系统集群的创建。

SOFARPC 协议类型

1. 进入 API 网关控制台页面，左侧导航栏选择 **API 发布 > 系统集群**。
2. 在系统集群列表页面，单击右上方的 **创建系统集群**。
3. 在新弹出窗口中，配置基础信息：

- **系统集群名称**：必填，用于识别系统集群，支持英文字母、中文、数字、下划线（_）、连字符（-），32 个字符以内。
- **协议类型**：选择 SOFARPC，蚂蚁金服自研的 RPC 调用框架。

4. 选择地址配置方式。

   手动配置：手动配置系统集群的 IP 地址或域名。

   - **IP地址/域名**：指定 IP 地址/域名，格式为 `地址：端口` 。
     - IP 地址或域名：
       - IP 地址格式为 `（1~200）.（0~200）.（0~200）.（0~200）` 。
       - 域名可以包含字母、数字或者半角的连字符 （-），总共不超过 200 个字符。
       - 必须以 `http://` 或 `https://` 开头。
     - 端口：HTTP 默认端口为 80，SOFARPC/MRPC/SOFAREST 默认端口为 12200。端口号范围为 1-65535。

   SOFA 注册中心：从当前租户的 SOFA 注册中心获取地址列表。

- **注册中心类型**：选择注册中心类型，可选 SOFA（默认）、SOFA（自定义）。

> 说明
>
> - SOFA（默认）表示 SOFA Registry，即蚂蚁默认的注册中心，选择该类型后无需再选择注册中心。
> - SOFA（自定义）表示您可以选择在 全局设置中创建的 SOFA 类型注册中心。

- **负载均衡策略**：指定负载均衡策略，支持 **随机** 或 **轮询**。

5. 完成其他字段配置。

   描述：选填，用于描述系统集群的作用等，64 个字符以内，可为空。

6. 单击 **确定**，完成系统集群的创建。

SOFAREST 协议类型

1. 进入 API 网关控制台页面，左侧导航栏选择 **API 发布 > 系统集群**。
2. 在系统集群列表页面，单击右上方的 **创建系统集群** 按钮。
3. 在新弹出窗口中，配置基础信息：

- **系统集群名称**：必填，用于识别系统集群，支持英文字母、中文、数字、下划线（_）、连字符（-），32 个字符以内。
- **协议类型**：选择 SOFAREST，基于 JAX-RS 标准的 SOA 解决方案。

4. 选择地址配置方式。

   手动配置：手动配置系统集群的 IP 地址或域名。

   - **IP地址/域名**：指定 IP 地址/域名，格式为 `地址：端口` 。
     - IP 地址或域名：
       - IP 地址格式为 `（1~200）.（0~200）.（0~200）.（0~200）` 。
       - 域名可以包含字母、数字或者半角的连字符（-），总共不超过 200 个字符。
       - 必须以 `http://` 或 `https://` 开头。

- 端口：HTTP 默认端口为 80，SOFARPC/MRPC/SOFAREST 默认端口为 12200。端口号范围为 1-65535。

SOFA 注册中心：从当前租户的 SOFA 注册中心获取地址列表。

- **注册中心类型**：选择注册中心类型，可选 SOFA（默认）、SOFA（自定义）。

> 说明
>
> - SOFA（默认）表示 SOFA Registry，即蚂蚁默认的注册中心，选择该类型后无需再选择注册中心。
> - SOFA（自定义）表示您可以选择在 全局设置中创建的 SOFA 类型注册中心。

- **负载均衡策略**：指定负载均衡策略，支持 **随机** 或 **轮询**。

5. 完成其他字段配置。
   - **认证方式**：需要选择认证方式，即网关向后端系统集群发送请求时是否加签。可以选择 **无** 或 **密钥**。

> 说明
>
> - 开启系统集群认证后，需要将客户端 SDK 的验签开关打开。
> - 关闭系统集群认证后，需要将客户端 SDK 的验签开关关闭，否则会导致验签失败。

   - **描述**：选填，用于描述系统集群的作用等，64 个字符以内，可为空。
6. 单击 **确定**，完成系统集群的创建。

DUBBO 协议类型

1. 进入 API 网关控制台页面，左侧导航栏选择 **API 发布 > 系统集群**。
2. 在系统集群列表页面，单击右上方的 **创建系统集群**。
3. 在新弹出窗口中，配置基础信息：

- **系统集群名称**：必填，用于识别系统集群，支持英文字母、中文、数字、下划线（_）、连字符（-），32 个字符以内。
- **协议类型**：选择 DUBBO，提供高性能 RPC 服务以及 SOA 服务治理方案的服务框架。

4. 选择地址配置方式。

   手动配置：手动配置系统集群的 IP 地址或域名。

   - **IP地址/域名**：指定 IP 地址/域名，格式为 `地址：端口` 。
     - IP 地址或域名：
       - IP 地址格式为 `（1~200）.（0~200）.（0~200）.（0~200）` 。
       - 域名可以包含字母、数字或者半角的连字符 （-），总共不超过 200 个字符。
       - 必须以 `http://` 或 `https://` 开头。
     - 端口：HTTP 默认端口为 80。

   注册中心：从当前租户已创建好的注册中心获取地址列表。

- **注册中心类型**：支持 ZK 和 NACOS 类型。
- **注册中心名称**：选择已经创建好的注册中心名称。
- **负载均衡策略**：指定负载均衡策略，**支持** 随机 或 **轮询**。

5. 完成其他字段配置。

   描述：选填，用于描述系统集群的作用等，64 个字符以内，可为空。

6. 单击 **确定**，完成系统集群的创建。

1.3.2. 管理系统集群

在 **系统集群** 页面，您可以根据业务需要，查看并管理已创建的系统集群。

查看系统集群

在系统集群列表页，您可以查看该系统集群的概览信息，包括系统集群名称 、地址配置方式、协议类型、地址/域名 、绑定 API 数量、认证方式以及创建人。

查看系统集群详情

在系统集群列表中，您还可以单击系统集群名称，进入该系统集群的详情页，查看详细信息，主要包括以下信息：

- **基本信息**：展示了系统集群名称、集群地址获取方式、地址/域名、负载均衡策略、创建人、创建时间与描述。
- **绑定的API**：
  - 列出了绑定了该系统集群作为后端服务类型的所有 API 及其基本信息，例如 API 名称，前端协议类型、OperationType/ 接口名称、请求路径、方法、状态、创建时间等。
  - 在此列表，您还可以单击任一 API 名称，进入该 API 的详情页查看 API 的具体信息及监控数据等。
- **认证配置**：表示网关向后端系统集群发请求时加签配置信息。可以在此处获取系统集群的 Access Key 与 Secret Key。

编辑系统集群

> **说明**
>
> 如果当前系统集群已绑定 API，则无法被编辑。需前往 API 编辑页面，将对应 API 的系统集群替换为其他系统集群后，方可编辑当前系统集群。

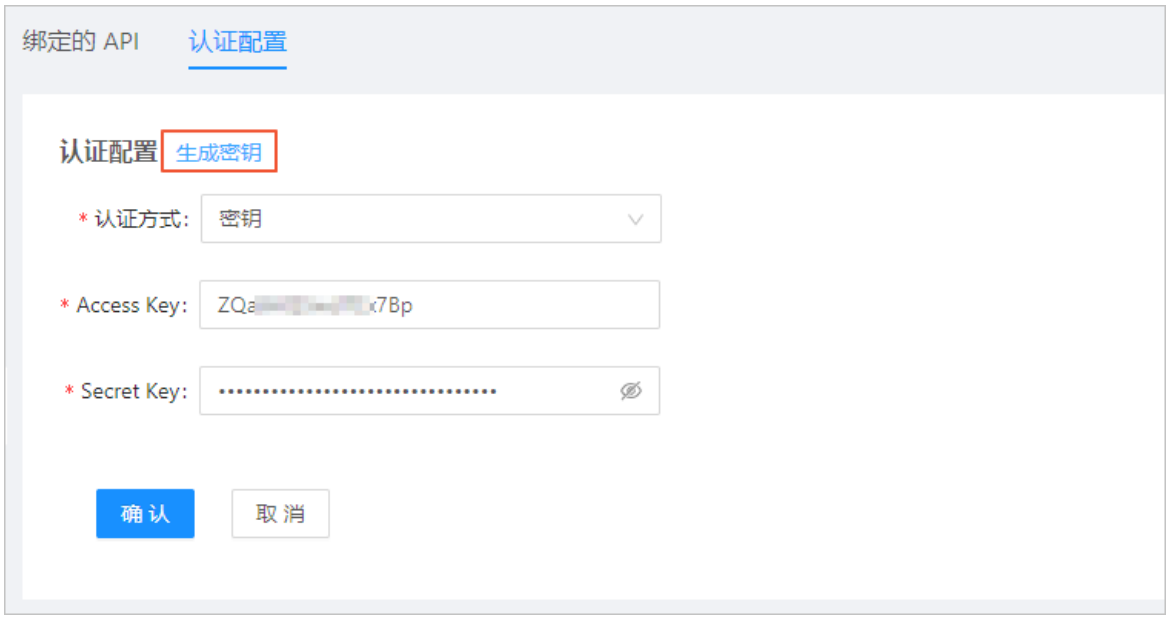
您可以通过以下任一方式编辑系统集群：

- 在系统集群列表中，找到目标集群，单击右侧 **操作** 列中的 **编辑** 按钮。
- 在系统集群的详情页，单击右上角的 **编辑** 按钮。

修改认证配置

1. 在系统集群详情页，切换至 **认证配置** 标签。
2. 单击 **认证配置** 右侧图标，即可修改认证方式。

3. 对于密钥认证方式，您还可以单击 **生成密钥**，获取一对新密钥。



删除系统集群

对于无用的系统集群，您可以将其删除。只需在系统集群列表中，找到目标集群，单击右侧 **操作** 列中的 **删除** 按钮并确认即可。

> **说明**
>
> 如果当前系统集群已绑定 API，则无法被删除。需前往 API 编辑页面，将对应 API 的系统集群替换为其他系统集群后，方可删除当前系统集群。

1.4. 路由规则

1.4.1. 创建路由规则

在网关接收到请求后，会通过路由规则将请求路由到后端系统，您可在此创建相关的路由规则。

操作步骤

1. 登录 API 网关控制台页面。
2. 在左侧导航栏中单击 **路由规则**，进入路由规则列表。

3. 单击路由规则列表右上方的 **创建路由规则**。
4. 在弹出的 **创建路由规则** 窗口中，选择或输入相关信息：

- **规则名称**：必填，用于识别路由规则的作用。
- **后端协议类型**：必选，选择该路由规则的后端协议类型，可选择 HTTP 或 SOFARPC。
- **路由方式**：必选，选择该路由规则的路由方式。
- **规则详情**：必填，所创建的路由规则的规则详情，您可单击底部的 **+ 添加** 来设置多条规则。
  - 若 **路由方式** 选择了 **根据 Header 选择路由**，则需要在规则详情中输入 **参数名**、**匹配方式**、**参数值** 以及 **业务集群**。其中，业务集群下拉菜单中的内容为已创建的系统集群。

- 若 **路由方式** 选择了 **根据权重选择路由**，则需要在规则详情中输入 **流量比例** 和 **业务集群**。其中，业务集群下拉菜单中的内容为已创建的系统集群。

> **说明**
>
> 所有创建的业务集群的流量比例之和必须为 100。

- 若 **路由方式** 选择了 **根据 LDC 选择路由**，则需要在规则详情中选择或输入 **参数位置**、**参数名称**、**截取规则** 以及 **业务集群**。其中，业务集群下拉菜单中的内容为已创建的系统集群。

5. 单击 **确定**，即可完成路由规则的创建。

已创建的路由规则可用于在创建 API 时，选择 API 的路由规则。

1.4.2. 管理路由规则

在 **路由规则** 页面，您可以根据业务需要，查看并管理已创建的路由规则。

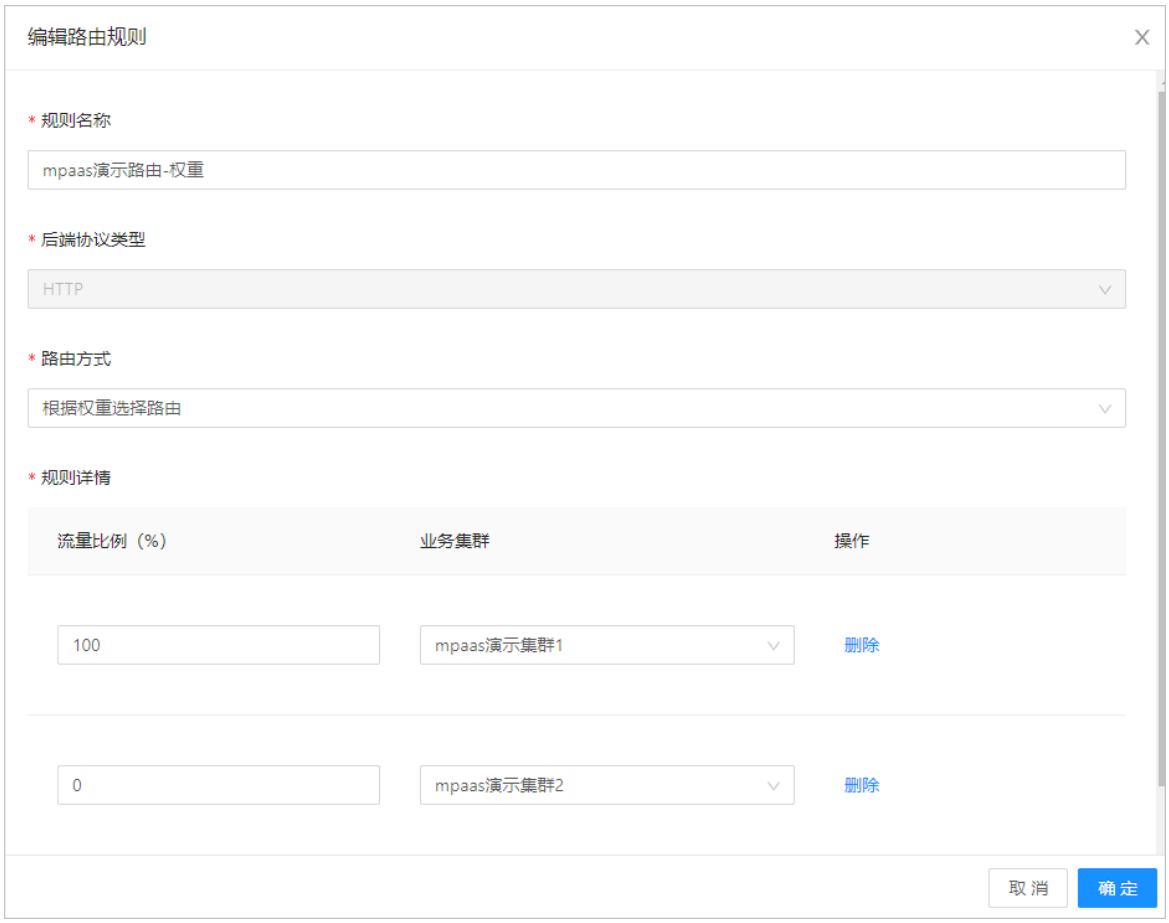
编辑路由规则

> **说明**
>
> 如果当前路由规则已绑定 API，则无法被编辑。需前往 API 编辑页面，将对应 API 的路由规则替换为其他路由规则后，方可编辑当前路由规则。

1. 在 **路由规则** 页面，通过以下任一方式打开路由规则编辑界面：
   - 在路由规则列表页，单击路由规则右侧 **操作** 列中的 **编辑** 即可打开路由规则编辑界面。
   - 单击路由规则名进入路由规则详情页后，单击页面右上方的 **编辑** 即可打开路由规则编辑界面。
2. 打开的 **编辑路由规则** 界面与创建路由规则界面内容一致，您可修改 **规则名称**、**路由方式** 以及 **规则详情**。

> **说明**
>
> 后端协议类型一旦创建则无法更改。

3. 单击 **确定**，即可完成路由规则的修改。

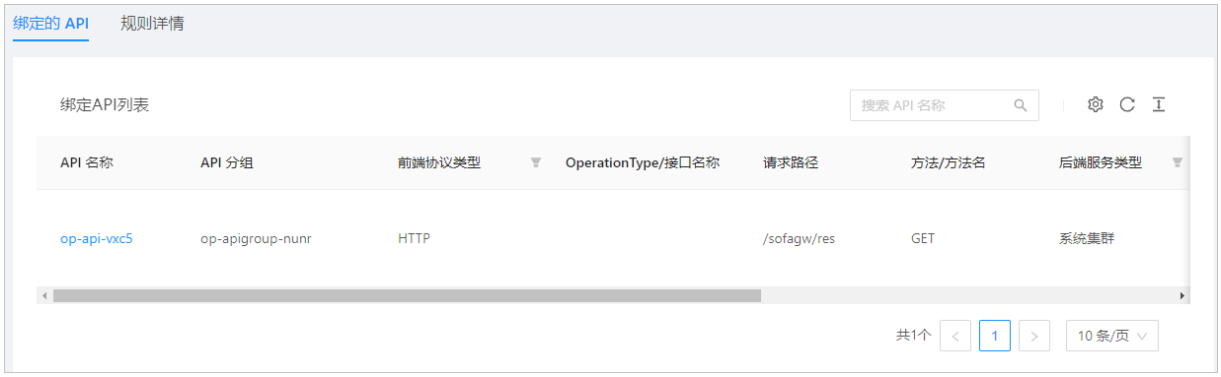


查看路由规则详情

在路由规则列表页，单击路由规则名可进入路由规则详情页。

绑定的 API

在路由规则详情页的 **绑定的 API** 标签页中，可查看当前路由规则已绑定的 API。



在绑定的 API 列表中，您可以：

- **查看 API 详情**：单击 API 名称，即可跳转至该 API 的详情页，您可查看该 API 的 **授权对象**、**配置信息**、**版本管理** 等详细信息。
- **筛选前端协议类型**：
  i. 单击 **前端协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的前端协议并单击 **确定** 即可进行筛选。
- **筛选后端服务类型**：
  i. 单击 **后端服务类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的后端服务并单击 **确定** 即可进行筛选。
- **筛选后端协议类型**：
  i. 单击 **后端协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的后端协议并单击 **确定** 即可进行筛选。
- **筛选状态**：
  i. 单击 **状态** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择 **已发布** 或 **未发布**，并单击 **确定** 即可进行筛选。
- **搜索 API**：在 API 列表右上方的搜索框中，可通过 API 名称搜索已绑定的 API。
- **更改展示设置**：
  - **更改列设置**：在 API 列表右上方的功能区域，单击 **列设置** 按钮。在打开的列展示框中，您可勾选或取消 API 列表的展示项目，也可将某一项固定在左/右侧，或拖动展示项进行排序。
  - **调整列密度**：在 API 列表右上方的功能区域，单击 **密度** 按钮。在打开的密度选择框中，可调整 API 列表的展示密度。

规则详情

此处的规则详情即为 创建路由规则 时所填写的规则详情。

删除路由规则

在路由规则列表页，单击路由规则右侧 **操作** 列中的 **删除**，在确认框中单击 **确定** 即可删除当前路由规则。

> **说明**
>
> 如果当前路由规则已绑定 API，则无法被删除。需前往 API 编辑页面，将对应 API 的路由规则替换为其他路由规则后，方可删除当前路由规则。

1.5. 数据模型

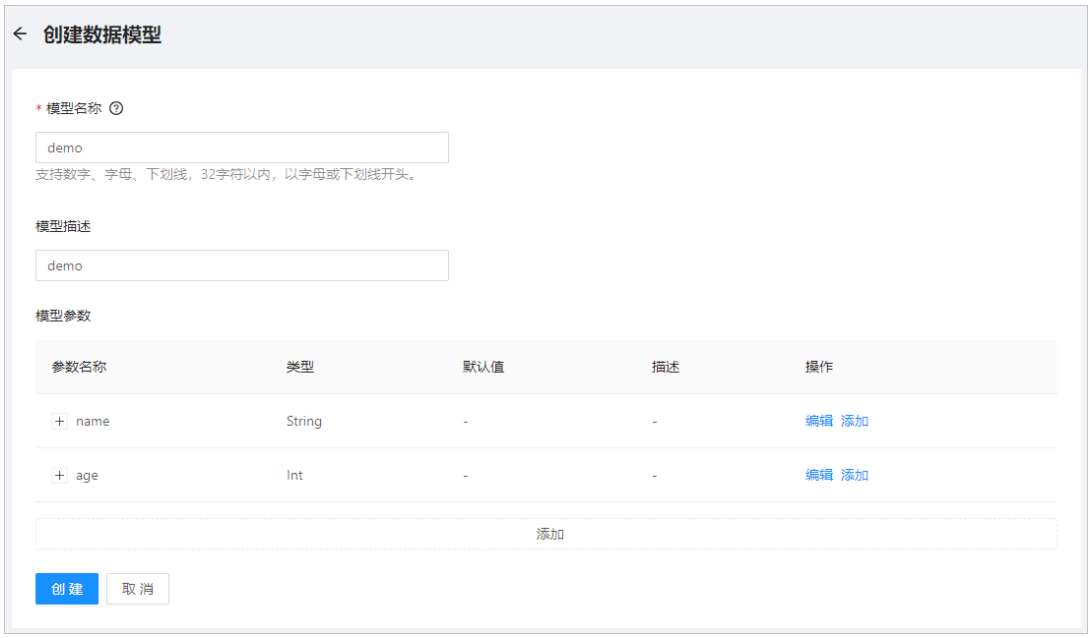
1.5.1. 创建数据模型

API 网关支持将 API 服务的请求与响应参数定义成数据模型。继而在创建 API 时，可以直接引用数据模型的配置参数，提高参数设置效率。本文介绍如何快速创建 API 参数数据模型。

操作步骤

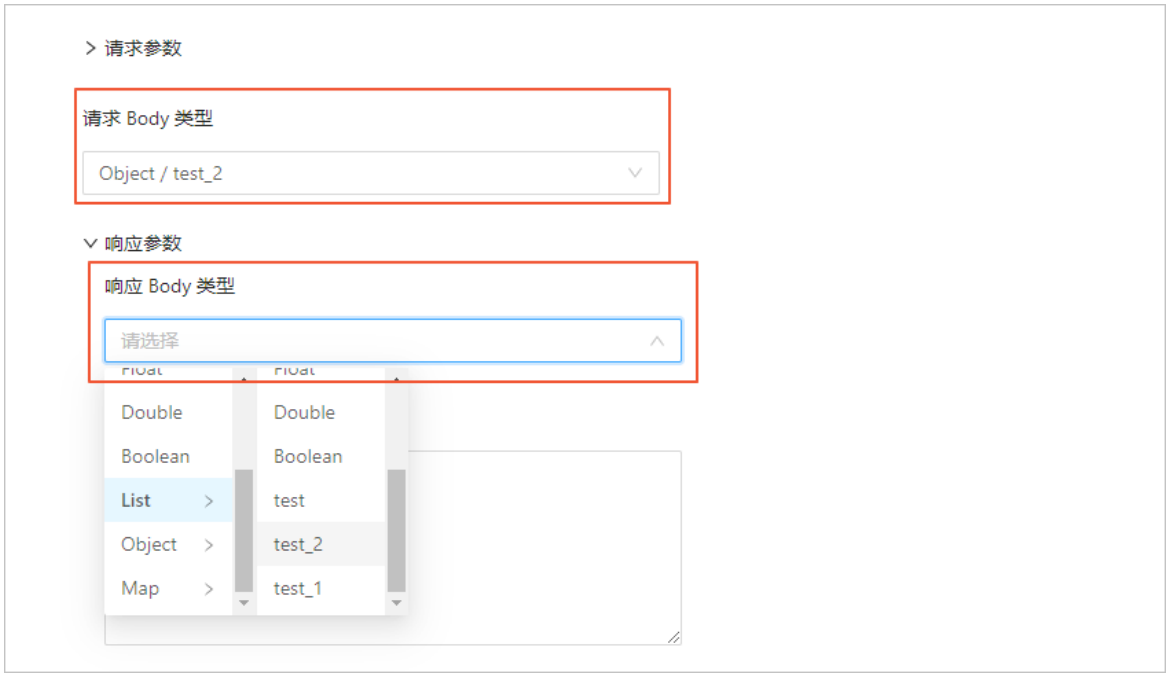
1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > 数据模型**。
2. 在数据模型列表页，单击列表右上方的 **创建数据模型** 按钮。
3. 在新打开的 **创建数据模型** 页面中，配置如下信息：
   - **模型名称**：必填，用于识别数据模型，需保证唯一性。支持数字、字母、下划线，32 个字符以内，以字母或下划线开头。
   - **模型描述**：选填，用于描述该数据模型的业务意义或使用范围等，64 个字符以内，可为空。
   - **模型参数**：必填，根据业务需要配置相应的模型参数。
     a. 单击 **添加** 按钮，添加需要的参数信息：
     - **参数名称**：必填，参数的唯一标识。
     - **类型**：必填，可选的参数类型包括 String、Int、Long、Double、Float、Boolean、List、Object、Map。
     - **默认值**：选填，在 API 请求的参数值为空时使用的默认值。默认值类型必须和参数类型一致。
     - **描述**：选填，用于描述参数的业务意义等，可为空。

b. 添加完成后，单击 **保存**，才可继续添加其他参数。



4. 确认信息无误后，单击 **创建**。

   数据模型创建成功后，后续在创建 API 配置请求/响应参数时，您可以直接选择对应的数据模型进行绑定，如下图所示：



1.5.2. 导入数据模型

为了方便用户快速生成数据模型，API 网关支持通过 Proto 文件和 Swagger 文件来快速导入数据模型。

操作步骤

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布 > 数据模型**，进入数据模型页面。

3. 在页面右上角单击 **导入数据模型**，在弹出的窗口中配置数据模型信息。

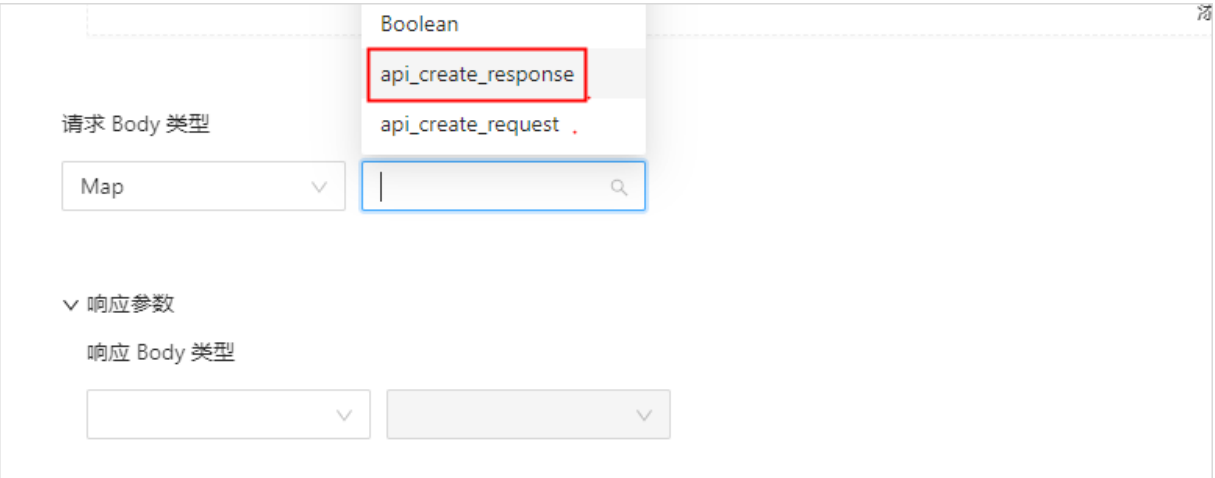
- **选择导入类型：** API 网关提供 Swagger 和 ProtoBuf 两种数据模型的导入，您可以根据实际业务需求进行选择。
  - Swagger 模型：Swagger 是一个规范和完整的框架，用于生成、描述、调用和可视化 RESTful 风格的 Web 服务。
  - ProtoBuf 模型：ProtoBuf 是一个开源的结构化信息传递工具，用于数据存储、传输协议格式等场景。
- **导入方式：**
  - URL：输入swagger配置的 URL地址，将远程的数据模型解析后，上传至 API 网关平台。
  - FILE：选择好导入类型后，单击 **上传文件**，将swagger的 **2.0** 版本 或 **3.0** 版本的 **json** 或**yaml** 文件，上传至 API 网关平台。

- **性能优化：**

> **说明**
>
> 导入类型为 ProtoBuf，Swagger 时，都不在需要输入模型名称。
>
> API网关会从swagger 2.0文件中的Model，Swagger文件 3.0 的Schema进行解析参数名称，参数结构，参数类型并转换后导入数据模型，可直接在API中自动生成示例入参的JSON，数据模型也可直接在流程编排中进行参数配置，参数映射等。

- 完成后单击 **确定**，完成数据模型的导入。

数据模型导入成功后，后续在创建 API 配置请求/响应参数时，您可以直接选择对应的数据模型进行绑定，如下图所示：



完成数据模型的导入后，您可以按需 生成 SDK并下载到本地进行项目开发。

其他操作

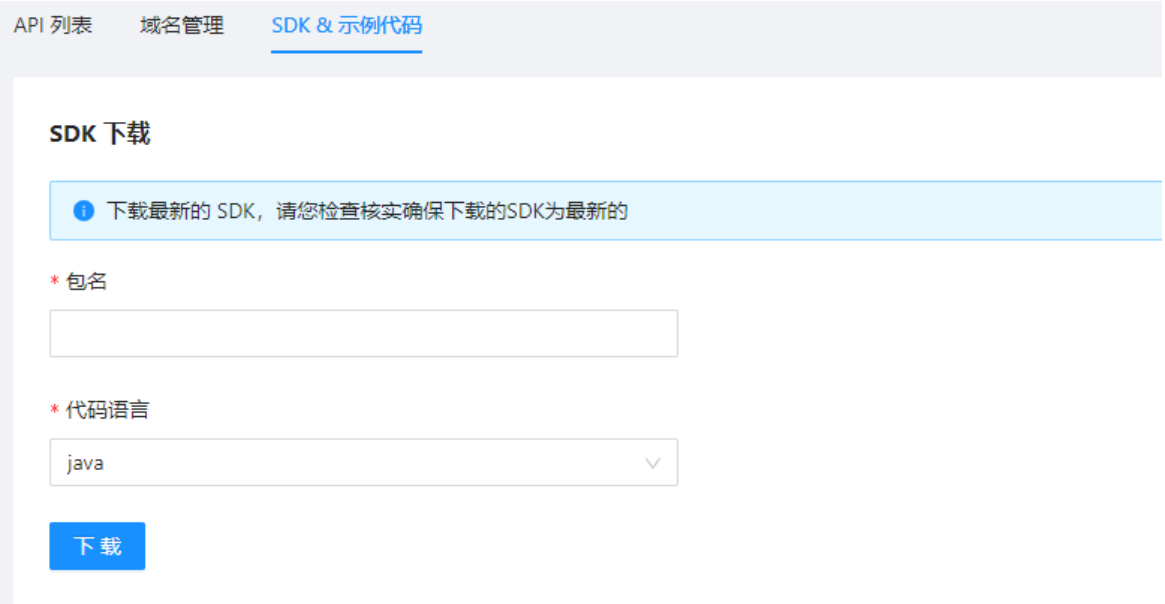
生成 SDK

为了方便您调用 API 接口，同时为了避免重复编写接口代码，API 网关提供下载 SDK 的操作，简化编写代码的同时，满足您对外输出 SDK 的需求。

> **说明**
>
> 目前 API 网关支持从 **分组** 和 **应用** 两个维度下载 SDK，以下操作骤以 **分组** 角度为例。

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **API 分组**，进入 API 分组页面。
3. 在分组列表右侧单击 **下载 SDK**。

4. 在分组详情页配置下载 SDK 的信息。



- **包名：** 必填，用于生成类的包名。
- **代码语言：** 选择下载 SDK 的代码语言，目前仅支持 java。

5. 完成后单击 **下载**，完成 SDK 的生成。

> **说明**
>
> 下载后的 SDK 形式为 zip文件，您可以将 zip 包里面的 Java 类文件导入对应项目中用于 SDK 的调用。

注意事项

网关 SDK 生成不支持的数据类型

- 容器类型不能多层嵌套。
- List 或 Map 必须有泛型信息。
- List 或 Map 的泛型信息不能是 Array 类型。
- 不支持单字节（字节数据 byte[] 是支持的。）
- 不支持对象数组，请用 List 代替。
- 属性名不能是 `data` 和 `description`，会与 iOS 的属性冲突。
- Map 类型的 key 必须是 String 类型。
- Map 类型的 value 不能是 Object 类型。
- 类型不能是抽象类。
- 类型不能是接口类。

错误的写法为：

```
public class Req {
    private Map<String,List<Person>> map; //容器类型不能多层嵌套。
    private List<Map<Person>> list; //容器类型不能多层嵌套。
    private List list1; //List或Map必须有泛型信息。
    private Map map1; //List或Map必须有泛型信息。
    private List<Person[]> listArray; //List或Map的泛型信息不能是Array类型。
    private byte b; //不能为单字节
    private Person[] personArray; //不支持对象数组，请用List代替
    private String description; //属性名不能description
}
```

网关 SDK 生成支持的数据类型

```
boolean, double, float, int, long, short
java.lang.Boolean
java.lang.Double
java.lang.Float
java.lang.Integer
java.lang.Long
java.lang.Short
java.lang.String
java.util.List,但：必须使用类型参数；不能使用其具体子类 以下简称List
java.util.Map,但：必须使用类型参数；不能使用其具体子类；key类型必须是String 以下简称Map
Enum
byte[]
```

正确的写法为：

```
public class Req {
    private String s = "ss";
    private int i;
    private double d;
    private Long l;
    private long l1;
    private boolean b;
    private List<String> stringList;
    private List<Person> personList;
    private Map<String,Person> map;
    private byte[] bytes;
    private EnumType type;
}

public class Person {
    private String name;
    private int age;
}
```

1.5.3. 管理数据模型

在数据模型页面，您可以根据业务需要，查看并管理已创建的数据模型。

查看数据模型

在数据模型列表页，您可以查看所有数据模型及其绑定的 API 数量与创建时间。

- **绑定的 API数量**：
  - 数据模型被api的请求数据模型和响应数据模型中同时被使用，则数量按照1条计算

查看数据模型详情

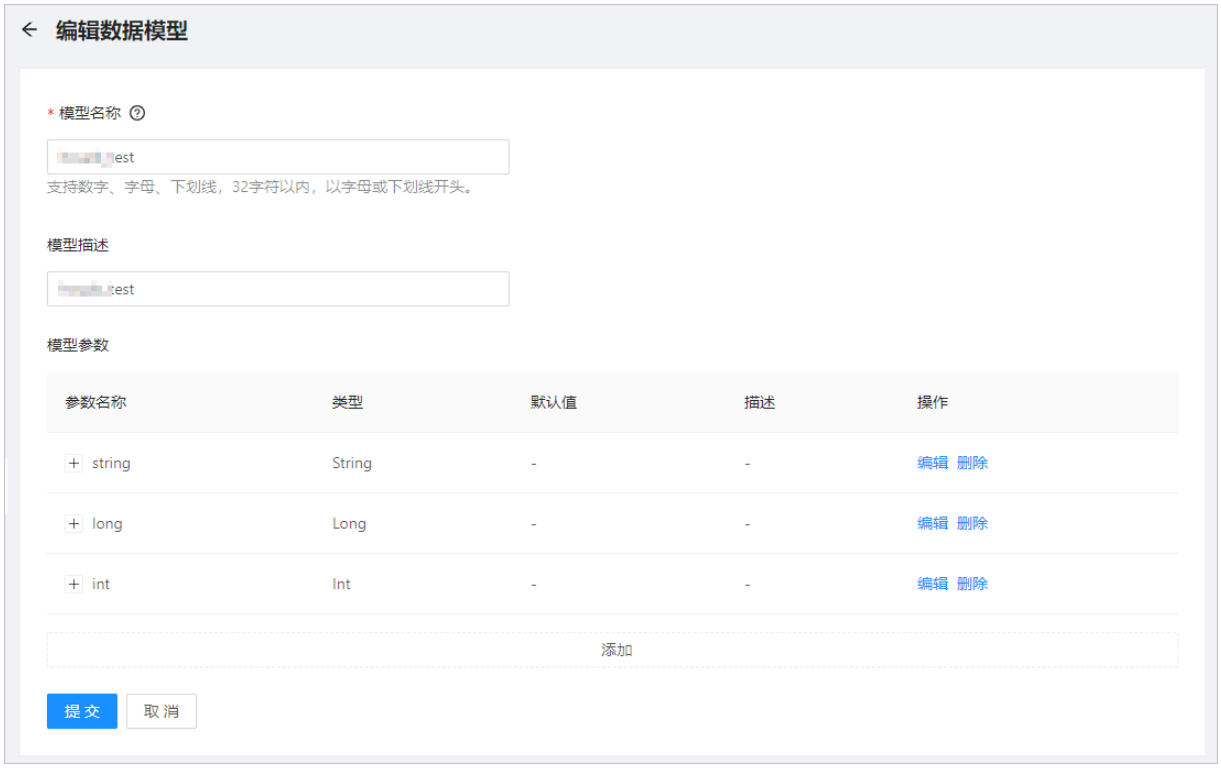
在数据模型列表中，您还可以单击任一数据模型名称，进入该模型的详情页，查看详细信息，主要包括以下信息：

- **基本信息**：展示了数据模型名称、创建人、创建时间与描述。
- **绑定的 API**：
  - 列出了绑定了该数据模型的所有 API 及其基本信息，包括 API 名称、API 分组、授权应用数量、前端协议类型、请求路径等。
  - 在此列表，您还可以单击任一 API 名称，进入该 API 的详情页查看 API 的具体信息及监控数据等。
- **配置详情**：展示了该数据模型配置的参数列表。

编辑数据模型

您可以通过以下任一方式编辑数据模型。

- 在数据模型列表中，找到目标模型，单击右侧 **操作** 列中的 **编辑** 按钮。
- 在数据模型的详情页，单击右上角的 **编辑** 按钮。



> **说明**
>
> 如果选择数据模型的子类模型中存在循环依赖的数据模型，则无法编辑

删除数据模型

> **说明**
>
> 如果当前数据模型已被引用或已绑定了 API，则无法被删除。

您可以通过以下任一方式删除数据模型：

- 在数据模型列表页，单击相应数据模型右侧 **删除** 按钮，在确认框中单击 **确定** 即可。
- 在数据模型的详情页，单击右上角的 **删除** 按钮，然后单击 **确定** 即可。

1.5.4. SDK 生成

为了方便您调用 API 接口，同时为了避免重复编写接口代码，API 网关提供下载 SDK 的操作，简化编写代码的同时，满足您对外输出 SDK 的需求。

前提条件

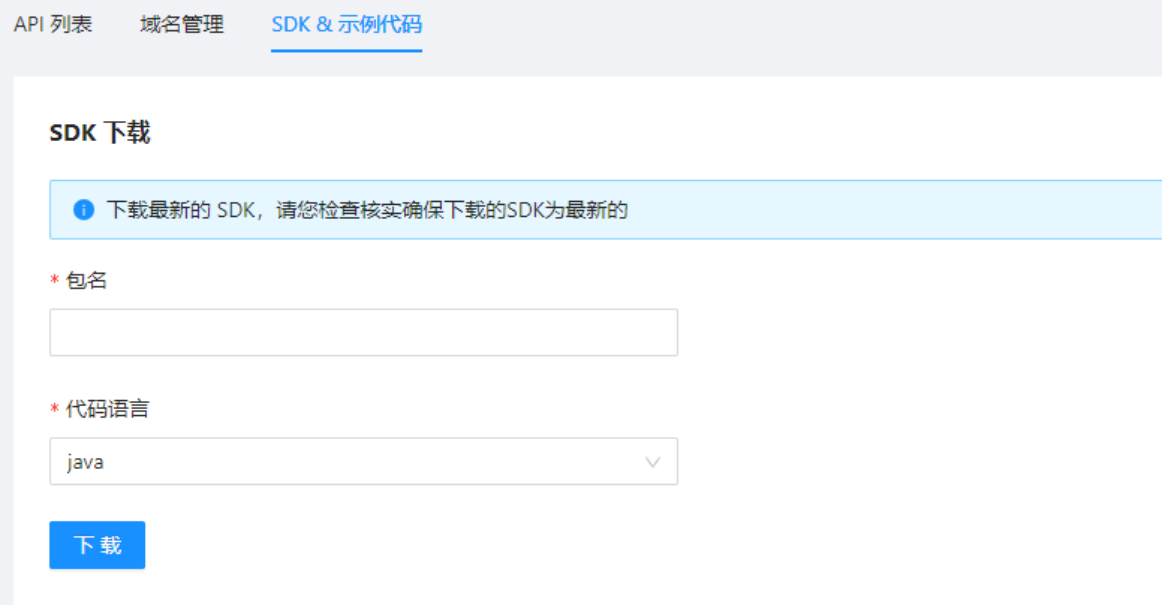
- 已创建数据模型
- 需要生成 SDK 的 API 已绑定数据模型

操作步骤

> **说明**
>
> 目前 API 网关支持从 **分组** 和 **应用** 两个维度下载 SDK，以下操作骤以 **分组** 角度为例。
>
> 注意: 使用SDK下载api必须绑定数据模型否则会下载失败

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **API 分组**，进入 API 分组页面。
3. 在分组列表右侧单击 **下载 SDK**。

4. 在分组详情页配置下载 SDK 的信息。



   - **包名**：必填，用于生成类的包名。
   - **代码语言**：选择下载 SDK 的代码语言，目前仅支持 java。
5. 完成后单击 **下载**，完成 SDK 的生成。

> **说明**
>
> 下载后的 SDK 形式为 zip文件，您可以将 zip 包里面的 Java 类文件导入对应项目中用于 SDK 的调用。

注意事项

网关 SDK 生成不支持的数据类型

- 容器类型不能多层嵌套。
- List 或 Map 必须有泛型信息。

- List 或 Map 的泛型信息不能是 Array 类型。
- 不支持单字节（字节数据 byte[] 是支持的。）
- 不支持对象数组，请用 List 代替。
- 属性名不能是 `data` 和 `description`，会与 iOS 的属性冲突。
- Map 类型的 key 必须是 String 类型。
- Map 类型的 value 不能是 Object 类型。
- 类型不能是抽象类。
- 类型不能是接口类。

错误的写法为：

```
public class Req {
    private Map<String,List<Person>> map; //容器类型不能多层嵌套。
    private List<Map<Person>> list; //容器类型不能多层嵌套。
    private List list1; //List或Map必须有泛型信息。
    private Map map1; //List或Map必须有泛型信息。
    private List<Person[]> listArray; //List或Map的泛型信息不能是Array类型。
    private byte b; //不能为单字节
    private Person[] personArray; //不支持对象数组，请用List代替
    private String description; //属性名不能description
}
```

网关 SDK 生成支持的数据类型

```
boolean, double, float, int, long, short
java.lang.Boolean
java.lang.Double
java.lang.Float
java.lang.Integer
java.lang.Long
java.lang.Short
java.lang.String
java.util.List,但：必须使用类型参数；不能使用其具体子类 以下简称List
java.util.Map,但：必须使用类型参数；不能使用其具体子类；key类型必须是String 以下简称Map
Enum
byte[]
```

正确的写法为：

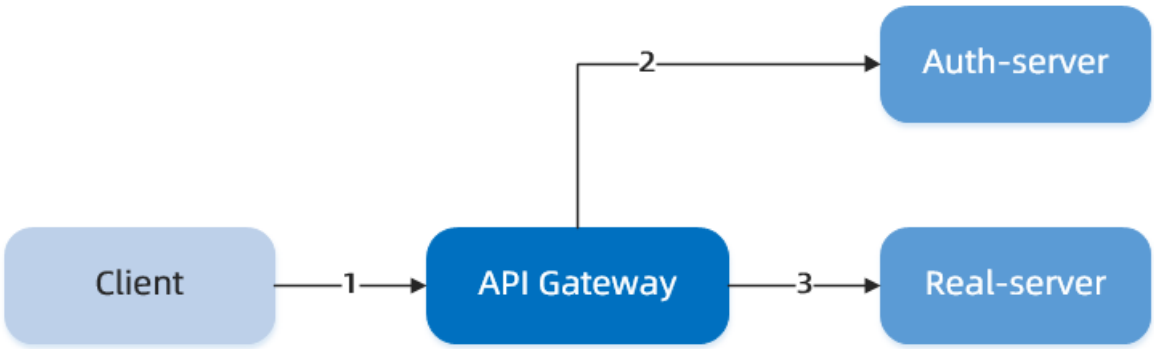
```
public class Req {
    private String s = "ss";
    private int i;
    private double d;
    private Long l;
    private long l1;
    private boolean b;
    private List<String> stringList;
    private List<Person> personList;
    private Map<String,Person> map;
    private byte[] bytes;
    private EnumType type;
}

public class Person {
    private String name;
    private int age;
}
```

1.6. 外部授权 API

1.6.1. 使用外部授权 API

API 网关支持外部授权功能。通过该功能，用户可以在网关将请求转发给 real-server 之前，插入一个自定义的远程接口。网关会先将请求转发给这个自定义接口，该接口继而选择是否允许这个请求正常转发到 real-server。

如下图所示，步骤 2 即为调用自定义的远程接口。



本文将引导您使用外部授权 API。完整的操作步骤如下：

1. **编写外部授权 API：** 在本地开发定义一个外部授权接口服务。
2. **创建外部授权 API：** 在 API 网关控制台，配置对应的外部授权 API。
3. **绑定 API：** 将创建好的外部授权 API 绑定一个 API，使其生效。

编写外部授权 API

您需要在本地开发一个外部授权接口。当 API 需要验证授权关系时，会调用该外部授权接口进行授权校验。外部授权接口不限制协议类型，但会限制请求体和响应体。请求和响应均为 JSON 字符串，格式如下：

- **请求体**

```
{
"context":{
"key":"value"
}
}
```

字段说明：Request 中只有一个 context 字段，格式为 kv。这些参数来自于 client 的 request，需要在 API 网关控制台平台进行配置。

- **响应体**

```
{
"success":true/false,
"principal":{
"key":"value"
},
"failResponseHeader":{
"key":"value"
},
"failResponseBody": jsonarray/jsonobject
"failResponseStatus": ${httpcode}
}
```

字段说明：

- `success`：是否允许该请求转发到 real-server。
- `principal`：如果允许转发到 real-server，可以将一些信息传递给 real-server，kv 格式。
  - 如果 real-server 是 HTTP 接口，则 principal 会放到 header 中。
  - 如果 real-server 是 SOFARPC 接口，则 principal 会放到 baggage 中。
- `failResponseHeader`：如果不允许转发到 real-server，可以设置响应头返回给 client。
- `failResponseBody`：如果不允许转发到 real-server，可以设置响应 body 给 client。
- `failResponseStatus`：如果不允许转发到 real-server，可以设置响应 HTTP 状态码。

对应外部授权 API 的定义标准如下：

- **AuthRequest**

```
publicclassAuthRequest{
privateMap<String,String> context;
}
```

- **AuthResponse**

```
publicclassAuthResponse{
privateboolean              success;
privateMap<String,String>  principal;
privateMap<String,String> failResponseHeader;
private Object               failResponseBody;
privateint                  failResponseStatus;
}
```

- **接口示例**

```
@PostMapping("/testAuth")
publicAuthResponse testAuth(@RequestBodyAuthRequest authRequest){
String sid = authRequest.getContext().get("sid");
AuthResponse response =newAuthResponse();

if(sid !=null){
Map<String,String> principal =newHashMap<>();
      principal.put("uid", sid +"_uid");
      response.setSuccess(true);
      response.setPrincipal(principal);
}else{
      response.setSuccess(false);

Map<String,String> failResponseHeader =newHashMap<>();

String queryKey = request.getContext().get("queryKey");
if(queryKey.equalsIgnoreCase("q")){
          failResponseHeader.put("header-to-client","query");
}else{
          failResponseHeader.put("header-to-client","no-query");
}

      response.setFailResponseHeader(failResponseHeader);

Map<String,String> failResponseBody =newHashMap<>();

String bodyKey = request.getContext().get("bodyKey");
if(bodyKey.equalsIgnoreCase("b")){
          failResponseBody.put("body-to-client","b");
}else{
          failResponseBody.put("body-to-client","no-b");
}

      response.setFailResponseBody(failResponseBody);

      response.setFailResponseStatus(401);

}

return authResponse;
}
```

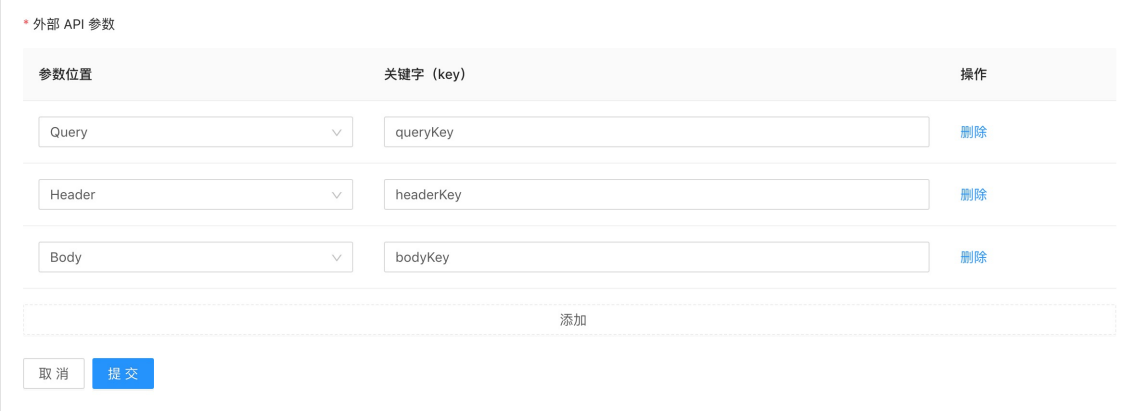
创建外部授权 API

本地编写完外部授权接口后，您需要前往 API 网关的控制台上配置一个对应的外部授权 API。操作步骤如下：

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > 外部授权 API**。
2. 在外部授权 API 列表页，点击列表右上方的 **创建外部授权 API**。
3. 在新打开的页面中，选择或输入外部授权 API 相关信息：
   - **外部授权 API 名称**：必填，用于识别该 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内，以中文或字母开头。
   - **后端服务类型**：目前仅支持 **系统集群**，即表示网关接收到前端请求后转发到真实业务系统的集群。
   - **协议类型**：支持多种协议。根据协议的不同，需要配置的信息也有所不同。
     - HTTP 协议：
       - **请求路径**：必填，API 请求所指向的资源 URL。以斜杠（/）开头，支持字母、数字、下划线（_）、连接符（-），200 字符以内。格式示例：`/home/{id}`。
       - **方法**：外部授权 API 必须使用 POST 方法，即将数据发送到服务器进行处理。
     - SOFARPC 协议：
       - **接口名称**：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
       - **服务标识**：选填，支持指定服务的 uniqueId。
       - **方法名**：支持英文字母和数字，以小写字母作为开头，格式示例：`getUserID`。
     - MRPC 协议：

       OperationType：必填，针对 mPaaS 移动应用设置的 API 服务标识，用于定位要访问的资源。
     - SOFAREST 协议：
       - **方法**：必选，外部授权 API 必须使用 POST 方法。
       - **接口名称**：必填，支持英文字母、数字、英文句号（.）、冒号（:）、@，以小写字母作为开头，格式示例：`com.alipay.testapp.xxxx.ConfigFacade:1.0@DEFAULT`。
       - **服务标识**：选填，支持指定服务的 uniqueId。
       - **请求路径**：必填，API 请求所指向的资源 URL。以斜杠（/）开头，支持字母、数字、下划线（_）、连接符（-），200 字符以内。格式示例：`/home/{id}`。
   - **超时时间**：必填，API 请求超时时间，单位为毫秒（ms）。
   - **缓存开关**：即选择授权结果是否开启缓存。
     - 开启缓存后，如果 auth-server 返回 success，该结果会被缓存，缓存有效期内不会再次请求 auth-server。
     - 缓存时间不能超过 1 小时，不能低于 5s。
   - **路由策略**：必选，表示当网关接收到请求后使用的路由策略。详见 创建路由规则。
   - **系统集群**：必选，选择提供该认证服务的后端集群。

- **外部 API 参数**：即 `request.context` 中的参数，可以从请求的 Header、Body、Cookie 和 Query 中提取。



如上图配置所示，网关在调用 auth-server 时，会进行如下参数提取：

- 从 Header 中取出 headerKey 的 value。
- 从 Query 中取出 queryKey 的 value。
- 从 Body 中取出 bodyKey 的 value。如 Body 为 `{"bodyKey":"abc", "other": "xx"}`，则会取出 `abc` 这个 value。使用上述所有的 kv 构造一个 context，然后调用 auth-server，最终的请求如下：

```
{
"context":{
"bodyKey":"abc",
"queryKey":"q",
"headerKey":"h"
}
}
```

4. 确认 API 配置无误后，单击 **创建** 即可。

绑定 API

外部授权 API 配置完成后，即可前往绑定 API 生效。操作步骤如下：

1. 在 **API 发布 > 外部授权 API** 页面，单击刚刚创建的外部授权 API 名称，进入其详情页。

2. 在 **外部授权 API 详情 > 绑定的 API** 标签页下，单击 **绑定 API**。

3. 在弹出的窗口中，选择需要绑定的 API 后，单击 **确定** 即可。

后续如需解绑，直接在 **外部授权 API 详情 > 绑定的 API** 中，选择需要解绑的 API，单击其右侧的 **解除绑定** 按钮，然后确定即可。

> **说明**
>
> - 一个业务 API 只能绑定一个外部授权 API。
> - 外部授权 API 只能绑定相同前端协议的业务 API。
> - 若外部授权 API 的协议为 SOFARPC 则它只能绑定 post 或 put 方法的业务 API。

1.6.2. 管理外部授权 API

在 **外部授权 API** 页面，您可以根据业务需要，进行以下操作：

查看外部授权 API

在外部授权 API 列表页，您可以查看外部授权 API 的概览信息，包括外部授权 API 名称、协议类型、请求路径/接口名称、后端服务类型、绑定 API 数量以及创建时间。

查看外部授权 API 详情

在外部授权 API 列表中，您还可以单击任一外部授权 API 名称，进入该 API 的详情页，查看详细信息，主要包括以下信息：

- **基本信息**：展示了外部授权 API 名称、协议类型、创建人、创建时间。

- **绑定的 API**：
  - 列出了该外部授权 API 绑定的所有 API 及其基本信息，包括 API 名称、API 分组、发布状态、前端协议类型、创建时间等。
  - 在此列表，您还可以单击任一 API 名称，进入该 API 的详情页查看 API 的具体信息及监控数据等。
- **配置信息**：展示了该外部授权 API 的详细配置，包括请求路径/接口名称、认证系统集群、超时时间以及参数配置等。

编辑外部授权 API

您可以通过以下任一方式编辑外部授权 API。除了协议类型与方法无法修改，其他配置均可修改。

- 在外部授权 API 列表中，找到目标 API，单击右侧 **操作** 列中的 **编辑** 按钮。
- 在外部授权 API 的详情页，单击右上角的 **编辑** 按钮。

删除外部授权 API

> **说明**
>
> 如果当前外部授权 API 已绑定了 API，则无法被删除。需将对应 API 解绑，才可删除。

您可以通过以下任一方式删除参数映射规则：

- 在外部授权 API 列表页，单击外部授权 API 规则右侧 **操作** 列中的 **删除**，在确认框中单击 **确定** 即可删除该规则。
- 在外部授权 API 的详情页，单击右上角的 **删除** 按钮，然后单击 **确定** 即可。

1.7. 授权管理

1.7.1. 创建授权对象

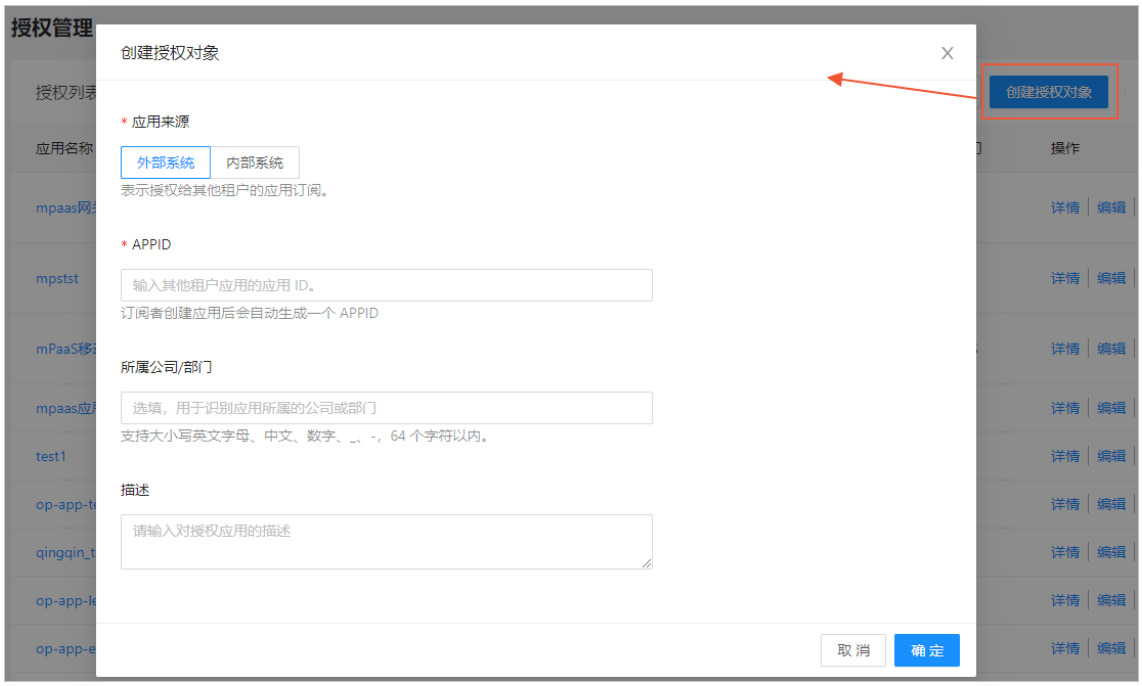
授权管理用于定义授权对象的基本信息，一个授权对象可以绑定多个 API。在 API 订阅者创建应用后，API 发布者可根据应用创建授权对象。

操作步骤

1. 登录 API 网关控制台。
2. 在左侧导航栏中单击 **API 发布** > **授权管理**，进入授权列表。
3. 单击授权列表右上方的 **创建授权对象**。
4. 在弹出的 **创建授权对象** 窗口中，选择或输入相关信息：
   - **应用来源**：必选。
     - **外部系统**：表示授权给其他租户的应用订阅。
     - **内部系统**：表示授权给当前租户的当前环境下的应用订阅。
   - 对于外部系统的应用，您需要输入其 **APPID**，可自动获取其应用名称与类型。

     > **说明**
     >
     > 当应用创建成功后，系统会自动给应用分别配一个 APPID，可在 **API 订阅** > **应用管理** 中的应用详情页查看。

   - 对于内部系统的应用，您需要输入其 **应用名称**，可自动获取其 APPID 与类型。
   - **所属公司/部门**：可选，用于识别应用所属的公司或部门。
   - **描述**：可选，输入对该授权对象的备注信息。



5. 单击 **确定**，即可完成授权对象的创建。

   完成授权对象的创建后，您就可以进行 绑定 API 的操作。

1.7.2. 管理授权对象

在 **授权管理** 页面，您可以根据业务需要，查看并管理已创建的授权对象。

编辑授权对象

打开授权对象编辑界面有以下两种方式：

- 在授权列表页，单击授权应用右侧 **操作** 列中的 **编辑** 即可打开编辑应用界面。
- 单击授权应用名进入授权应用详情页后，单击页面右上方的 **编辑** 即可打开编辑授权对象界面。

编辑授权对象 界面与创建授权对象界面内容一致，您可修改授权应用的 **应用来源**、**APPID/应用名称**、**所属公司/部门** 以及 **描述**。

完成后单击 **确定**，即可完成授权对象信息的修改。

查看授权应用详情

在授权应用列表页，单击授权应用名可进入授权应用详情页。

绑定的 API

在授权应用详情页的 **绑定的 API** 标签页中，可查看当前授权应用已绑定的 API。

在绑定的 API 列表中，您可以：

- **查看 API 详情**：单击 API 名称，即可跳转至该 API 的详情页，您可查看该 API 的 **授权对象**、**配置信息**、**版本管理** 等详细信息。
- **筛选前端协议类型**：
  i. 单击 **前端协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的前端协议并单击 **确定** 即可进行筛选。
- **筛选 API 状态**：
  i. 单击 **状态** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择 **已发布** 或 **未发布**，并单击 **确定** 即可进行筛选。
- **搜索 API**：在 API 列表右上方的搜索框中，可通过 API 名称搜索已绑定的 API。
- **更改展示设置**：
  - **更改列设置**：在 API 列表右上方的功能区域，单击 **列设置** 按钮。在打开的列展示框中，您可勾选或取消 API 列表的展示项目，也可将某一项固定在左/右侧，或拖动展示项进行排序。
  - **调整列密度**：在 API 列表右上方的功能区域，单击 **密度** 按钮。在打开的密度选择框中，可调整 API 列表的展示密度。

联系方式

联系方式 标签页仅出现在 **应用来源** 为 **外部系统** 的授权对象中。

添加联系人

1. 在 **联系方式** 标签页中，单击联系人列表右上方的 **添加联系人**。
2. 在弹出的 **添加联系人** 窗口中，输入相关信息：

- **姓名**：必填，输入联系人姓名。
- **所属公司/部门**：选填，用于标识该联系人的公司/部门。
- **电话**：必填，输入该联系人的电话。
- **邮箱**：选填，输入该联系人的电子邮箱。

3. 单击 **确定**，完成联系人添加。

删除授权应用

在授权应用列表页，单击 **操作** 列中的 **删除**，在确认框中单击 **确定** 即可删除当前授权应用。

> **说明**
>
> 如果当前授权应用已绑定 API，则无法被删除，需前往 API 详情页 解除绑定 后再删除授权应用。

1.7.3. 绑定/解绑 API

授权对象创建成功后，您可在该授权对象下批量绑定或解绑指定的 API。

绑定 API

1. 在 **授权管理** 页面，找到目标应用，单击其名称进入该授权应用详情页。
2. 在打开的 **绑定的 API** 标签页中，单击 API 列表右上方的 **绑定 API**。

3. 在弹出的 **绑定 API** 窗口中，勾选需要绑定的 API。
4. 单击页面底部的 **确认**，即可完成绑定。

解绑 API

1. 在 **授权管理** 页面，找到目标应用，单击其名称进入该授权应用详情页。
2. 解除 API 绑定。
   - **单个解绑**：找到需要解绑的 API，单击 **操作** 列中的 **解除绑定**，在弹出的确认框中单击 **确定**，即可解除该 API 的绑定。
   - **批量解绑**：若需要一次解绑当前授权应用下的多个 API，可先勾选多个 API 后，单击页面底部的 **批量解除绑定**，在弹出的确认框中单击 **确定**，即可批量解除绑定多个 API。

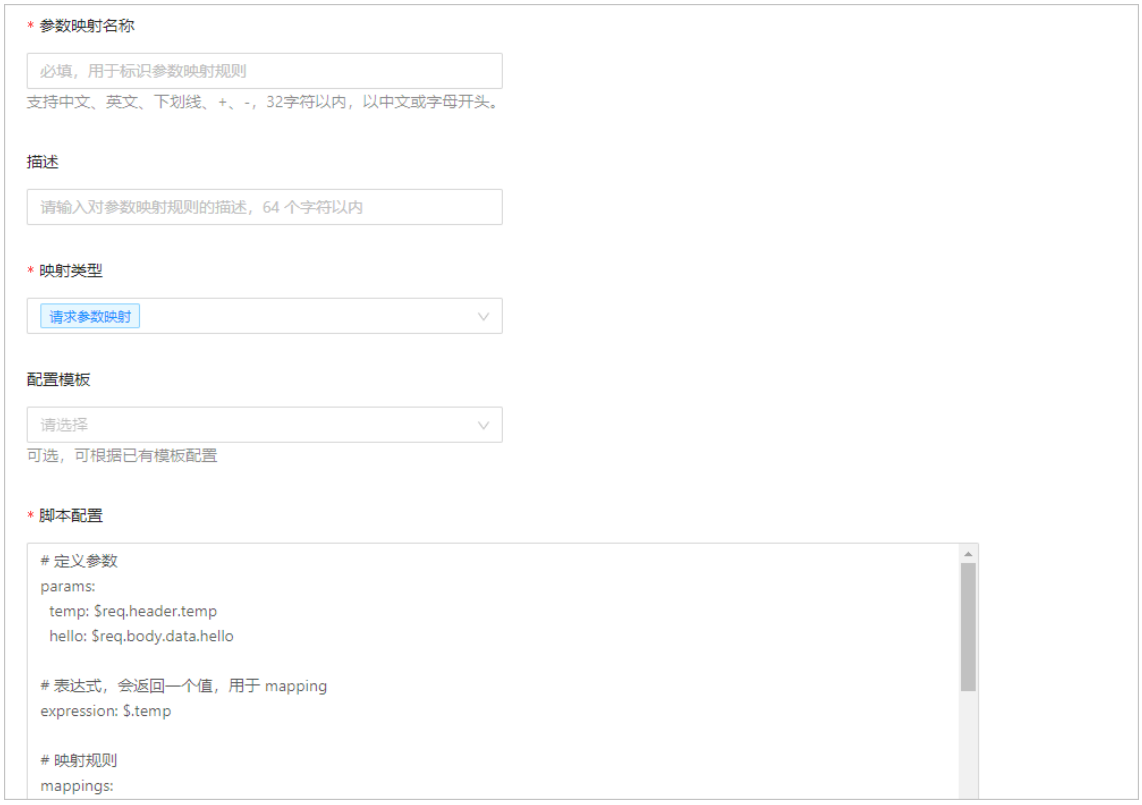
1.8. 参数映射

1.8.1. 创建参数映射

API 网关支持参数映射。参数映射是将请求或响应参数中的不规范或者不统一的参数，映射为符合客户端或者服务端规范的参数。该功能常用于多部门协同、实现参数统一规范，以及实现统一错误码等场景。

操作步骤

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > 参数映射**。
2. 在参数映射列表页，单击列表右上方的 **创建参数映射**。
3. 在新打开的 **创建参数映射** 页面中，选择或输入相关信息：
   - **参数映射名称**：必填，用于识别参数映射规则。支持中文、英文、下划线、加号（+）、连接符（-），32字符以内，以中文或字母开头。
   - **描述**：选填，用于描述该参数映射的业务意义或使用范围等，64 个字符以内。
   - **映射类型**：必选，用于指定该参数映射的使用场景，是 **请求参数映射** 还是 **响应参数映射**。
   - **配置模板**：可选，可从下拉菜单中选择已有的模板，即可在该模板的基础上进行脚本配置。
   - **脚本配置**：必填，根据业务需要配置具体的参数映射模板脚本。详细的模板配置语法，参见 配置参数映射模板。



4. 确认信息无误后，单击 **创建** 即可。

后续操作

参数映射创建成功后，即可在后续创建 API 或编辑 API 时，选择该映射配置实现请求或响应参数映射。

1.8.2. 配置参数映射模板

API 网关支持通过配置参数映射模板的方式对请求和响应数据进行自动化修改。本文将详细介绍参数映射模板的配置语法及注意事项等。

模板示例

- **请求模板**

```
# 请求 body

{
 "x" : {"name": "jack"},
  "y" : {
   "z" : ["a", "b", "c"]
  }
}
```

```
# 定义参数
params:
  temp: $.Req.Header.temp
  body1: $.Req.Body.x
  body2: $.Req.Body.y
  bodyArray: $.Req.Body.y.z[0]
  body: $.Req.Body

# 表达式，会返回一个值，用于 mapping
expression: $.temp

# 映射规则
mappings:
  # 映射规则，至少一个
  ## 如果 expression 返回了 hello, 也就是 header temp 的 value 是 hello
  hello:
    # 执行下面的映射逻辑
    ## body 覆盖模板 可选
    bodyOverride: '[{"name": "{{ $.temp }}"}, {"temp": {{ $.body1}}}]'
    ## request header 设置，先执行 delete，再执行 add 可选
    header:
      addKeyValue:
        ah: $.temp
      deleteKey:
        - temp
    ## request cookie 设置，先执行 delete，再执行 add 可选
    cookie:
      addKeyValue:
        ac: $.temp
      deleteKey:
        - cookiekey
    ## request query 设置，先执行 delete，再执行 add 可选
    query:
      addKeyValue:
        aq: $.temp
      deleteKey:
        - querykey
  中文:
    # 执行下面的映射逻辑
    ## body 覆盖模板 可选
    bodyOverride: |
```

```
      [
       {"name": {{ $.body1 }}},
        {"temp": {{ $.body2}}},
        {"array":{{$.bodyArray}}},
        {"body": {{$.body}}}
      ]

# 默认映射规则
default:
  bodyOverride: '{"name": "{{ $.temp }}"}'
  header:
    addKeyValue:
      ah: $.temp
    deleteKey:
      - temp
  cookie:
    addKeyValue:
      ac: $.temp
    deleteKey:
      - cookiekey
  query:
    addKeyValue:
      aq: $.temp
    deleteKey:
      - querykey
```

- **响应模板**

```
# 响应 body

{
 "x" : {"name": "jack"},
  "y" : {
   "z" : ["a", "b", "c"]
  }
}
```

```
# 定义参数
params:
    temp: $.Resp.Header.temp
    bodyx: $.Resp.Body.x
    bodyy: $.Resp.Body.y
    bodyArray: $.Resp.Body.y.z[0]
    body: $.Resp.Body

# 表达式，会返回一个值，用于 mapping
  expression: $.temp

# 映射规则
  mappings:
# 映射规则，至少一个
## 如果 expression 返回了 hello，也就是 header temp 的 value 是 hello
    hello:
# 执行下面的映射逻辑
## body 覆盖模板 （可选操作）
      bodyOverride:|
[{"name":"{{ $.temp }}"},{"temp":{{ $.body1}}}]
## response header 设置，先执行 delete，再执行 add； （可选操作）
      header:
        addKeyValue:
          ah: $.temp
        deleteKey:
- temp
    world:
# 执行下面的映射逻辑
## body 覆盖模板； （可选操作）
      bodyOverride:|
"xx"

# 默认映射规则
default:
    bodyOverride:|
{"name":{{ $.temp }}}
```

如上示例所示，模板语法主要分为以下几个对象：

- `params` ：用于从请求或响应中提取参数。详见下文 params：提取参数。
- `expression` ：一个表达式，该表达式会返回一个值。详见下文 expression：表达式。
  - 如果表达式是一个判断表达式，则返回 true/false。
  - 如果表达式是一个字符串拼接，则返回一个字符串。
- `mappings` ：根据 `expression` 的返回内容，配置不同的映射规则。详见下文 mappings：映射规则。
- `default` ：（可选操作），存储一个默认的映射规则。

params：提取参数

`params` 可以从 request 或者 response 中提取参数，格式为 `k:v` 。其中 key 为自定义的参数名，value 是一个参数表达式，格式为： `$.Type.Location.key` 。

- `Type` ：可以是 Req 或者 Resp，分别表示请求和响应。
- `Location` ：可以是 Header、Query、Cookie 或者 Body。
  - Resp 只支持 Header 和 Body。
  - Body 必须为 JSON 格式。

示例

- **示例 1**：从请求 Header 中提取参数

```
请求的 header：temp = headervalue
定义：temp = $.Req.Header.temp
结果：==> temp = headervalue
```

- **示例 2**：从请求 Cookie 中提取参数

```
请求的 cookie：temp = cookievalue
定义：temp = $.Req.Cookie.temp
结果：==> temp = cookievalue
```

- **示例 3**：从请求 Body 中提取参数

```
请求的 body：{"hello":{"world":["1","2","3"]}}
定义：temp = $.Req.Body.hello.world[0]
结果：==> temp =1
```

- **示例 4**：从响应 Body 中提取参数

```
响应的 body：{"hello":{"world":["1","2","3"]}}
定义：temp = $.Resp.Body.hello.world[0]
结果：==> temp =1
```

expression：表达式

`expression` 表达式会返回一个值。表达式可以引用 `params` 中定义的变量，形式为 `$.name` 。

- **示例 1**

```
请求的 body：{"hello":{"world":["1","2","3"]}}
定义：temp = $.Req.Body.hello.world[0]
表达式： $.temp ==1
结果：true
```

- **示例 2**

```
请求的 cookie：temp = cookievalue
定义：temp = $.Req.Cookie.temp
表达式： $.temp
结果：cookievalue
```

mappings：映射规则

映射规则支持覆盖 Body、修改 Header、Query 和 Cookie。其中 Response 只支持修改 Body 和 Header。

覆盖 Body

`bodyOverride` 即覆盖 Body。值是 Go template 标准模板，可以通过 `$.name` 引用 `params` 中定义的变量。示例如下：

```
请求的 body：{"hello":{"world":["1","2","3"]}}
定义：temp = $.Req.Body.hello.world[0]
bodyOverride：'{"name": "{{ $.temp }}"}'
结果：{"name":"1"}
```

其中，`bodyOverride` 是一个字符串模板，所以要用前后要加一个单引号。示例如下：

```
# 只能放在同一行，不能换行
bodyOverride:'{"name": "{{ $.name }}" }'
```

如果 `bodyOverride` 很长，一行放不下，可以使用以下方式多行输入。示例如下：

```
# 在 bodyOverride 后加一个 |，即可多行输入
# 换行输入要空两格，符合 yaml 标准语法
bodyOverride:|
[
{"name":{{ $.body1 }}},
{"temp":{{ $.body2}}},
{"array":{{$.bodyArray}}},
{"body":{{$.body}}}
]
```

`bodyOverride` 的内容是一个 JSON 模板，要保证渲染后的结果是一个 JSON 字符串。

- 如果引用的参数变量是 String 类型，则需要使用 `""` ，如： `{"name": "{{ $.string }}" }` 。
- 如果引用是参数变量是其他类型，则不需要 `""` ，如： `{"name": {{ $.object }} }` 。

因此，配置模板时，必须确认参数的类型。示例如下：

```
假设 param.str = "jack"
定义 bodyOverride: '{"name": "{{ $.str }}" }'
渲染结果： {"name": "jack"}

符合标准 json

------------

假设 param.str = "jack"
定义 bodyOverride: '{"name": {{ $.str }} }'
渲染结果： {"name": jack}

不符合标准 json


------------

假设 param.obj = {"firstName":"tom"}
定义 bodyOverride: '{"name": {{ $.obj }} }'
渲染结果： {"name": {"firstName":"tom"}}

符合标准 json
------------

假设 param.obj = {"firstName":"tom"}
定义 bodyOverride: '{"name": "{{ $.obj }}" }'
渲染结果： {"name": "{\"firstName\":\"tom\"}"}

符合标准 json，但是不符合预期
```

修改 Query、Header、Cookie

这三个 key 均可以定义 addKeyValue 和 deleteKey。

- addKeyValue：kv 结构，会在相应位置添加对应的 kv，通过 `$.name` 引用 `params` 中定义的变量。
- deleteKey：list，会将定义的所有 key 从相应位置删除。

执行时，会先执行删除操作，再执行添加操作。示例如下：

```
请求的 cookie：temp = cookievalue
定义：temp = $.Req.Cookie.temp
映射：cookie:
        addKeyValue:
            newkey: $.temp
        deleteKey:
-     temp
结果：cookie: newkey = cookievalue
```

1.8.3. 管理参数映射

在 **参数映射** 页面，您可以根据业务需要，查看并管理已创建的参数映射规则。

查看参数映射

在参数映射列表页，您可以查看该参数映射的概览信息，包括参数映射名称 、配置模板、绑定 API 数量、描述 、映射类型以及创建时间。

查看参数映射详情

在参数映射列表中，您还可以单击任一参数映射名称，进入该映射规则的详情页，查看详细信息，主要包括以下信息：

- **基本信息**：展示了参数映射名称、配置模板、映射类型、创建人、创建时间与描述。
- **绑定的 API**：
  - 列出了绑定了该映射规则的所有 API 及其基本信息，包括 API 名称，API 分组、发布状态、前端协议类型、创建时间等。
  - 在此列表，您还可以单击任一 API 名称，进入该 API 的详情页查看 API 的具体信息及监控数据等。
- **配置详情**：展示了该参数映射规则的配置模板以及具体的脚本配置数据。

编辑参数映射

> **说明**
>
> 如果当前参数映射规则已绑定 API，则无法被编辑。需前往 API 编辑页面，将对应 API 的参数映射规则替换为其他规则后，方可编辑当前参数映射规则。

您可以通过以下任一方式编辑参数映射规则：

- 在参数映射列表中，找到目标规则，单击右侧 **操作** 列中的 **编辑** 按钮。
- 在参数映射的详情页，单击右上角的 **编辑** 按钮。

删除参数映射

> **说明**
>
> 如果当前参数映射规则已绑定 API，则无法被删除。需前往 API 编辑页面，将对应 API 的参数映射规则替换为其他规则后，方可删除当前参数映射规则。

您可以通过以下任一方式删除参数映射规则：

- 在参数映射列表页，单击参数映射规则右侧 **操作** 列中的 **删除**，在确认框中单击 **确认** 即可删除该规则。

- 在参数映射的详情页，单击右上角的 **删除** 按钮，然后单击 **确认** 即可。

1.9. IP 策略管理

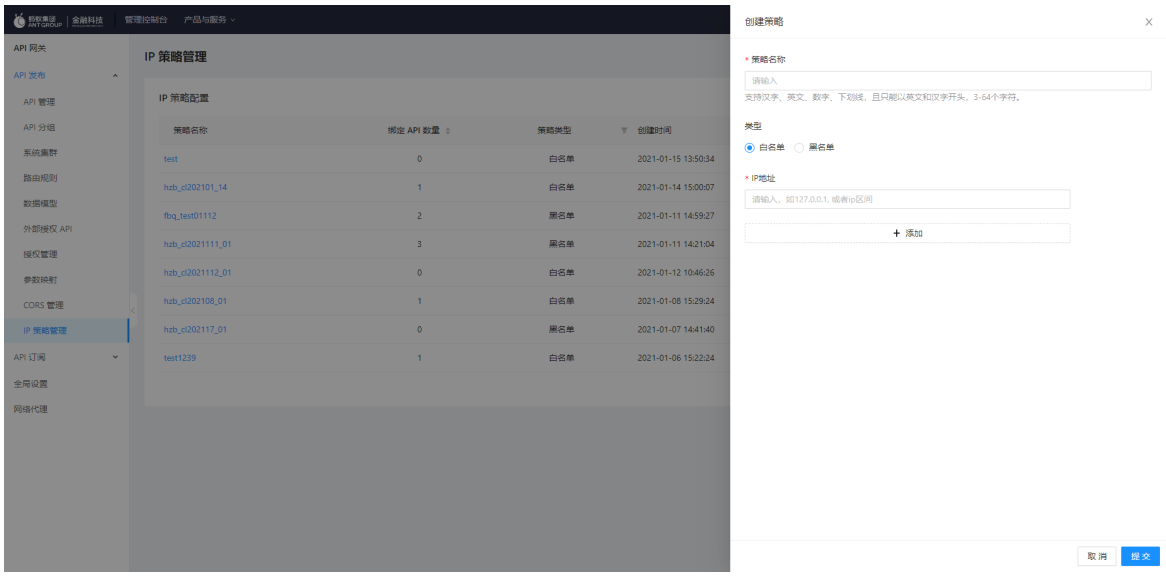
1.9.1. 创建 IP 策略

IP 策略是用来控制访问 API 的 IP 地址，您可以通过设置 IP 策略来允许或拒绝某个 IP 地址访问 API，本文将介绍在 API 网关中如何创建 IP 策略。

> **说明**
>
> IP 策略和 API 是两个独立的概念，只有将创建好的 IP 策略开启并绑定到 API 上后，该策略才会对 API 生效。

操作步骤

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布 > IP 策略管理**，进入 IP 策略管理页面。
3. 在页面右上角单击 **创建策略**，在弹出的窗口中输入要创建的策略信息，完成后单击 **提交**。



字段说明如下表所示：

字段	描述
策略名称	IP 策略的名称。支持英文字母、中文、数字、_、-、:、.，且只能以英文和汉字开头，3-64个字符。
类型	白名单：仅允许白名单类型的 IP 地址访问 API。 例如：设置 IP 白名单为 127.0.0.0，然后将该策略绑定到 API a，则API a **只能**被 127.0.0.0 的 IP 地址访问。
	黑名单：不允许黑名单类型的 IP 地址访问 API。 例如：设置 IP 黑名单为 127.0.0.0，然后将该策略绑定到 API b，则API b **不能**被 127.0.0.0 的 IP 地址访问。
IP 地址	输入需要允许或者禁止访问 API 的 IP 地址或 IP 地址区间。

1.9.2. 管理 IP 策略

在 IP 策略管理页面，您可以查看并管理已创建的 IP 策略。

开启/关闭 IP 策略

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 在 **IP 策略配置** 列表中，通过 **操作** 列中的按钮来 **开启**（或 **关闭**）该策略。

> **说明**
>
> 当您创建好一个 IP 策略后，该策略默认是关闭状态，您需要单击 **开启** 使该策略生效。

编辑 IP 策略

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 打开策略编辑界面有以下两种方式：
   - 在 IP 策略管理页面，单击策略右侧 **操作** 列中的 **编辑** 即可打开编辑规则界面。
   - 在 IP 策略管理页面，单击策略名称进入策略详情页后，单击页面右上方的 **编辑** 即可打开编辑规则界面。

4. 完成后 单击 **提交** 完成对策略的修改。

查看 IP 策略详情

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 单击目标策略名称进入 **策略管理详情** 页面。

绑定的 API

在 **绑定的 API** 标签页下，您可以看到此 IP 策略已经绑定的 API 信息，包括名称、分组、协议类型等。

单击 API 名称，即可跳转至该 API 的详情页。

规则详情

在 规则详情 标签页下，您可以看到此IP 策略的策略的详细信息，如下图所示。

删除 IP 策略

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 删除策略有以下两种方式：
   - 在 IP 策略管理页面，单击策略右侧 **操作** 列中的 **删除** 即可删除 IP 策略。
   - 在 IP 策略管理页面，单击策略名称进入策略详情页后，单击页面右上方的 **删除** 即可删除 IP 策略。

> **说明**
>
> 您只能删除没有绑定 API 的策略，否则需要先解绑 API 后，再进行删除操作，具体请参见 绑定/解绑 API 。

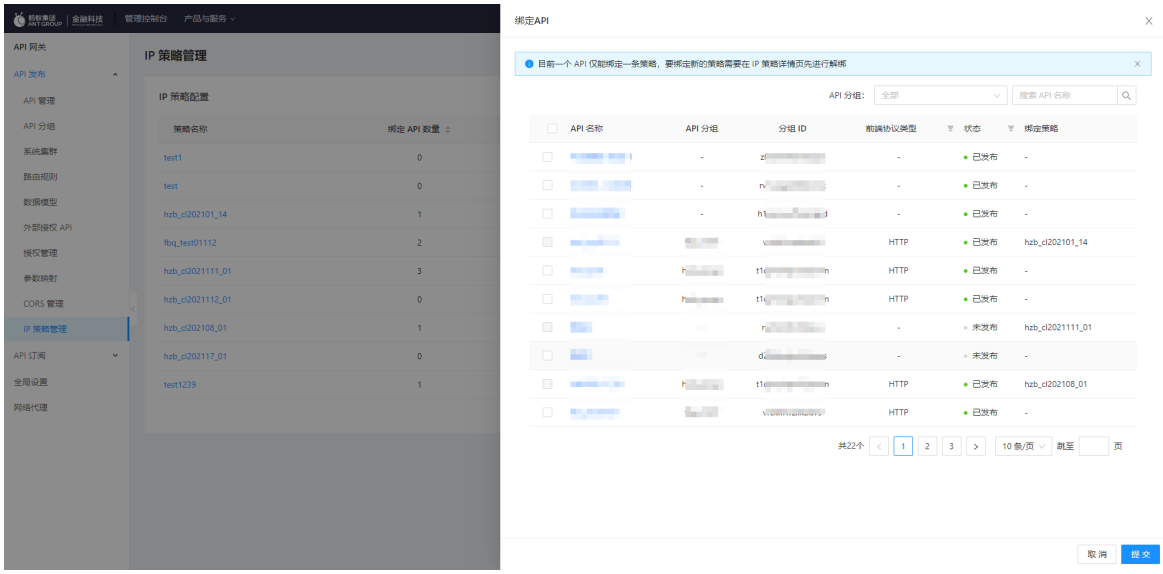
1.9.3. 绑定/解绑 API

创建 IP 策略则后，您可以根据业务需求把 IP 策略绑定到指定的 API 上。

绑定/解绑 API

绑定 API

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 在目标 IP 策略右侧单击 **绑定 API**，在弹出的窗口中选择需要绑定的 API。



说明

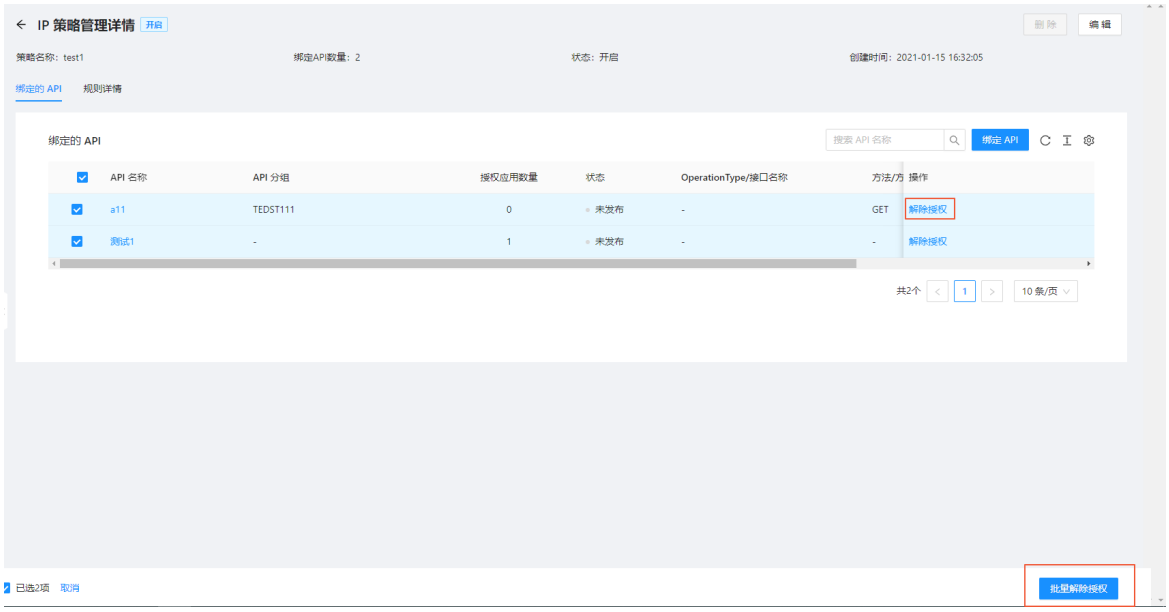
- 一个 IP 策略可以绑定多个 API。
- 一个 API 仅能绑定一个 IP 策略，要绑定新的策略需要在 IP 策略详情页先进行解绑。

解绑 API

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **API 发布** > **IP 策略管理**，进入 IP 策略管理页面。
3. 单击策略名称，在 **IP 策略管理详情** 页面可以进行解绑 API 操作。
   - **单个解绑**

     找到需要解绑的 API，单击 API 右侧 **操作** 列中的 **解除授权**，在弹出的确认框中单击 **确定**，即可解除该 API 的绑定。
   - **批量解绑**

     若需要一次解绑当前规则下的多个 API，可先通过左侧的复选框勾选多个 API 后，单击页面底部的 **批量解除授权**，在弹出的确认框中单击 **确定**，即可批量解除 API 的绑定。



1.10. CORS 配置

1.10.1. 创建 CORS 规则

跨域访问是指请求一个与自身资源不同源（不同的域名、协议或端口）的资源。不同源可以是不同的域名、协议或端口。浏览器出于安全考虑设置了同源策略，限制了从脚本内发起跨域请求。但在实际应用中，经常会发生跨域访问。为此，W3C 提供了一个标准的跨域解决方案：跨域资源共享（Cross-Origin Resource Sharing，CORS），支持安全的跨域请求和数据传输。

操作步骤

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布** > **CORS 管理**，进入 **CORS 配置** 页面。
2. 单击 CORS 列表右上方的 **创建 CORS**。
3. 在弹出的 **新建 CORS** 窗口中，选择或输入相关信息：
   - **CORS 名称**：必填，输入 CORS 的名称用以识别 CORS 的作用。支持大小写英文字母、中文、数字、下划线（_）、中划线（-），32 个字符以内。
   - **CORS 状态**：必选，即创建后的 CORS 规则的状态，可选择 **开启** 或 **关闭**，默认为 **关闭**。

     > **说明**
     >
     > 若选择关闭，您可在创建后通过 CORS 规则右侧 **操作** 列中的 **开启** 按钮将指定的 CORS 规则开启。

   - **允许来源**：必填，输入允许的来源域名。
     - 域名前需加 `http://` 或 `https://` 。
     - 若想允许所有来源的域名，可输入星号（*）。
     - 您可单击输入框下方的 **+ 添加允许来源** 来添加多个域名。

- **允许方法**：必填，输入 CORS 规则中允许的请求方法。可选 **GET**、**POST**、**PUT** 或 **DELETE**，允许多选。
- **允许标头**：选填，输入允许跨域的所有请求头信息字段。
  - 允许标头中可使用 * 通配符。
  - 您可单击输入框下方的 **+ 添加允许标头** 来添加多个允许标头。
- **公开标头**：选填，输入跨域访问允许查看的返回头信息字段。
  - 公开标头中 **不允许** 使用 * 通配符。
  - 您可单击输入框下方的 **+ 添加公开标头** 来添加多个公开标头。
- **有效期**：选填，输入浏览器的最大缓存时间。
- **允许凭证**：表示是否允许发送 Cookie。
  - 默认情况下为 **关闭**，即不允许发送 Cookie。
  - 选择 **开启**，即表示服务器明确许可，Cookie 可以包含在请求中，一起发送给服务器。

4. 单击 **确定**，即可完成 CORS 规则的创建。

注意事项

- 通过 **创建 CORS** 按钮所创建的 CORS 规则的生效范围均为 **API 级别**。
- 在 CORS 规则列表中，有一条名为 **当前环境级别 CORS 规则** 的置顶规则，生效范围为 **当前环境级别**。该规则无法删除，默认状态为关闭。
  - 如果开启该环境级别规则，则对所有 API 生效。但如果单个 API 已额外绑定其他规则，则已绑定的规则优先生效。
  - 如果关闭该环境级别规则，则执行单个 API 额外绑定的规则。

后续操作

CORS 规则创建后，您可通过 绑定 API 功能将规则与 API 进行绑定。

1.10.2. 管理 CORS 规则

在 **CORS 配置** 页面，您可以根据业务需要，查看并管理已创建的 CORS 规则。

开启/关闭 CORS 规则

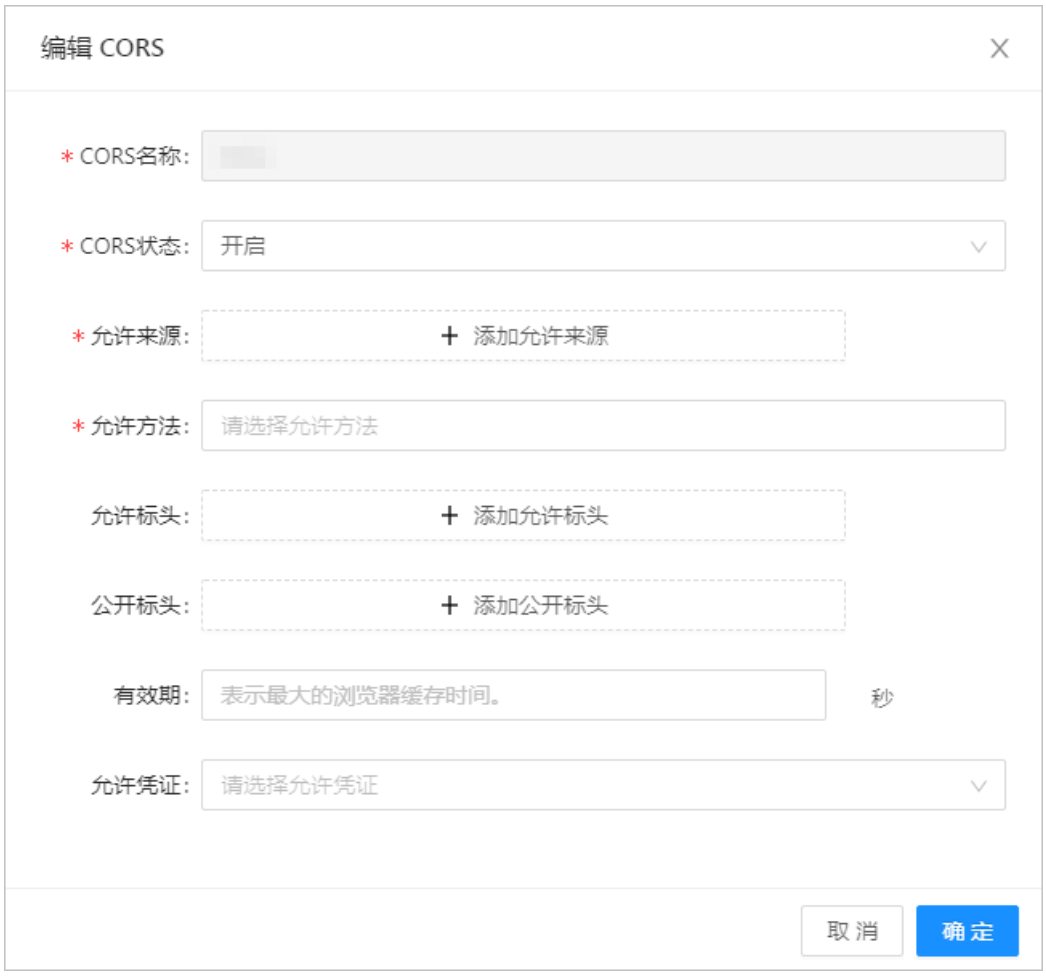
创建规则时，如果您选择的规则状态为 **关闭**（或 **开启**），在创建完成后，您可在规则列表中通过 **操作** 列中的按钮来 **开启**（或 **关闭**）该规则。

说明

- 当前环境级别规则：
  - 如果开启环境级别规则，则对所有 API 生效。但如果单个 API 已额外绑定其他规则，则已绑定的规则优先生效。
  - 如果关闭环境级别规则，则执行单个 API 额外绑定的规则。
- API 级别规则：
  - 如果开启/关闭 API 级别规则，则对该规则下所有已绑定的 API 生效。
  - 如果想关闭规则下的某个 API，则可从规则下将该 API 解除绑定。如果需要再度开启该 API，则将规则与 API 重新绑定即可。

编辑 CORS 规则

1. 打开规则编辑界面有以下两种方式：
   - 在规则列表页，单击规则右侧 **操作** 列中的 **编辑** 即可打开编辑规则界面。
   - 单击规则名进入规则详情页后，单击页面右上方的 **编辑** 即可打开编辑规则界面。
2. 打开的 **编辑 CORS** 界面与创建 CORS 规则界面内容一致，您可修改除 **CORS 名称** 以外的其余项目。



3. 单击 **确定**，即可完成对 CORS 规则的修改。

查看 CORS 规则详情

在 CORS 规则列表页，单击 CORS 规则名可进入 CORS 规则详情页。

绑定的 API

在 CORS 规则详情页的 **绑定的 API** 页签中，可查看当前 CORS 规则已绑定的 API。

在绑定的 API 列表中，您可以：

- **查看 API 详情**：单击 API 名称，即可跳转至该 API 的详情页，您可查看该 API 的 **配置信息**、**版本管理**、**监控**、**流量治理**、**CORS**、 **授权对象**、**API 调试** 等详细信息。
- **筛选前端协议类型**：
  i. 单击 **前端协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的前端协议并单击 **确定** 即可进行筛选。
- **筛选 API 状态**：
  i. 单击 **状态** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择 **已发布** 或 **未发布**，并单击 **确定** 即可进行筛选。
- **筛选后端服务类型**：
  i. 单击 **后端服务类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的后端服务并单击 **确定** 即可进行筛选。
- **筛选后端协议类型**：
  i. 单击 **后端协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的后端协议并单击 **确定** 即可进行筛选。
- **搜索 API**：在 API 列表右上方的搜索框中，可通过 API 名称搜索已绑定的 API。
- **更改展示设置**：
  - **更改列设置**：在 API 列表右上方的功能区域，单击 **列设置** 按钮。在打开的列展示框中，您可勾选或取消 API 列表的展示项目，也可将某一项固定在左/右侧，或拖动展示项进行排序。
  - **调整列密度**：在 API 列表右上方的功能区域，单击 **密度** 按钮。在打开的密度选择框中，可调整 API 列表的展示密度。

规则详情

在 CORS 规则详情页的 **规则详情** 页签中，可查看当前 CORS 规则的规则详情。其中的展示信息与创建时相同。

删除 CORS 规则

在 CORS 规则列表页，单击 CORS 规则右侧 **操作** 列中的 **删除**，在确认框中单击 **确定** 即可删除当前 CORS 规则。

> **说明**
>
> - 无法删除置顶的环境级别规则。
> - 无法删除已绑定 API 的规则，如需删除，请先 解绑 API。

1.10.3. 绑定/解绑 API

创建 CORS 规则后，您可在规则下绑定/解绑指定的 API。

绑定 API

1. 在 CORS 规则列表页，单击规则右侧 **操作** 列中的 **绑定 API**。
2. 在打开的 **绑定的 API** 标签页中，单击 API 列表右上方的 **绑定 API**。

3. 在弹出的 **绑定 API** 窗口中，勾选需要绑定的 API。
4. 单击页面底部的 **确认**，即可完成绑定。

> **说明**
>
> 一个规则下可绑定多个 API，但一个 API 只可绑定一个规则。

解绑 API

1. 在 CORS 规则列表页，单击规则右侧 **操作** 列中的 **绑定 API**。
2. 在打开的 **绑定的 API** 标签页中，可看到已绑定的 API。
3. 解除 API 绑定。
   - **单个解绑**：找到需要解绑的 API，单击 API 右侧 **操作** 列中的 **解除授权**，在弹出的确认框中单击 **确定**，即可解除该 API 的绑定。
   - **批量解绑**：若需要一次解绑当前规则下的多个 API，可先通过左侧的复选框勾选多个 API 后，单击页面底部的 **批量解除授权**，在弹出的确认框中单击 **确定**，即可批量解除 API 的绑定。

1.11. 接口编排

1.11.1. 创建接口编排

前提条件

已 创建好需要编排的 API。

操作步骤

一个完整的接口编排从创建到发布共分为 4 步：

1. 创建编排
2. 配置编排流程
3. 配置流程返回参数
4. 发布接口编排

创建编排

1. 登录 API 网关控制台。
2. 在左侧导航栏到单击 **接口编排**。
3. 在 **编排设置** 页面右上角单击 **创建编排**。
4. 配置编排信息。

参数说明如下：

参数	说明
是否选择已有编排模式进行编排	默认不开启，开启后可选择一个创建好的编排为模板进行配置。
编排名称	必填，编排名称信息。
协议类型	必选，表示网关接收到请求后转发给的后端服务使用的通信协议类型。
返回数据类型	必选，选择参数返回的数据类型。

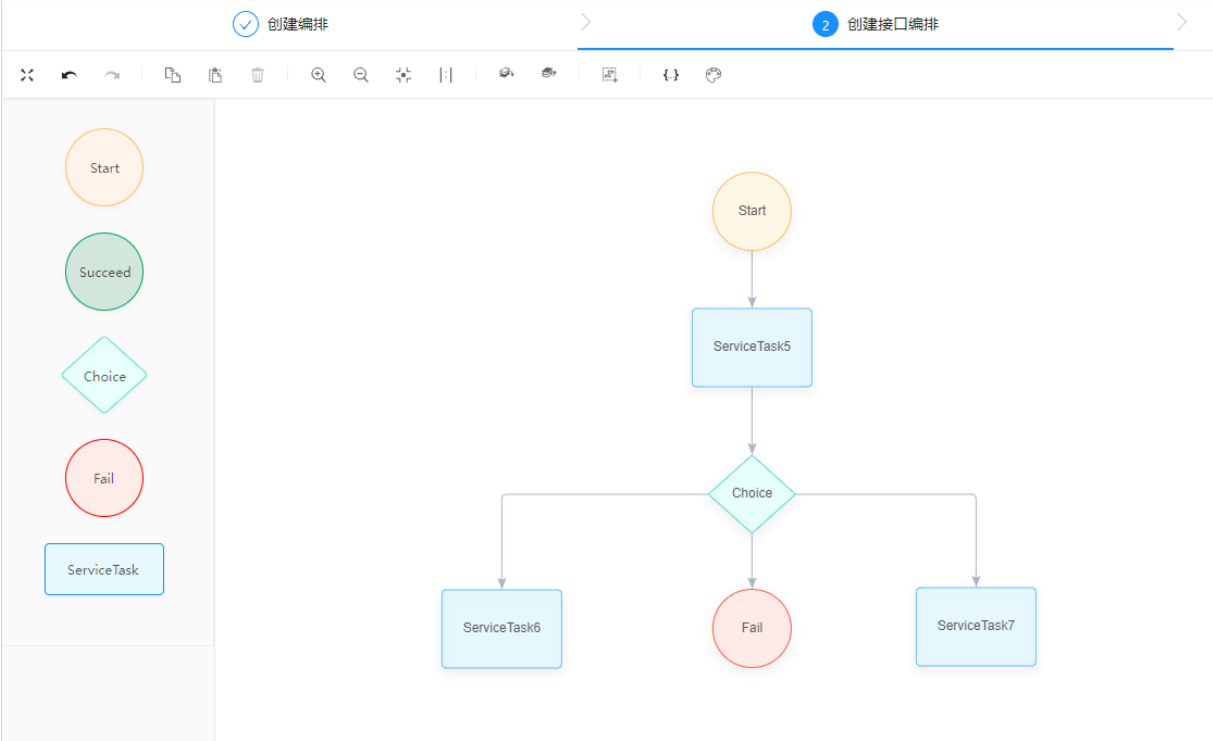
参数	说明
请求方式	必选，表明要对给定的资源执行的操作，请求方式和编排中第一个api的前端协议类型保持一致。 ○ GET：检索服务器中的数据。 ○ POST：将数据发送到服务器进行处理。 ○ PUT：更新服务器中存储的数据。 ○ DELETE：删除服务器中的数据。
请求路径	必填，针对应用设置的请求资源的 URL，通过请求路径可以定位到要请求的资源。
路径匹配规则	必选，选择请求路径的匹配规则。 ○ 绝对匹配：表示只有请求的路径和配置的路径完全一样时才能匹配成功。 ○ 前缀匹配：如希望带有某个前缀的请求全部匹配到某一个 API，则可以使用前缀匹配。
请求参数	必填，编排请求参数信息，详见 API 属性说明 > 请求参数。 说明 ○ 该参数是整个编排服务的启动参数，作为映射的入参映射到需要调用的服务（API）中，在后续的节点配置中，可以作为被调用 API 请求参数的取值来源。 ○ 此处编排请求参数的类型必须和被调用的服务（API）请求参数的类型相同，

5. 完成后单击 **下一步**，进行编排流程的配置。

配置编排流程

在此页面您可以通过简单的拖拉拽操作，将网关提供的各种节点进行编排，然后以流程的方式将接口功能展示出来，如下图所示。

> **说明**
>
> 在编排流程中，下游节点可能会调用上游节点的返回参数，为保证配置效率，您需要按照连接顺序依次编排节点信息。



网关现在支持 5 种流程节点：Start、ServiceTask、Choice、Successed 和 Fail，各节点配置方式如下。

- **Start**

  流程开始节点，可对流程的基本信息进行编辑。示例如下：

```
{
  "StateMachine": {
    "Name": "文档测试",
    "Comment": "",
    "Version": "0.0.1"
  }
}
```

- **ServiceTask**

  流程中执行具体操作的节点。选取已经在网关创建好的 API，具体配置如下：

- 基础设置

  选择需要调用的 API，并配置超时时限、重试次数参数信息。

- 参数设置

  配置被调用 API 的请求参数与编排流程参数之间的映射关系，即被调用 API 请求参数实际值的来源。

> **说明**
>
> - 映射入参（请求参数实际值来源）可以是整个流程编排的请求参数值，也可以是之前所有步骤中某个 API 的返回参数。
> - 映射入参的类型必须和请求参数的类型相同，如上图所示：请求参数 `name` 的类型为 String，如果要配置映射入参，只能选择类型同为 String 的 `people_name`，或者 `people_sex`。

- **Choice**

  流程判断节点，配置编排执行过程中的路由转发。

- **Successed**

  流程成功节点。可对该节点添加说明内容。

- **Fail**

  流程失败节点，可对该节点添加错误码等内容。示例如下：

```
{
  "ErrorCode": "404",
  "Message": "网络连接失败"
}
```

完成后单击 **下一步**，配置编排流程返回参数。

配置编排流程返回参数

您可以对编排流程中每个节点的返回结果做数据处理，然后将处理后的结果返回给调用端。

- **定义返回参数**

  编排流程执行结束后，自定义流程返回的参数。

- **映射的参数**

  流程返回参数的取值来源，按需选择对应的 API 的返回数据，针对每个API都可以返回结果中的部分参数或者完整参数映射到定义返回参数中。

完成后单击 **完成**，完成接口编排的创建。

发布接口编排

完成接口编排的创建后，在接口编排页面，你可以单击列表右侧 **发布**，将编排服务发布上线。

> **说明**
>
> - 编排流程节点中如果存在未上线 API，则无法进行上线操作。
> - 编排流程上线后，该编排流程节点中的 API 不能进行下线操作。

1.12. 全局设置

1.12.1. 跨域设置

通过在两个网络隔离的区域各部署一个网关，可以实现两个原本网络隔离区域之间的网络互通。本文主要介绍需要在 API 网关中如何进行双网关跨域设置。

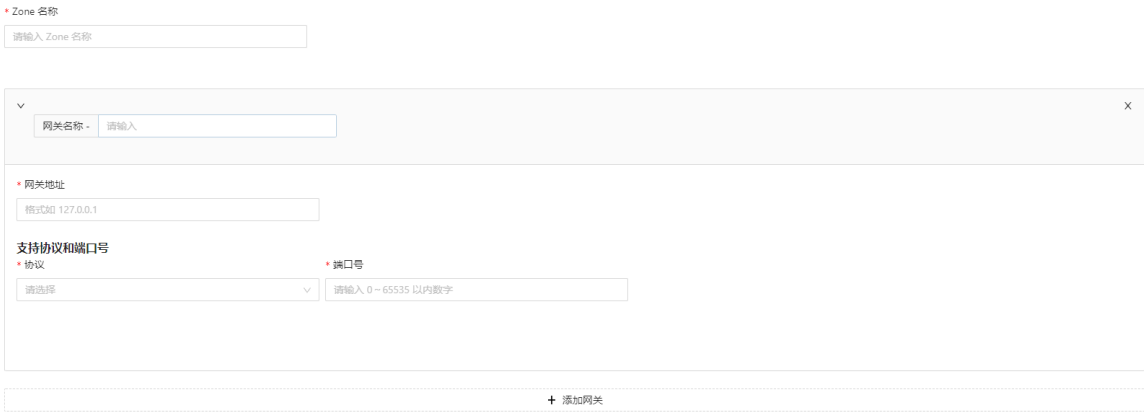
> **说明**
>
> 完成跨域设置后，您就可以通过 创建双网关跨云 API来将两个网络不通的地域打通。
>
> 只有管理员角色才可以进行跨域设置，如何配置角色权限请联系蚂蚁技术支持团队。

添加 workspace

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **全局设置** > **跨域设置**。

> **说明**
>
> - 仅 **双网关跨域通信场景** 下需要维护全局的机房（网络区域）信息，如每个机房或者网络区的网关地址、注册中心的地址等，以保证网络的正常通信。
> - 该配置为全局配置，仅管理员角色可以修改全局配置，配置完成后在其他工作区（workspace ）中仍然生效。

3. 在页面右上角单击 **编辑** > **添加 Zone**，添加 workspace。



参数说明如下：

参数	说明
Zone 名称	必填，输入添加 Zone 的名称信息。
网关名称	必填，输入添加网关的名称信息。
网关地址	必填，输入网关地址信息。
协议	必填，选择网关协议类型。 **说明** 目前仅支持 CRPC 协议。
端口号	必填，网关端口号信息，仅支持 0~65535 内的数字字符，

如果您想在一个 workspace 内添加多个网关，单击 **添加网关**，输入相应信息后保存即可。

4. 完成后单击 **提交**，完成一个 workspace 的设置。

> **说明**
>
> 至少配置两个 workspace 才能进行双网关跨域设置的操作。

1.12.2. 注册中心设置

目前 API 网关默认支持 SOFA Registry（SOFA注册中心），此外 API 网关支持添加其他类型的注册中心以应对用户多种操作场景。

> **说明**
>
> 注册中心配置为全局配置，仅管理员角色能修改全局配置。

新增注册中心

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **全局设置** > **注册中心**。
3. 在页面右上角单击 **创建注册中心参数**。
4. 在弹出页面配置注册中心参数。

   注册中心类型有 3 种：SOFA、ZK 和 NACOS，不同的类型配置方式不同，具体差异如下：

   - SOFA

     需要填写注册中心名称和 IP 地址信息。

   - ZK

除了名称和 IP 必填外，用户可自定义配置注册协议类型和注册中心分组。

> **说明**
>
> 注册协议有两种填写方式，默认为空。
>
> - 填写 DAP，注册标识为 DAP 时，mosng 就会改成浙江农信专用的 DAP 协议方式，可以选择协议版本。
> - 为空，注册标识为空时，mosng 为普通的 DUBBO 协议。

- NACOS

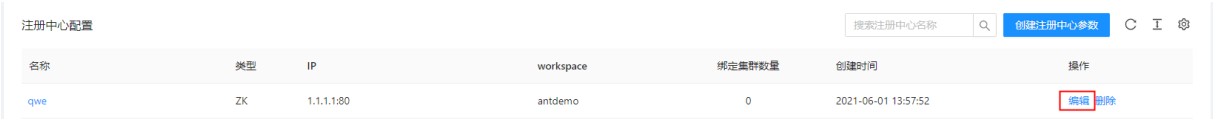
  除了名称和 IP 必填外，用户可自定义配置命名空间 ID、注册中心分组和集群名称。

5. 完成单击 **确定**，完成注册中心的创建。

> **说明**
>
> 注册中心创建完成后，信息全局可见，您在配置系统集群信息时可以选择合适的注册中心。

编辑注册中心

在注册中心页面，单击注册中心列表右侧 **编辑** 按钮，即可对注册中心信息进行修改。

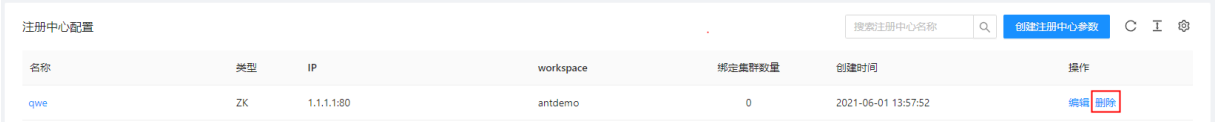


说明

- 编辑注册中心时，信息配置同 新增注册中心。
- 当存在已发布的 API 使用某注册中心寻址时，该注册中心无法进行编辑操作，请修改接口的后端集群寻址方式后进行编辑操作。

删除注册中心

在注册中心页面，单击注册中心列表右侧 删除 按钮，即可删除注册中心。



说明

当存在使用某个注册中心寻址的系统集群时，该注册中心无法进行删除操作，请修改集群的寻址方式后删除。

2.订阅 API

2.1. 创建应用

在调用 API 前，订阅者需在订阅端创建一个应用。这个应用就是一个订阅对像，API 发布者给该应用授权后，便可将该应用和 API 关联在一起。

操作步骤

1. 登录 API 网关控制台。
2. 在左侧导航栏中单击 **API 订阅** > **应用管理**，进入应用列表。
3. 单击应用列表右上方的 **创建应用**。
4. 在弹出的 **创建应用** 窗口中，选择或输入以下相关信息：
   - **应用类型**：
     - **应用**：表示订阅的应用运行在 Web 浏览器、H5 容器、小程序、JS、业务系统中。
     - **mPaaS 移动应用**：（仅专有云）表示订阅的应用运行在 mPaaS 移动客户端中，支持 iOS、Andrord，需要和 mPaas 移动开发平台一起使用。
   - **应用名称**：输入应用的名称，用于识别应用。
   - **描述**：选填，用于描述应用的作用等，64 个字符以内。
   - **是否选择加密**：选择在应用调用 API 时是否对数据内容进行加密。详情请参见 网关数据加密。

     > **说明**
     >
     > 目前仅 HTTP 类型的 API 支持数据加密。

5. 单击 **确定**，即可完成应用的创建。

应用创建完成后，订阅者需在应用详情页获取该应用的 ID（即 APPID）。随后，将 APPID 提供给想要订阅的 API 的发布者，获得该 API 的访问授权。

2.2. 数据加密

统一网关提供了对客户端与网关之间传输的数据进行加密的能力，支持 RSA、ECC、国密三种加密算法，保证数据传输过程中的安全性。

> **说明**
>
> 目前仅支持对 HTTP 类型的 API 调用进行数据加密。

使用流程

1. 根据加密算法，生成密钥。
2. 网关管控平台创建 App 并开启加密能力。
3. 客户端编码配置密钥（当前仅支持 Java 语言）。
4. 发起调用进行验证。

前提条件

安装 OpenSSL 1.1.1 或更新版本。

步骤一：生成密钥

使用 OpenSSL 工具生成 RSA、ECC 或国密算法的密钥，并记录保存生成的密钥。

生成 RSA 密钥

1. 生成 RSA 私钥。

   > **说明**
   >
   > 可选择 1024 或 2048 密钥长度。

   ```
   openssl genpkey -algorithm RSA -out private_key.pem -pkeyopt rsa_keygen_bits:2048
   ```

2. 根据 RSA 私钥生成 RSA 公钥。

   ```
   openssl rsa -pubout -in private_key.pem -out public_key.pem
   ```

生成 ECC 密钥

1. 生成 ECC 的私钥。

   > **说明**
   >
   > 必须选择 secp256k1 曲线。

   ```
   openssl ecparam -name secp256k1 -genkey -noout -out secp256k1-key.pem
   ```

2. 根据 `secp256k1-key.pem` 密钥对生成 ECC 公钥。

   ```
   openssl ec -in secp256k1-key.pem -pubout -out ecpubkey.pem
   ```

生成国密密钥

1. 生成 SM2 的私钥。

```
openssl ecparam -name SM2 -genkey -noout -out sm2-key.pem
```

2. 根据 SM2 私钥生成 SM2 公钥。

```
openssl ec -in sm2-key.pem -pubout -out sm2pubkey.pem
```

步骤二：配置应用加密

1. 创建应用。
2. 在 **数据加密** 页签，开启加密开关，选择加密算法，填写密钥。

> 说明
>
> 格式必须携带 `-----BEGIN PUBLIC KEY-----` 及 `-----END PUBLIC KEY-----` 。

对应用开启加密后，默认该 App 可访问的所有 API 都会进行加解密，如果部分 API 不需要加解密，可以在 API 列表关闭。

步骤三：配置客户端密钥

配置客户端加密，必须升级 SDK 到 2.4.0 版本及以上。代码示例如下：

说明

- 当前只支持 java-sdk。
- 公钥是一个字符串，请保留密钥的换行符。
- 只有创建时 apiClient 传递了加密算法和公钥，并且 `request.NeedEncrypt` 为 true 时，才会进行加解密。任意条件不满足都不会进行加解密。
- HTTP 应用类型的请求，必须签名生成 `SignUtil.signWithUriDate`，但不要求验签。

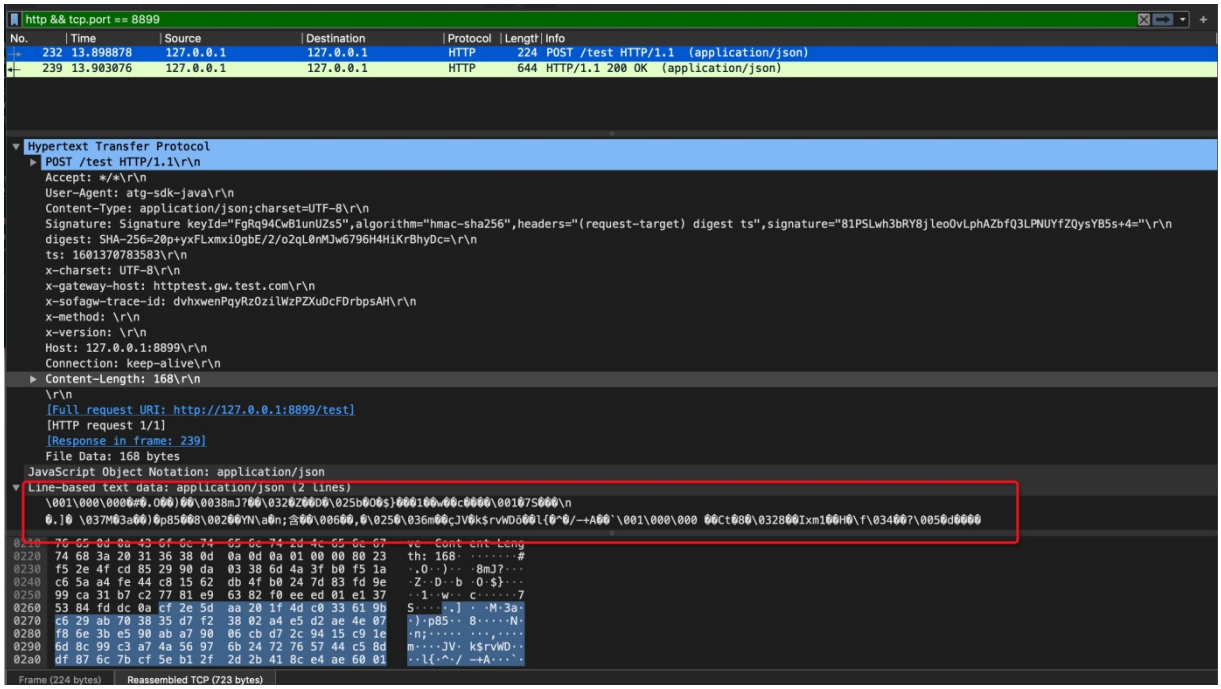
```
String subAppAccessKey ="Z9c**********";
String subAppSecretKey ="3ayD********************";
String pubKey ="-----BEGIN PUBLIC KEY-----\n"+
"MFYwEAYHKoZIzj0CAQYFK4EEAAoDQgAEN4TGDN+1Dkd5Q91ZA3lCqU1sMWB8qLF/VFhMknQ3bcVI4G1W8unh9labvM
hRtEugtfV7HMcQfsVNfPYP4U5BlA==\n"+
"-----END PUBLIC KEY-----";
String gatewayUrl ="http://127.0.0.1:8080";
String path ="/xxxxx/test/post";
String methodType ="POST";
String body ="{}";//post的请求内容

ApiSecretKey apiSecretKey =newApiSecretKey(subAppAccessKey, subAppSecretKey);
List<ApiSecretKey> secretKeys =newArrayList<>();
secretKeys.add(apiSecretKey);


DefaultSdkApiRequest req =newDefaultSdkApiRequest();
req.setPath(path);
req.setClientCheckSign(true);
req.setRequestType(methodType);
req.setNeedEncrypt(true);

//填充签名
SignResponse<String,String>  signResponse =SignUtil.signWithUriDate(
newSignRequest(path, methodType, body.getBytes(), subAppAccessKey, subAppSecretKey),false);
for(String k : signResponse.keySet()){
    req.getHeaderParams().put(k, signResponse.get(k));
}

//需要加密
ApiClient apiClient =newDefaultApiClient(gatewayUrl, secretKeys,EncryptEnum.ECC, pubKey);
//apiClient = new DefaultApiClient(gatewayUrl, secretKeys);
```

步骤四：发起请求并验证

apiClient 配置完成之后，即可对 API 发起调用。代码示例如下：

```
@Test
publicvoid testNoSignHttp(){
ParamPostRequest request =newParamPostRequest();
        request.setPath("simple/demo");
// 是否对响应进行签名校验
        request.setClientCheckSign(false);
ApiResponse response = apiClient.execute(request);
System.out.println(JSON.toJSONString(response));
}
```

如需对加密功能进行验证，可以在统一网关打开 debug 日志，在 `mosng-debug.log` 查看 body 是否加解密成功。下图所示为加密后的效果。



2.3. 管理应用

在 **应用管理** 页面，您可以根据业务需要，查看并管理已创建的应用。

编辑应用信息

1. 打开应用编辑界面有以下两种方式：
   - 在应用列表页，单击应用右侧 **操作** 列中的 **编辑** 即可打开编辑应用界面。
   - 单击应用名进入应用详情页后，单击页面右上方的 **编辑** 即可打开编辑应用界面。

2. 打开的 **编辑应用** 界面与创建应用界面内容一致，您可修改应用的 **应用类型**、**应用名称** 以及 **描述**。

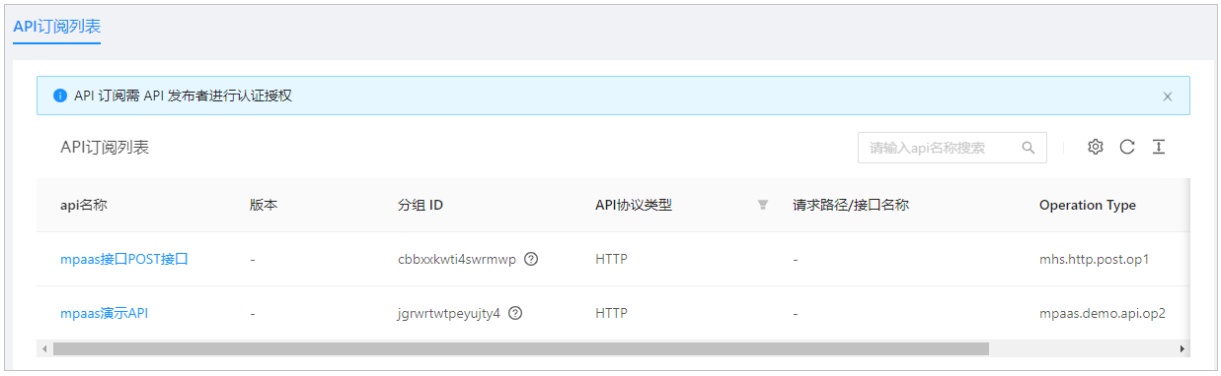
3. 单击 **确定**，即可完成应用信息的修改。

查看应用详情

在应用列表页，单击应用名可进入应用详情页。

API 订阅列表

在应用详情页的 **API 订阅列表** 标签页中，可查看当前应用已订阅的 API。



在 API 订阅列表中，您可以：

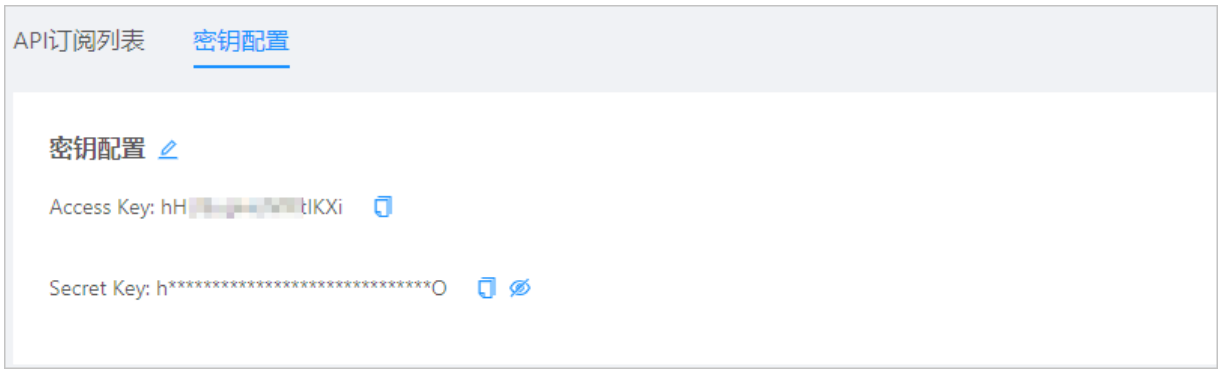
- **查看 API 详情**：单击 API 名称，即可跳转至该 API 的详情页，您可查看该 API 的 **定义**、**请求参数**、**响应示例**、**错误码** 等详细信息。

- **筛选 API 协议类型**：
  i. 单击 **API 协议类型** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择对应类型的 API 协议并单击 **确定** 即可进行筛选。
- **筛选 API 状态**：
  i. 单击 **状态** 右侧的筛选按钮。
  ii. 在打开的筛选框中，选择 **已发布** 或 **未发布**，并单击 **确定** 即可进行筛选。
- **搜索 API**：在 API 列表右上方的搜索框中，可通过 API 名称搜索已订阅的 API。
- **更改展示设置**：
  - **更改列设置**：在 API 列表右上方的功能区域，单击 **列设置** 按钮。在打开的列展示框中，您可勾选或取消 API 列表的展示项目，也可将某一项固定在左/右侧，或拖动展示项进行排序。
  - **调整列密度**：在 API 列表右上方的功能区域，单击 **密度** 按钮。在打开的密度选择框中，可调整 API 列表的展示密度。

密钥配置

订阅 API 的应用访问授权的 API 时，需使用密钥进行身份认证。

在应用详情页的 **密钥配置** 标签页中，您可查看并复制当前应用的 **Access Key** 和 **Secret Key**。



删除应用

在应用列表页，单击应用右侧 **操作** 列中的 **删除**，在确认框中单击 **确定** 即可删除当前应用。

> **说明**
>
> 如果当前应用已订阅 API，则无法被删除，需前往 API 详情页 **解除绑定** 后再删除应用。

3.最佳实践

3.1. 双网关配置

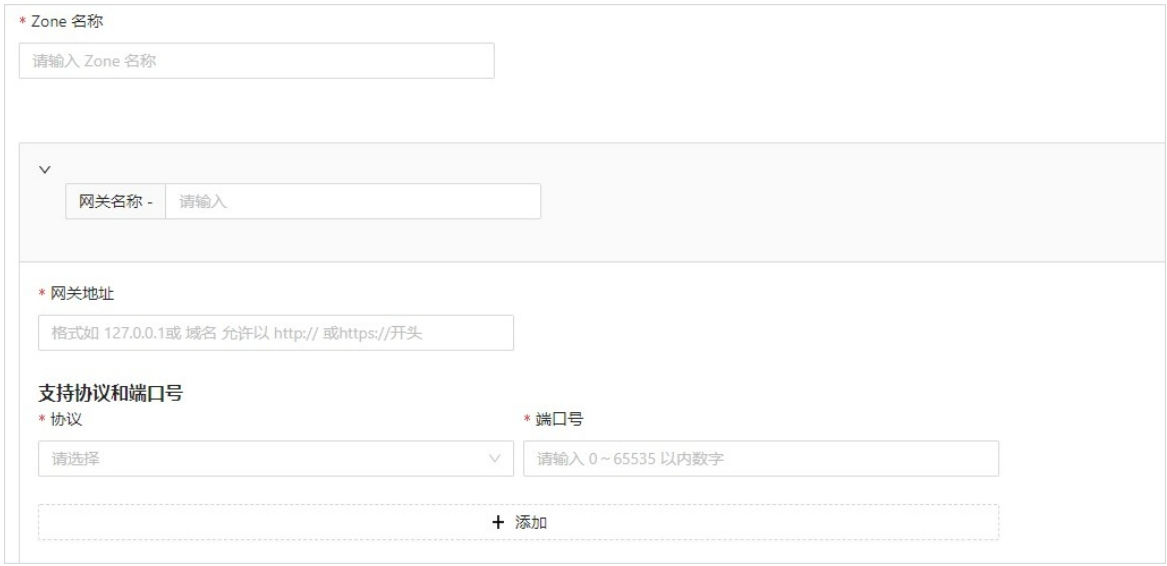
通过在两个网络隔离的区域各部署一个网关，可以实现两个原本网络隔离区域之间的网络互通。本文主要介绍在 API 网关中如何进行双网关跨域设置。

步骤一：跨域设置

1. 登录 API 网关控制台。
2. 在左侧导航栏单击 **跨域设置**，进入全局设置页面。

> 说明
>
> - 仅 **双网关跨域通信场景** 下需要维护全局的机房（网络区域）信息，如每个机房或者网络区的网关地址、注册中心的地址等，以保证网络的正常通信。
> - 该配置为全局配置，仅管理员角色可以修改全局配置，配置完成后在其他工作区（workspace）中仍然生效。

3. 在页面右上角单击 **编辑** > **添加 Zone**，配置以下信息：



- **Zone 名称**：必填，输入添加 Zone 的名称信息。
- **网关名称**：必填，输入添加网关的名称信息。
- **网关地址**：必填，输入网关地址信息。
- **协议**：必填，选择网关协议类型。
- **端口号**：必填，网关端口号信息，仅支持 0~65535 内的数字字符。

  如果您想在一个 workspace 内添加多个网关，单击 **添加网关**，输入相应信息后保存即可。

4. 完成后单击 **提交**，完成一个 workspace 的设置。

> **说明**
>
> 至少配置两个 workspace 才能进行双网关跨域设置的操作。

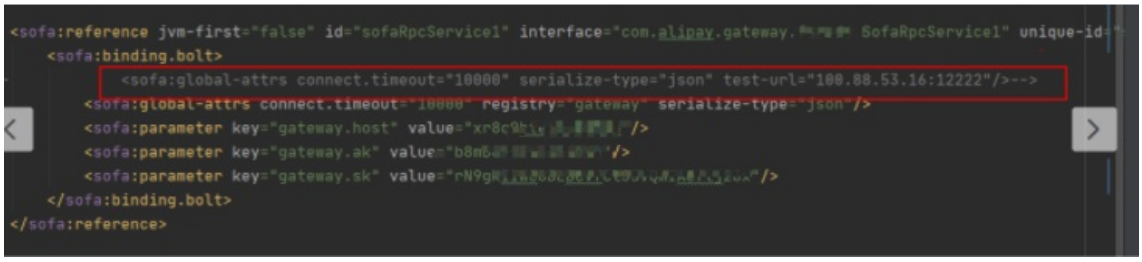
步骤二：配置双网关 API

通过在两个网络隔离的区域各部署一个网关，可以实现两个原本网络隔离区域之间的网络互通。

1. 进入 API 网关控制台页面，在左侧导航栏中选择 **API 发布 > API 管理**。
2. 在 API 列表页，单击列表右上方的 **创建 API**。
3. 在新页面中，选择 **双网关跨云 API** 类型，单击 **创建**。
4. 在 **API 定义** 步骤中，您需要配置以下信息：
   - **所属分组**：必选，用于将 API 进行逻辑的分组，分组下的 API 相同的分组 ID 进行隔离。
   - **API 名称**：必填，用于识别 API，支持英文字母、中文、数字、下划线（_）、连接符（-），32 个字符以内。
   - **描述**：选填，用于描述 API 的作用等，64 个字符以内。
   - **出口网关协议类型**：必选，是调用端发起调用的协议类型。支持 HTTP 和 SOFARPC 两种类型。
   - **入口网关协议类型**：必选，是后端系统集群使用的协议类型。支持 HTTP 和 SOFARPC 两种类型。

   > **说明**
   >
   > 入口网关协议类型种类个数由出口网关协议类型决定。
   >
   > - 出口网关协议类型为 HTTP 时，入口网关协议类型支持 HTTP、SOFARPC 和 SOFAREST 三种类型。
   > - 出口网关协议类型为 SOFARPC 时，入口网关协议类型支持 HTTP、SOFARPC 两种类型。

   当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，网关需要采取直接连接的模式，如下图：



> **说明**
>
> `test-url` =网关地址:12222

- **API 授权应用类型**：必选，指定可以订阅并调用该 API 的应用类型。
  - 应用以及 mPaaS 移动应用：（仅专有云）表示 API 发布后支持 Web 浏览器、H5 容器、小程序、JS、业务系统的调用，同时也支持 mPaaS 移动客户端（iOS、Android）的调用。
  - mPaaS 移动应用：（仅专有云）表示 API 发布后支持 mPaaS 移动客户端（iOS、Android）的调用。
  - 应用：表示 API 发布后支持 web 浏览器、H5 容器、小程序、JS、业务系统的调用。
- **OperationType**：当授权应用类型包含 mPaaS 移动应用时，必填，是针对 mPaaS 移动应用设置的 API 服务标识，用于定位要访问的资源。
  - 支持英文字母、数字、英文句号（.），200 字符以内。
  - 需按照 `com.xxx.opt1` 的格式输入。
- **请求路径**：必填，是针对应用设置的请求资源的 URL，通过请求路径可以定位到要请求的资源。
  - 支持英文字母、数字、下划线（_）、连接符（-），200 字符以内。
  - 以斜杠 / 开头，不能以斜杠 / 结尾，格式示例： `/getUserInfo` 。
- **路径匹配规则**：如订阅并调用该 API 的应用类型为 **应用** 时，必填。
  - 绝对匹配：表示只有请求的路径和配置的路径完全一样时才能匹配成功。
  - 前缀匹配：如希望带有某个前缀的请求全部匹配到某一个 API，则可以使用前缀匹配。
- **接口名称**：出口网关协议类型为 SOFARPC时，必填，是服务的接口类型，支持英文字母、数字、英文句号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。
- **服务标识**：出口网关协议类型为 SOFARPC时，选填，支持英文字母、数字、英文句号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。

  > **说明**
  >
  > 若服务标识不为空，接口名称中优先使用服务标识作为 uniqueId。

- **方法名**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必填，服务方法名，以小写英文字母开头，格式示例：getUserID。
- **方法**：当出口网关类型为 HTTP 时，必填，表明要对给定的 HTTP 资源执行的操作：
  - **GET**：检索服务器中的数据。
  - **POST**：将数据发送到服务器进行处理。
  - **PUT**：更新服务器中存储的数据。
  - **DELETE**：删除服务器中的数据。
- **报文类型**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必选，仅支持 appliction/json。
- **报文编码**：当出口网关类型为 SOFARPC，入口网关类型为 HTTP 时，必选，仅支持 UTF-8。
- **出口网关和入口网关**：必选，选择在 跨域设置 中配置好的网关信息。

> **说明**
>
> 注册中心类型仅支持 NONE。

- **安全认证**：必选，根据实际需要自行选择。
  - 密钥认证：表示订阅应用访问授权的 API 时使用密钥进行身份认证。
  - 无需认证：表示订阅应用访问授权的 API 时不需要身份认证。
- **请求参数**：可选，请参见 请求参数。
- **响应参数**：可选，请参见API 属性说明响应参数。

> **说明**
>
> 当入口网关类型和出口网关类型都为 SOFARPC 时，不需要进行后端配置。

5. 单击 **下一步**，进入 **后端配置** 页面，完成后端配置信息。
   - **后端服务类型**：无法编辑，默认为系统集群。
   - **协议类型**：无法编辑，为上一步中入口网关协议类型。
   - **接口名称**：入口网关协议类型为 SOFARPC 或 SOFAREST 时，必填，是服务的接口类型，支持英文字母、数字、英文句号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。
   - **服务标识**：入口网关协议类型为 SOFARPC 或 SOFAREST 时，选填，支持英文字母、数字、英文句号（.）、冒号（:）、@、下划线（_）、连接符（-），以小写字母作为开头。

     > **说明**
     >
     > 若服务标识不为空，接口名称中优先使用服务标识作为 uniqueId。

   - **方法名**：当入口网关类型为 SOFARPC，必填，服务方法名，以小写英文字母开头，格式示例：getUserID。
   - **请求路径**：请求到后端时显示的路径，必须以斜杠 / 开头，不能以斜杠 / 结尾。
   - **超时时间**：必填，API 请求超时时间，单位为毫秒（ms）。
   - **路由策略**：必选，表示当网关接收到请求后使用的路由策略。详见 创建路由规则。
     - 根据 Header 路由：根据 Header 里带的参数路由。
     - 根据权重路由：数值范围为 0~100，所有规则加起来为 100。
     - 根据请求路径/接口路由：即直接转发。
     - 根据 LDC 路由：即 LDC 单元化路由转发，仅支持协议类型为 SOFARPC 或 SOFAREST 时可选。

       > **说明**
       >
       > 使用单元化路由转发前，需要在单元化应用服务中创建统一接入实例，具体内容请参见 创建统一接入实例。

   - **系统集群**：必选，选择当前环境内已配置好的系统集群，支持选择跨 VPC 集群。您也可以单击 **创建系统集群** 直接新建一个系统集群。详见 创建系统集群。

- **响应参数映射自定义配置**：可选，支持从下拉菜单中选择现有的响应参数映射规则，或单击 **新建参数映射规则** 直接创建一个响应参数映射规则。更详细的创建步骤，参见 创建参数映射。
- **请求参数映射配置**：可选，支持选择 **参数名映射** 或 **自定义配置**。选择 **自定义配置**，即可使用现有的请求参数映射规则。

6. 完成后，单击 **创建**，完成双网关跨云 API 的创建。

注意事项

订阅方接入 SDK 使用

- `request.getHeaderParams().put("x-gateway-host","5jzkcn0idt3w4czs.apigateway.inc.alipay.net")`

  。

  > **说明**
  >
  > `x-gateway-host` 需要填写出口网关的网关地址。

- `apiClient=newDefaultApiClient(gatewayUrl, secretKeys)` 。

  > **说明**
  >
  > `gatewayUrl` 需要填写出口网关地址和对应端口的 SLB IP 信息。
  >
  > - SLB 默认端口为 8080 和 12222。
  > - Pod 端口为 8088 和 12222。

项目启动注意事项

- 需保证云上云下网关网络互通。
- 云上云下双网关需要启动两个网关服务：**云下跨域网关数据面** 和 **云上跨域网关数据面**，由于跨域网关从控制台拉取数据，所以控制台需要 SLB 提供一个 9876 GRPC 连接端口以供数据面建立连接，您需要填写的参数如下：
  - **GATEWAY_TYPE="CROSS"**，代表跨域网关数据面，不填默认标准网关。（后续会优化只需要填写 MOSNG_ZONE 即可）
  - **MOSNG_ZONE**，代表用户自定义数据面所在 Zone 信息。
  - **GATEWAY_CONSOLE_ENDPOINT**，代表控制台建立的链接信息。

> **说明**
>
> 如果不使用普通网关数据面，必须配置以上三个启动参数。

3.2. 代码生成

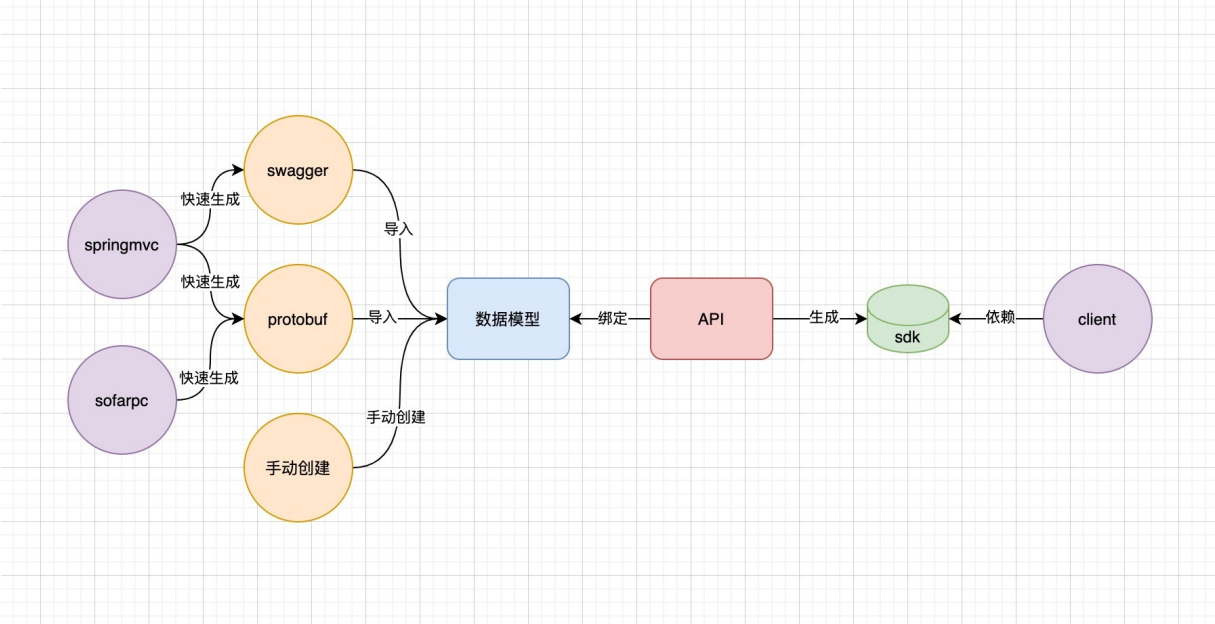
3.2.1. 总览

API 绑定数据模型后，网关支持生成代码的操作，以方便使用 SDK 发起调用。

> **说明**
>
> 目前生成代码功能只支持 Java 类型的 SDK。

使用方式

1. 在网关管控平台通过多种方式创建数据模型。
2. API 绑定 request 和 response 对应的数据模型。
3. 在 API 分组下生成 SDK。
4. 客户端引入生成的 SDK 发起调用。



快速创建数据模型

数据模型可以在管控平台自行创建，但是效率较低。为此我们提供了多种方式可以快速生成数据模型。

- **Proto 文件导入**

  可以使用 Protobuf 来描述数据模型，网关会为每一个 message 生成对应的数据模型。为了方便操作，网关提供了快速生成 Proto 文件的 SDK，您只需简单的配置，即可生成 Proto 文件并导入网关。详情请参见：SOFARPC 生成 Proto 文件。

- **Swagger 文件导入**

  网关支持标准 Swagger 格式的文件，可以使用任意开源组件生成 Swagger 文档并导入网关。

说明

网关只会解析 Swagger 文档中的 model 部分。

附录

代码生成不支持的数据类型

- 容器类型不能多层嵌套。
- List 或 Map 必须有泛型信息。
- List 或 Map 的泛型信息不能是 Array 类型。
- 不支持单字节（字节数据 `byte[]` 是支持的。）
- 不支持对象数组，请用 List 代替。
- 属性名不能是 `data` 和 `description`，会与 iOS 的属性冲突。
- Map 类型的 key 必须是 String 类型。
- Map 类型的 value 不能是 Object 类型。
- 类型不能是抽象类。
- 类型不能是接口类。

错误的写法如下所示：

```
public class Req {
    private Map<String,List<Person>> map; //容器类型不能多层嵌套。
    private List<Map<Person>> list; //容器类型不能多层嵌套。
    private List list1; //List 或 Map 必须有泛型信息。
    private Map map1; //List 或 Map 必须有泛型信息。
    private List<Person[]> listArray; //List 或 Map 的泛型信息不能是 Array 类型。
    private byte b; //不能为单字节。
    private Person[] personArray; //不支持对象数组，请用 List 代替。
    private String description; //属性名不能 description。
}
```

代码生成支持的数据类型

```
boolean, double, float, int, long, short
java.lang.Boolean
java.lang.Double
java.lang.Float
java.lang.Integer
java.lang.Long
java.lang.Short
java.lang.String
java.util.List,但：必须使用类型参数；不能使用其具体子类，以下简称 List。
java.util.Map,但：必须使用类型参数；不能使用其具体子类；key 类型必须是 String 以下简称 Map。
Enum
byte[]
```

正确的写法如下所示：

```
public class Req {
    private String s = "ss";
    private int i;
    private double d;
    private Long l;
    private long l1;
    private boolean b;
    private List<String> stringList;
    private List<Person> personList;
    private Map<String,Person> map;
    private byte[] bytes;
    private EnumType type;
}

public class Person {
    private String name;
    private int age;
}
```

3.2.2. SOFARPC 生成 Proto 文件

步骤一：引入近端包

```
        <dependency>
            <groupId>com.alipay.mosng</groupId>
                                    <artifactId>server-codegen-sofarpc</artifactId>
            <version>1.0.0</version>
        </dependency>
```

步骤二：对需要生成代码的服务打注解

```
public interface SofaRpcService {

    String hello();

    @CodeGen
    User hello1(String param);

    @CodeGen
    String hello2(String param, String param1, String param2);
}
```

> **说明**
>
> 只有打了注解的 method，才会生成相应的 Req 和 Resp 数据模型。

多参数问题

对于多参数的 method，由于网关只能绑定一个 Request，所以会将多个参数打包为一个新数据模型。比如：

```
@CodeGen
String hello2(String param, String param1, String param2);
```

会生成如下 Proto：

```
## Hello2Req.proto

syntax = "proto3";

option java_package = "com.alipay.gateway.facade";
package com_alipay_gateway_facade;

import "google/protobuf/descriptor.proto";

extend google.protobuf.MessageOptions {
   bool mosng_expand = 50001;
}

message Hello2Req {
  option (mosng_expand) = true;

  string param = 1;
  string param1 = 2;
  string param2 = 3;
}
```

为了可以取到参数的 name，要求 JDK 版本为 1.8 及以上，并且 Javac 参数加上 -parameters，maven 项目可以参考：

```
<plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <configuration>
                <source>1.8</source>
                <target>1.8</target>
                <compilerArgument>-parameters</compilerArgument>
                <encoding>UTF-8</encoding>
        </configuration>
        <version>3.5.1</version>
</plugin>
```

集合类

如果返回值是 List 或者 Map，要求 Map 的 key 必须是 String 类型。

如果 List 的泛型和 Map 的 value 是一个 Object，网关只会为 Object 生成 Proto 文件。 List 和 Map 需要在管控平台进行设置。

请求 Body 类型

Map | com_alipay_gatewa...

响应参数

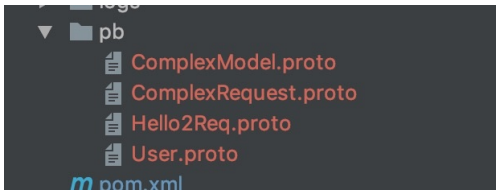
响应 Body 类型

List | Float

步骤三：配置代码生成服务

```
    <bean id="sofaRpcCodegenService" class="com.alipay.apigateway.codegen.SofaRpcCodegenSer
vice" >
        <property name="localDir" value="./pb"/>
    </bean>
```

localDir 是存放生成的 Proto 文件的地址。

步骤四：获取 Proto 文件

本地启动项目，即可在 localDir 下找到生成的 Proto 文件，直接将 Proto 文件导入网关即可生成网关数据模型。