

# 服务网格 CSM



2023-10-18

# 目录

1	文档简介 .....	7
1.1	产品说明 .....	7
1.2	名词解释 .....	7
1.3	接口工具测试 .....	7
2	使用流程 .....	7
3	全局参数列表 .....	8
3.1	全局参数 .....	8
4	目录参数列表 .....	错误! 未定义书签。
4.1	应用管理 .....	错误! 未定义书签。
4.2	应用分组 .....	错误! 未定义书签。
4.3	应用发布实例运维 .....	错误! 未定义书签。
4.4	环境管理 .....	错误! 未定义书签。
4.5	通用应用镜像部署配置管理 .....	错误! 未定义书签。
4.6	流水线管理 .....	错误! 未定义书签。
4.7	制品库管理 .....	错误! 未定义书签。
4.8	项目管理 .....	错误! 未定义书签。
4.9	服务连接管理 .....	错误! 未定义书签。
4.10	K8s 配置项管理 .....	错误! 未定义书签。
4.11	数据权限管理 .....	错误! 未定义书签。
4.12	流水线成员权限管理 .....	错误! 未定义书签。
4.13	成员管理 .....	错误! 未定义书签。
5	接口列表 .....	8
5.1	应用管理 .....	8
5.1.1	新建通用类型应用 .....	错误! 未定义书签。
5.1.2	删除应用 .....	错误! 未定义书签。
5.1.3	分页获取应用列表 .....	错误! 未定义书签。
5.1.4	给应用设置分组 .....	错误! 未定义书签。
5.1.5	编辑通用类型应用信息 .....	错误! 未定义书签。
5.1.6	新建微服务应用 .....	错误! 未定义书签。

5.1.7	获取编译环境信息列表 .....	错误! 未定义书签。
5.1.8	获取运行环境信息列表 .....	错误! 未定义书签。
5.2	应用分组 .....	错误! 未定义书签。
5.2.1	删除应用分组 .....	错误! 未定义书签。
5.2.2	新建应用分组 .....	错误! 未定义书签。
5.2.3	获取应用分组列表 .....	错误! 未定义书签。
5.2.4	更新应用分组 .....	错误! 未定义书签。
5.3	应用发布实例运维 .....	错误! 未定义书签。
5.3.1	新建 K8s 服务 Service .....	错误! 未定义书签。
5.3.2	删除 K8s 应用 .....	错误! 未定义书签。
5.3.3	删除 K8s 服务 Service .....	错误! 未定义书签。
5.3.4	获取 K8s 应用的 Pod 信息 .....	错误! 未定义书签。
5.3.5	获取 K8s 应用的 Deployment 信息 .....	错误! 未定义书签。
5.3.6	查看变更流程详情 .....	错误! 未定义书签。
5.3.7	获取部署应用详情 .....	错误! 未定义书签。
5.3.8	查询部署应用状态 .....	错误! 未定义书签。
5.3.9	获取 K8s 服务 Service 列表 .....	错误! 未定义书签。
5.3.10	获取已部署应用列表 .....	错误! 未定义书签。
5.3.11	列出应用历史变更单列表 .....	错误! 未定义书签。
5.3.12	重启 K8s 应用 .....	错误! 未定义书签。
5.3.13	手动扩缩容 K8s 应用 .....	错误! 未定义书签。
5.3.14	启动 K8s 应用 .....	错误! 未定义书签。
5.3.15	停止 K8s 应用 .....	错误! 未定义书签。
5.3.16	更新 K8s 服务 Service .....	错误! 未定义书签。
5.3.17	执行下一批发布 .....	错误! 未定义书签。
5.3.18	部署 K8s 应用 .....	错误! 未定义书签。
5.3.19	获取历史版本信息 .....	错误! 未定义书签。
5.3.20	立即回滚 .....	错误! 未定义书签。
5.3.21	回滚应用 .....	错误! 未定义书签。
5.4	环境管理 .....	错误! 未定义书签。
5.4.1	创建环境 .....	错误! 未定义书签。

5.4.2	创建 K8s 命名空间	错误! 未定义书签。
5.4.3	删除环境	错误! 未定义书签。
5.4.4	删除 K8s 命名空间	错误! 未定义书签。
5.4.5	删除导入环境的资源	错误! 未定义书签。
5.4.6	获取容器服务 K8s 集群列表	错误! 未定义书签。
5.4.7	资源导入环境	错误! 未定义书签。
5.4.8	环境类型列表	错误! 未定义书签。
5.4.9	环境列表	错误! 未定义书签。
5.4.10	查询环境下资源信息	错误! 未定义书签。
5.4.11	获取 K8s 命名空间列表	错误! 未定义书签。
5.4.12	资源池列表	错误! 未定义书签。
5.4.13	更新环境	错误! 未定义书签。
5.4.14	获取注册中心列表	错误! 未定义书签。
5.5	通用应用镜像部署配置管理	错误! 未定义书签。
5.5.1	新建通用应用镜像部署配置	错误! 未定义书签。
5.5.2	删除部署配置	错误! 未定义书签。
5.5.3	获取通用应用的镜像部署配置详情	错误! 未定义书签。
5.5.4	不分页获取应用的部署配置列表,不含配置详情	错误! 未定义书签。
5.5.5	更新部署配置中的环境变量	错误! 未定义书签。
5.5.6	更新部署配置中的 Pod 规格	错误! 未定义书签。
5.6	流水线管理	错误! 未定义书签。
5.6.1	删除流水线	错误! 未定义书签。
5.6.2	分页查询流水线列表	错误! 未定义书签。
5.6.3	中止流水线	错误! 未定义书签。
5.6.4	流水线执行详情	错误! 未定义书签。
5.6.5	流水线失败重试	错误! 未定义书签。
5.6.6	执行流水线	错误! 未定义书签。
5.7	制品库管理	错误! 未定义书签。
5.7.1	通用制品程序包下载	错误! 未定义书签。
5.7.2	查询制品-docker 镜像详情	错误! 未定义书签。
5.7.3	查询制品-通用制品详情	错误! 未定义书签。

5.7.4	查询制品-docker 镜像分页列表.....	错误! 未定义书签。
5.7.5	查询制品-通用制品分页列表.....	错误! 未定义书签。
5.8	项目管理.....	错误! 未定义书签。
5.8.1	创建项目.....	错误! 未定义书签。
5.8.2	删除项目.....	错误! 未定义书签。
5.8.3	分页查询项目列表.....	错误! 未定义书签。
5.8.4	更新项目.....	错误! 未定义书签。
5.9	服务连接管理.....	错误! 未定义书签。
5.9.1	获取指定类型服务连接列表.....	错误! 未定义书签。
5.10	K8s 配置项管理.....	错误! 未定义书签。
5.10.1	创建 K8s 配置项.....	错误! 未定义书签。
5.10.2	删除 K8s 配置项.....	错误! 未定义书签。
5.10.3	获取 K8s 配置项列表.....	错误! 未定义书签。
5.10.4	更新 K8s 配置项.....	错误! 未定义书签。
5.11	数据权限管理.....	错误! 未定义书签。
5.11.1	授予指定应用成员操作权限.....	错误! 未定义书签。
5.11.2	授予指定环境成员操作权限.....	错误! 未定义书签。
5.11.3	获取指定应用成员操作权限列表.....	错误! 未定义书签。
5.11.4	获取指定应用可添加成员权限列表.....	错误! 未定义书签。
5.11.5	获取指定环境成员操作权限列表.....	错误! 未定义书签。
5.11.6	获取指定环境可添加成员权限列表.....	错误! 未定义书签。
5.11.7	移除指定应用成员操作权限.....	错误! 未定义书签。
5.11.8	移除指定环境成员操作权限.....	错误! 未定义书签。
5.12	流水线成员权限管理.....	错误! 未定义书签。
5.12.1	切换流水线权限模式.....	错误! 未定义书签。
5.12.2	删除流水线成员权限.....	错误! 未定义书签。
5.12.3	添加流水线成员权限.....	错误! 未定义书签。
5.12.4	获取流水线查询可添加的成员分页列表.....	错误! 未定义书签。
5.12.5	获取流水线成员权限类型列表.....	错误! 未定义书签。
5.12.6	获取流水线成员权限分页列表.....	错误! 未定义书签。
5.12.7	获取流水线权限模式.....	错误! 未定义书签。

5.12.8	修改流水线成员权限 .....	错误! 未定义书签。
5.13	成员管理 .....	错误! 未定义书签。
5.13.1	授权成员角色 .....	错误! 未定义书签。
5.13.2	获取可添加的子账号用户分页列表 .....	错误! 未定义书签。
5.13.3	获取角色列表 .....	错误! 未定义书签。
5.13.4	获取子账号用户分页列表 .....	错误! 未定义书签。
5.13.5	移除子账号 .....	错误! 未定义书签。
6	附录 .....	61
6.1	返回码列表 .....	61

# 1 文档简介

## 1.1 产品说明

本开发手册对该系统功能接口进行详细的描述，通过该指南可以对本系统有全面的了解，使技术人员尽快掌握本系统的接口，并能够在本系统上进行开发。

## 1.2 名词解释

名词缩写	名词定义
客户端	本文档中的客户端为 ApiPost 客户端。
服务端	本文档中的服务端为用户端。
环境变量	本系统中动态的 KV 参数。

## 1.3 接口工具测试

暂无

## 1.4 请求头参数说明

参数名	示例值	参数类型	是否必填	参数描述
regionId	bb9fdb42056f11ed a1610242ac110002	String	是	资源池 ID, 请参考调用前必知-概述

# 2 使用流程

### 1. 准备阶段：

- 申请测试号等信息；
- 取得开发手册（本文档）等资料；

### 2. 开发阶段：

- 根据提供的 DEMO 结合开发文档快速熟悉对接接口；
- 根据本系统提供的接口，在自己的系统上进行开发，实现所需要的业务功能；
- 对自己系统的业务功能进行全面测试；
- 与测试环境进行联调。

- 3. 生产使用:
  - A. 使用系统提供的正式资料。

## 3 全局参数列表

### 3.1 全局参数

目录描述: 暂无描述

## 4 接口列表

### 4.1 集群管理

- 4.1.1 获取服务网格所在集群的信息列表  
本接口获取服务网格所在集群的信息列表, 包括主集群和从集群。

接口 URL: /csm/v1/open/cluster/list

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
clusters	Array	array	集群列表

列表 clusters

参数名	示例值	参数类型	参数描述
clusterDomain	<b>example.com</b>	String	集群域名
clusterId	<b>kjye*****</b>	String	集群 ID
createdTime	<b>2023-07-24*****</b>	String	集群创建时间
name	<b>Name</b>	String	集群名称
regionId	<b>regionCode</b>	String	集群所在地域 ID
serviceMeshId	<b>e779bd8*****</b>	String	服务网格 ID
sgId	<b>djia54****</b>	String	安全组 ID
state	<b>running</b>	String	集群状态，取值： running：运行中。 starting：启动中。 stopping：终止中。 stopped：已停止。 failed：失败。
total	<b>200</b>	Integer	集群总数
updateTime	<b>2023-07-24*****</b>	String	集群最后修改时间
version	<b>v1.1</b>	String	集群版本号
vpclId	<b>9jf*****</b>	String	专有网络 ID

返回示例：成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "clusters": [{
      "clusterDomain": "example.com",
      "clusterId": "kjye*****",
      "createdTime": "2023-07-24*****",
      "name": "Name",
```

```
    "regionId": "regionCode",
    "serviceMeshId": "e779bd8*****",
    "sgId": "djia54*****",
    "state": "running",
    "total": 200,
    "updateTime": "2023-07-24*****",
    "version": "v1.1",
    "vpId": "9jf*****"
  }
},
"statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.1.2 获取服务网格的可用集群

本接口获取服务网格的可用集群，即该租户资源池内尚未加入网格的集群列表。

接口 URL: /csm/v1/open/cluster/list\_available

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接

			口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
clusters	array	array	集群列表

列表 clusters

参数名	示例值	参数类型	参数描述
clusterDomain	example.com	String	集群域名
clusterId	kjye*****	String	集群 ID
createdTime	2023-07-24*****	String	集群创建时间
name	Name	String	集群名称
regionId	regionCode	String	集群所在地域 ID
serviceMeshId	e779bd8*****	String	服务网格 ID
sgId	djia54****	String	安全组 ID
state	running	String	集群状态，取值： running：运行中。 starting：启动中。 stopping：终止中。 stopped：已停止。 failed：失败。
total	200	Integer	集群总数
updateTime	2023-07-24*****	String	集群最后修改时间
version	v1.1	String	集群版本号
vpId	9jf*****	String	专有网络 ID

返回示例：成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
```

```
"clusters": [{
  "clusterDomain": "example.com",
  "clusterId": "kjye*****",
  "createdTime": "2023-07-24*****",
  "name": "Name",
  "regionId": "regionCode",
  "serviceMeshId": "e779bd8*****",
  "sgId": "djia54****",
  "state": "running",
  "total": 200,
  "updateTime": "2023-07-24*****",
  "version": "v1.1",
  "vpId": "9jf*****"
}]
},
"statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.1.3 获取集群的 Namespace 列表

本接口获取指定集群的 namespace，集群需要从属于对应的服务网格实例。

接口 URL: /csm/v1/open/cluster/show\_namespace

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id
cluster_id	kjye*****	String	是	集群 id
show_labels	false	Boolean	否	是否同时返回命名空间的标签

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
namespaceList	array	Array[NameSpaceInfo]	集群命名空间列表

表 NameSpaceInfo

参数名	示例值	参数类型	参数描述
labels	{"istio-injection": "enabled"}	map[string] string	命名空间标签, 仅当 show_labels 为 true 时有值。
name	default	string	命名空间名称

返回示例: 成功

```
{
  "statusCode": 2000,
  "message": "SUCCESS",
  "code": "CSM_2000",
  "returnObj": {
    "namespaceList": [
      {
        "name": "default",
```

```
"labels": {
  "istio-injection": "enabled",
  "kubernetes.io/metadata.name": "default"
},
{
  "name": "istio-ingress",
  "labels": {
    "kubernetes.io/metadata.name": "istio-ingress"
  }
},
{
  "name": "istio-system",
  "labels": {
    "istio-injection": "enabled",
    "kubernetes.io/metadata.name": "istio-system"
  }
},
{
  "name": "opa-istio",
  "labels": {
    "kubernetes.io/metadata.name": "opa-istio"
  }
},
{
  "name": "test",
  "labels": {
    "istio-injection": "enabled",
    "kubernetes.io/metadata.name": "test"
  }
}
```

```
    ]
  }
}
```

返回示例：失败见 4.10

#### 4.1.4 获取集群的 Pod 列表

本接口获取指定 K8s 集群指定命名空间内的 Pod。

接口 URL: /csm/v1/open/cluster/show\_pods

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id
cluster_id	kjye*****	String	是	集群 id
namespace	default	String	是	命名空间

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
podList	array	Array[string]	pod 名称

返回示例：成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "podList": [
      "views",
      "productpage",
      "ratings"
    ]
  },
  "statusCode": 2000
}
```

返回示例：失败见 4.10

## 4.2 网格管理

### 4.2.1 获取服务网格详情

本接口用于获取服务网格的详情。

接口 URL: /csm/v1/open/servicemesh/detail

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
-----	-----	------	------

code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
endPoints	array	Array[EndPoint]	集群列表
meshInfo	object	ServiceMeshInfo	服务网格信息
meshSpec	object	ServiceMeshSpec	服务网格明细信息

表 ServiceMeshInfo

参数名	示例值	参数类型	参数描述
clusterType	String	String	实例类型 ENTERPRISE 企业版 BASE 基础版
createdTime	String	String	创建时间
name	String	String	服务网格名称
regionCode	String	String	资源池编码
regionName	String	String	资源池名称
serviceMeshId	String	String	服务网格 id
status	String	String	服务网格状态(状态 运行中、未就绪)
version	String	String	服务网格版本

表 ServiceMeshSpec

参数名	示例值	参数类型	参数描述
clusters	Array	Array[SimpleCluster]	实例类型 ENTERPRISE 企业版 BASE 基础版
meshConfig	Object	APIMeshConfig	服务网格配置

network	Object	Network	服务网络的网络信息
---------	--------	---------	-----------

表 SimpleCluster

参数名	示例值	参数类型	参数描述
clusterId	String	String	集群 id
clusterName	String	String	集群名

表 APIMeshConfig

参数名	示例值	参数类型	参数描述
accessLogSettings	Array	Array[AccessLogSetting]	访问日志的配置, 包括控制面和数据面的开关和日志项目
injectorConfig	Object	InjectorConfig	Sidecar 注入相关的配置
istioConfig	Object	IstioConfig	istio 相关的配置, 一般配置在 istio 的 configMap
monitorSetting	Object	MonitorSetting	监控配置
opa	Object	OPASetting	opa 配置

表 AccessLogSetting

参数名	示例值	参数类型	参数描述
enabled	true	Boolean	是否启用访问日志
project	istiod-accesslog-xxxxx	String	访问日志项目名称
type	data-panel	String	配置类型。取值 data-panel: 数据面 control-panel: 控制面

表 InjectorConfig

参数名	示例值	参数类型	参数描述
cniConfig	Object	CNIConfig	CNI 配置
globalConfig	Object	GlobalConfig	全局配置
SidecarInjectorWebhook	Object	SidecarInjectorConfig	自动注入配置

表 CNIConfig

参数名	示例值	参数类型	参数描述
enabled	Boolean	Boolean	是否启用 CNI
excludeNamespaces	Array	Array[string]	排除的命名空间

表 GlobalConfig

参数名	示例值	参数类型	参数描述
proxyConfig	Object	ProxyConfig	Sidecar 容器配置
proxyInitConfig	Object	ProxyInitConfig	istio-init 初始化容器配置

表 ProxyConfig

参数名	示例值	参数类型	参数描述
clusterDomain	cluster.domain	String	信任域
enableCoreDump	false	Boolean	是否开启 coredump
excludeIPRanges	127.0.0.1	String	不拦截指定 IP 范围, 多个 IP 用逗号分隔
excludeInboundPorts	80,8080	String	不拦截指定 Inbound 端口, 多个端口用逗号分隔
excludeOutboundPorts	80,8080	String	不拦截指定 Outbound 端口
includeIPRanges	192.168.0.0/16	String	包含的 IP 地址段
includeInboundPorts	443	String	包含的 Inbound 端

			□
includeOutboundPorts	443	String	包含的 Outbound 端口
logLevel	info	String	日志等级, 如果不设置则为 warning。可取值为: trace,debug,info,warning,error,critical,off
resourceCPULimit	20m	String	Sidecar 容器的 cpu 限制
resourceCPURequest	200Mi	String	Sidecar 容器的 cpu 所需
resourceMemoryLimit	10m	String	Sidecar 容器的内存限制
resourceMemoryRequest	100Mi	String	Sidecar 容器的内存所需

表 ProxyInitConfig

参数名	示例值	参数类型	参数描述
resourceCPULimit	20m	String	init 容器的 cpu 限制
resourceCPURequest	10m	String	init 容器的 cpu 所需
resourceMemoryLimit	200Mi	String	init 容器的内存限制
resourceMemoryRequest	100Mi	String	init 容器的内存所需

表 SidecarInjectorConfig

参数名	示例值	参数类型	参数描述
alwaysInjectSelector	Array	Array	标签选择列表

defaultTemplates	Array	Array	默认 Sidecar 配置
enableNamespacesByDefault	false	Boolean	是否在没有启用 istio-injection 标签的命名空间注入 Sidecar
injectedAnnotations	Object	Object	注入 Sidecar 后添加到 pod 元数据的注解
injectionURL	String	String	webhook 服务器的 URL
neverInjectSelector	Array	Array	指定哪些 pod 不注入 Sidecar
objectSelector	Object	Object	Object 的选择器
rewriteAppHTTPProbe	false	Boolean	是否重写 HTTP 探针
templates	Object	Object	Sidecar 的配置模板
useLegacySelectors	false	Boolean	是否使用旧标签选择器匹配 Pod

表 IstioConfig

参数名	示例值	参数类型	参数描述
accessLogEncoding	TEXT	String	访问日志的编码, 目前支持 TEXT 和 JSON
accessLogFile	/dev/stdout	String	访问日志的路径
accessLogFormat	[%START_TIME%] \"%REQ(:METHOD)% %REQ(X-ENVOY-ORIGINAL-PATH?:PATH)% %PROTOCOL%\" %RESPONSE_CODE% %RESPONSE_FLAGS% %RESPONSE_CODE_DET	String	访问日志的格式

	<pre> AILS% %CONNECTION_TERMINATION_DETAILS%\ "%UPSTREAM_TRANSPORT_FAILURE_REASON%\ " %BYTES_RECEIVED% %BYTES_SENT% %DURATION% %RESP(X-ENVOY-UPSTREAM-SERVICE-TIME)% \ "%REQ(X-FORWARDED-FOR)%\ " \ "%REQ(USER-AGENT)%\ " \ "%REQ(X-REQUEST-ID)%\ " \ "%REQ(:AUTHORITY)%\ " \ "%UPSTREAM_HOST%\ " %UPSTREAM_CLUSTER% %UPSTREAM_LOCAL_ADDRESS% %DOWNSTREAM_LOCAL_ADDRESS% %DOWNSTREAM_REMOTE_ADDRESS% %REQUESTED_SERVER_NAME% %ROUTE_NAME%\n </pre>		
defaultConfig	Object	DefaultConfig	istio 默认配置
outboundTrafficPolicy	ALLOW_ANY	String	<p>出向流量策略。取值：ALLOW_ANY：允许访问所有外部服务。</p> <p>REGISTRY_ONLY：只能访问注册到网格内的服务。</p>
telemetry	false	Boolean	<p>是否开启采集 Prometheus 监控指标</p>

表 DefaultConfig

参数名	示例值	参数类型	参数描述
concurrency	2	Integer	工作线程数
enableDnsCapture	false	Boolean	是否开启 istio 内置 dns
holdApplicationUntilProxyStarts	true	Boolean	应用程序启动前是否等待 istio-proxy 启动完成
interceptionMode	REDIRECT	String	Sidecar 对入向流量的拦截策略， REDIRECT: 默认的拦截策略，Sidecar 将使用重定向方式拦截入向流量。 TPROXY: 透明代理拦截策略，Sidecar 将以透明代理的方式拦截入向流量。 NONE: 不拦截流量
proxyMetadata	Object	map[string]string	环境变量
terminationDrainDuration	5s	String	Istio Proxy 终止等待时长

表 MonitorSetting

参数名	示例值	参数类型	参数描述
enableMetrics	true	Boolean	是否开启监控指标
traceSampling	1	Integer	trace 采样率[0-100]

表 opa

参数名	示例值	参数类型	参数描述
enabled	true	Boolean	是否开启 OPA 注入
logLevel	error	String	OPA 代理容器日志级别

表 Network

参数名	示例值	参数类型	参数描述
SecurityGroupId	Array	Array	安全组 ID
vpcId	Object	Object	专有网络 ID

表 EndPoint

参数名	示例值	参数类型	参数描述
IntranetPilotEndpoint	192.168.xx.xx:15012	String	Istio Pilot 地址(内网)
IntranetApiServerEndpoint	https://192.168.xx.xx:6443	String	API Server 地址(内网)

返回示例：成功

```

{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "endPoints": [
      {
        "intranetApiServerEndpoint": "https://192.168.xx.xx:6443",
        "intranetPilotEndpoint": "192.168.xx.xx:15012"
      }
    ],
    "meshInfo": {

```

```
"clusterType": "ENTERPRISE",
"createdTime": "2023-07-18T10:21:20+08:00",
"name": "Name",
"regionCode": "d8bbd132b5****",
"regionName": "内蒙 6",
"serviceMeshId": "e779bd8****",
"status": "READY",
"version": "1.16"
},
"meshSpec": {
  "clusters": [
    {
      "clusterId": "efdsafe****",
      "clusterName": "ccse-****"
    }
  ],
  "meshConfig": {
    "accessLogSettings": [
      {
        "enabled": true,
        "project": "istiod-accesslog-xxxxx",
        "type": "data-panel"
      }
    ],
    "injectorConfig": {
      "cniConfig": {
        "enabled": false,
        "excludeNamespaces": [
          "kube-system",
          "istio-system"
        ]
      }
    }
  }
}
```

```
},
"globalConfig": {
  "proxyConfig": {
    "clusterDomain": "cluster.domain",
    "enableCoreDump": false,
    "excludeIPRanges": "127.0.0.1",
    "excludeInboundPorts": "80,8080",
    "excludeOutboundPorts": "80,8080",
    "includeIPRanges": "192.168.0.0/16",
    "includeInboundPorts": "443",
    "includeOutboundPorts": "443",
    "logLevel": "trace",
    "resourceCPULimit": "20m",
    "resourceCPURequest": "10m",
    "resourceMemoryLimit": "200Mi",
    "resourceMemoryRequest": "100Mi"
  },
  "proxyInitConfig": {
    "resourceCPULimit": "20m",
    "resourceCPURequest": "10m",
    "resourceMemoryLimit": "200Mi",
    "resourceMemoryRequest": "100Mi"
  }
},
"SidecarInjectorWebhook": {
  "alwaysInjectSelector": [
    {
      "fields": {
        "property1": {
          "kind": {}
        }
      },
    },
  ],
}
```

```
        "property2": {
            "kind": {}
        }
    },
    ],
    "defaultTemplates": [
        "string"
    ],
    "enableNamespacesByDefault": {
        "value": true
    },
    "injectedAnnotations": {
        "fields": {
            "property1": {
                "kind": {}
            },
            "property2": {
                "kind": {}
            }
        }
    },
    "injectionURL": "string",
    "neverInjectSelector": [
        {
            "fields": {
                "property1": {
                    "kind": {}
                },
                "property2": {
                    "kind": {}
                }
            }
        }
    ]
}
```

```
        }
      }
    }
  ],
  "objectSelector": {
    "fields": {
      "property1": {
        "kind": {}
      },
      "property2": {
        "kind": {}
      }
    }
  },
  "rewriteAppHTTPProbe": {
    "value": true
  },
  "templates": {
    "fields": {
      "property1": {
        "kind": {}
      },
      "property2": {
        "kind": {}
      }
    }
  },
  "useLegacySelectors": {
    "value": true
  }
}
```

```

    },
    "istioConfig": {
      "accessLogEncoding": "TEXT",
      "accessLogFile": "/dev/stdout",
      "accessLogFormat": "[%START_TIME%
\\\"%REQ(:METHOD)% %REQ(X-ENVOY-ORIGINAL-PATH?:PATH)% %PROTOCOL%
\\\" %RESPONSE_CODE% %RESPONSE_FLAGS% %RESPONSE_CODE_DETAILS% %
CONNECTION_TERMINATION_DETAILS%\\\"%UPSTREAM_TRANSPORT_FAILURE
_REASON%\\\" %BYTES_RECEIVED% %BYTES_SENT% %DURATION% %RESP(X-EN
VOY-UPSTREAM-SERVICE-TIME)% \\\"%REQ(X-FORWARDED-FOR)%\\\"
\\\"%REQ(USER-AGENT)%\\\" \\\"%REQ(X-REQUEST-ID)%\\\"
\\\"%REQ(:AUTHORITY)%\\\"
\\\"%UPSTREAM_HOST%\\\" %UPSTREAM_CLUSTER% %UPSTREAM_LOCAL_ADDR
ESS% %DOWNSTREAM_LOCAL_ADDRESS% %DOWNSTREAM_REMOTE_ADDRE
SS% %REQUESTED_SERVER_NAME% %ROUTE_NAME%\\n",
      "defaultConfig": {
        "concurrency": 2,
        "enableDnsCapture": false,
        "holdApplicationUntilProxyStarts": true,
        "interceptionMode": "REDIRECT",
        "proxyMetadata": {
          "ISTIO_META_DNS_AUTO_ALLOCATE": "true",
          "ISTIO_META_DNS_CAPTURE": "true"
        },
        "terminationDrainDuration": "5s"
      },
      "outboundTrafficPolicy": "ALLOW_ANY",
      "telemetry": true
    },
    "monitorSetting": {
      "enableMetrics": true,

```

```
        "traceSampling": 1
    },
    "opa": {
        "enabled": true,
        "logLevel": "error"
    }
},
"network": {
    "SecurityGroupId": "cg-****",
    "vpcId": "vpc-****"
}
},
"statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.2.2 获取 KubeConfig 配置

本接口用于获取连接服务网格的 KubeConfig 配置。

接口 URL: /csm/v1/open/servicemesh/get\_kube\_config

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8****	String	是	服务网格 id

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
-----	-----	------	------

code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
kubeConfig	apiVersion: v1 clusters: - cluster: server: https://xx.xx.xx.xx:6443 certificate-authority-data: ****	String	kubeconfig 配置内容

返回示例：成功

```

{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "kubeConfig": "apiVersion: v1 clusters: - cluster: server:
https://xx.xx.xx.xx:6443 certificate-authority-data: ****"
  },
  "statusCode": 2000
}

```

返回示例：失败见 4.10

#### 4.2.3 展示所有网格列表

本接口展示租户内所有的网格列表。

接口 URL: /csm/v1/open/servicemesh/list

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
-----	-----	------	------	------

无 URI 参数

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
meshInfos	Array	Array[ServiceMeshInfo]	服务网格信息

表 ServiceMeshInfo

参数名	示例值	参数类型	参数描述
clusterType	BASE	String	实例类型。BASE 为基础版。
createdTime	2023-07-18T10:21:20+08:00	String	创建时间
name	Name	String	服务网格名称
regionCode	d8bbd132b5****	String	资源池编码
regionName	内蒙 6	String	资源池名称
serviceMeshId	e779bd8*****	String	服务网格 id
status	READY	String	服务网格状态 READY: 运行中 NOTREADY: 未就绪
version	1.16	String	服务网格版本

返回示例: 成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "meshInfos": [
      {
        "clusterType": "ENTERPRISE",
        "createTime": "2023-07-18T10:21:20+08:00",
        "name": "Name",
        "regionCode": "d8bbd132b5****",
        "regionName": "内蒙 6",
        "serviceMeshId": "e779bd8****",
        "status": "READY",
        "version": "1.16"
      }
    ]
  },
  "statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.2.4 修改网格名称

本接口修改网格的名称。调用时需要传入网格 id。

接口 URL: /csm/v1/open/servicemesh/update\_name

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
name	NewName	String	是	服务网格的新名称
serviceMeshId	e779bd8****	String	是	服务网格 id

请求示例:

```
{
  "name": "NewName",
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

## 4.3 管理自定义资源模板

### 4.3.1 展示自定义资源模板

本接口查询服务网格中常见使用场景下的自定义资源 YAML 模板。

接口 URL: /csm/v1/open/resource/crd\_template/list

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
kind	String	String	是	需要获取常见模板的 istio 资源类型。目前支持的取值包括: AuthorizationPolicy RequestAuthentication PeerAuthentication

				WorkloadGroup WorkloadEntry Sidecar EnvoyFilter ServiceEntry Gateway DestinationRule VirtualService
--	--	--	--	--

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
templates	Arrary	Arrary	指定 Istio 资源类型的资源常用YAML模板

表 templates

参数名	示例值	参数类型	参数描述
chineseName	HTTP 基础路由	String	YAML 模板的中文名称
englishName	HTTP basic routing	String	YAML 模板的英文名称
yaml	apiVersion: networking.istio.io/v1beta1\n *****	String	YAML 模板的内容

返回示例: 成功

```
{"code": "CSM_2000", "message": "SUCCESS", "returnObj": {"templates":
```

```
[ { "chineseName": "HTTP 基础路由", "englishName": "HTTP basic routing",
"yaml": "apiVersion: networking.istio.io/v1beta1\n *****" } ] }, "statusCode":
2000 }
```

返回示例：失败

## 4.4 管理元数据

### 4.4.1 查询服务网格基本信息

本接口查询查询服务网格基本信息

接口 URL: /csm/v1/open/metadata/get

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无 URI 参数				

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
versionCRDs	Array	Array[VersionCRDs]	各版本网格包含的 crd 资源

表 VersionCRDs

参数名	示例值	参数类型	参数描述
crds	Array	Array[CRD]	服务网格的 crd 资源
version	1.16.2	String	网格的版本信息

表 CRD

参数名	示例值	参数类型	参数描述
apiGroup	networking.istio.io	String	CRD Group
apiVersion	v1alpha3	String	CRD 版本号
kind	DestinationRule	String	CRD 类型

返回示例：成功

```

{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "versionCRDs": [
      {
        "crds": [
          {
            "apiGroup": "networking.istio.io",
            "apiVersion": "v1alpha3",
            "kind": "DestinationRule"
          }
        ],
        "version": "1.16.2"
      }
    ]
  },
  "statusCode": 2000
}

```

返回示例：失败见 4.10

## 4.5 管理 Sidecar 配置

### 4.5.1 查询 Sidecar 配置

本接口查询命名空间级别的 Sidecar 配置设置。

接口 URL: /csm/v1/open/Sidecar/config/get

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id
namespace	default	String	否	命名空间, 不填则默认为 istio-system

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体, 接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
customProxyConfig	Object	CustomProxyConfig	命名空间级别配置内容

表 customProxyConfig

参数名	示例值	参数类型	参数描述
concurrency	2	Integer	Istio-Proxy 线程数
enableDnsCapture	false	Boolean	是否开启 istio 内置 dns

excludeIPRanges	127.0.0.1	String	不拦截对外访问的地址范围
excludeInboundPorts	80,8080	String	设置端口使入口流量免于经过 Sidecar 代理
excludeOutboundPorts	80,8080	String	设置端口使出口流量免于经过 Sidecar 代理
holdApplicationUntilProxyStarts	true	Boolean	生命周期管理
includeIPRanges	192.168.0.0/16	String	拦截对外访问的地址范围
includeInboundPorts	443	String	设置端口使入口流量经过 Sidecar 代理
includeOutboundPorts	443	String	设置端口使出口流量经过 Sidecar 代理
interceptionMode	REDIRECT	String	Sidecar 对入向流量的拦截策略, 取值为 REDIRECT, TPROXY, NONE
lifecycleStr	<pre>{\ "postStart\":{\ "exec\":{\ "command\":[\ "pilot-agent\","wait \"]}],\ "preStop\":{\ "exec\":{\ "command\":[\ "/bin/sh\","-c\","sleep 15\"]}}}</pre>	String	Sidecar 代理生命周期 JSON 字符串
logLevel	info	String	日志等级, 取值范围为 trace, debug, info, warning, error, critical, off
outboundTrafficPolicy	ALLOW_ANY	String	出向流量策略。取值: ALLOW_ANY: 允许访问所有外部服务。 REGISTRY_ONLY: 只能访问注册到网格内的服务。
proxyMetadata	ISTIO_META_DNS_CAPTURE:true,ISTIO_META_DNS_AUTO_ALLOCATE:true	map[string]string	环境变量
proxyStatsMatcher	Object	CustomProxyConfig_ProxyStatsMatcher	统计相关指标
SidecarProxyInitResourceLimit	Object	ResourcesLimit	Sidecar 代理初始化容器资源限制
SidecarProxyInitResourceRequest	Object	ResourcesRequest	Sidecar 代理初始化容器资源最低申请额度
SidecarProxyResourceLimit	Object	ResourcesLimit	Sidecar 代理容器资源

			限制
SidecarProxyResourceRequest	Object	ResourcesRequest	Sidecar 代理容器资源最低申请额度
terminationDrainDuration	5s	String	Istio Proxy 终止等待时长

表 CustomProxyConfig\_ProxyStatsMatcher

参数名	示例值	参数类型	参数描述
inclusion_prefixes	Array	Array	Envoy 监控统计的指标前缀
inclusion_regexps	Array	Array	Envoy 监控统计的指标正则表达式
inclusion_suffixes	Array	Array	Envoy 监控统计的指标后缀

表 ResourcesLimit

参数名	示例值	参数类型	参数描述
resourceCPULimit	100m	String	CPU 资源限制
resourceMemoryLimit	60Mi	String	内存资源限制

表 ResourcesRequest

参数名	示例值	参数类型	参数描述
resourceCPURequest	60m	String	CPU 资源最低要求
resourceMemoryRequest	30Mi	String	内存资源最低要求

返回示例：成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "customProxyConfig": {
      "concurrency": 2,
      "enableDnsCapture": false,
      "excludeIPRanges": "127.0.0.1",

```

```
"excludeInboundPorts": "80,8080",
"excludeOutboundPorts": "80,8080",
"holdApplicationUntilProxyStarts": true,
"includeIPRanges": "192.168.0.0/16",
"includeInboundPorts": "443",
"includeOutboundPorts": "443",
"interceptionMode": "REDIRECT",
"lifecycleStr":
"{\"postStart\":{\"exec\":{\"command\":[\"pilot-agent\",\"wait\"]}},\"preStop\":{\"exec\":{\"command\":[\"/bin/sh\",\"-c\",\"sleep 15\"]}}}\",
"LogLevel": "trace",
"outboundTrafficPolicy": "ALLOW_ANY",
"proxyMetadata": {
  "ISTIO_META_DNS_AUTO_ALLOCATE": "true",
  "ISTIO_META_DNS_CAPTURE": "true"
},
"proxyStatsMatcher": {
  "inclusion_prefixes": [
    "string"
  ],
  "inclusion_regexp": [
    "string"
  ],
  "inclusion_suffixes": [
    "string"
  ]
},
"SidecarProxyInitResourceLimit": {
  "resourceCPUlimit": "100 m",
  "resourceMemoryLimit": "60 Mi"
},
```

```
"SidecarProxyInitResourceRequest": {
  "resourceCPURequest": "60 m",
  "resourceMemoryRequest": "30 Mi"
},
"SidecarProxyResourceLimit": {
  "resourceCPULimit": "100 m",
  "resourceMemoryLimit": "60 Mi"
},
"SidecarProxyResourceRequest": {
  "resourceCPURequest": "60 m",
  "resourceMemoryRequest": "30 Mi"
},
"terminationDrainDuration": "5s"
}
},
"statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.5.2 更新 Sidecar 配置

本接口更新命名空间级别的 Sidecar 配置。

接口 URL: /csm/v1/open/Sidecar/config/update

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	否	命名空间
serviceMeshId	e779bd8*****	String	是	服务网格 id
customProxyConfig	Object	Object	否	命名空间级别配置内容

表 customProxyConfig

参数名	示例值	参数类型	参数描述
concurrency	2	Integer	Istio-Proxy 线程数
enableDnsCapture	false	Boolean	是否开启 istio 内置 dns
excludeIPRanges	127.0.0.1	String	不拦截对外访问的地址范围
excludeInboundPorts	80,8080	String	设置端口使入口流量免于经过 Sidecar 代理
excludeOutboundPorts	80,8080	String	设置端口使出口流量免于经过 Sidecar 代理
holdApplicationUntilProxyStarts	true	Boolean	生命周期管理
includeIPRanges	192.168.0.0/16	String	拦截对外访问的地址范围
includeInboundPorts	443	String	设置端口使入口流量经过 Sidecar 代理
includeOutboundPorts	443	String	设置端口使出口流量经过 Sidecar 代理
interceptionMode	REDIRECT	String	Sidecar 对入向流量的拦截策略, 取值为 REDIRECT,TPROXY,NONE
lifecycleStr	<pre>{\ "postStart\":{\ "exec\":{\ "command\":[\ "pilot-agent\","\ "wait \"]\}],\ "preStop\":{\ "exec\":{\ "command\":[\ "/bin/sh\","\ "-c\","\ "sleep 15\"]\}}}</pre>	String	Sidecar 代理生命周期 JSON 字符串
logLevel	info	String	日志等级, 取值范围为 trace,debug,info,warning,error,critical,off
outboundTrafficPolicy	ALLOW_ANY	String	出向流量策略。取值: ALLOW_ANY: 允许访问所有外部服务。 REGISTRY_ONLY: 只能访问注册到网格内的服务。
proxyMetadata	ISTIO_META_DNS_CAPTURE:true,ISTIO_META_DNS_AUTO_ALLOCATE:true	map[string]string	环境变量
proxyStatsMatcher	Object	CustomProxyConfig_ProxyStatsMatcher	统计相关指标
SidecarProxyInitResourceLimit	Object	ResourcesLimit	Sidecar 代理初始化容

			器资源限制
SidecarProxyInitResourceRequest	Object	ResourcesRequest	Sidecar 代理初始化容器资源最低申请额度
SidecarProxyResourceLimit	Object	ResourcesLimit	Sidecar 代理容器资源限制
SidecarProxyResourceRequest	Object	ResourcesRequest	Sidecar 代理容器资源最低申请额度
terminationDrainDuration	5s	String	Istio Proxy 终止等待时长

表 CustomProxyConfig\_ProxyStatsMatcher

参数名	示例值	参数类型	参数描述
inclusion_prefixes	Array	Array	Envoy 监控统计的指标前缀
inclusion_regexps	Array	Array	Envoy 监控统计的指标正则表达式
inclusion_suffixes	Array	Array	Envoy 监控统计的指标后缀

表 ResourcesLimit

参数名	示例值	参数类型	参数描述
resourceCPULimit	100m	String	CPU 资源限制
resourceMemoryLimit	60Mi	String	内存资源限制

表 ResourcesRequest

参数名	示例值	参数类型	参数描述
resourceCPURequest	60m	String	CPU 资源最低要求
resourceMemoryRequest	30Mi	String	内存资源最低要求

请求示例:

```
{
  "customProxyConfig": {
    "concurrency": 2,
```

```
"enableDnsCapture": false,
"excludeIPRanges": "127.0.0.1",
"excludeInboundPorts": "80,8080",
"excludeOutboundPorts": "80,8080",
"holdApplicationUntilProxyStarts": true,
"includeIPRanges": "192.168.0.0/16",
"includeInboundPorts": "443",
"includeOutboundPorts": "443",
"interceptionMode": "REDIRECT",
"lifecycleStr":
"{\"postStart\":{\"exec\":{\"command\":[\"pilot-agent\",\"wait\"]},\"preStop\":{\"exec\":{\"command\":[\"/bin/sh\",\"-c\",\"sleep 15\"]}}},
  \"logLevel\": \"trace\",
  \"outboundTrafficPolicy\": \"ALLOW_ANY\",
  \"proxyMetadata\": {
    \"ISTIO_META_DNS_AUTO_ALLOCATE\": \"true\",
    \"ISTIO_META_DNS_CAPTURE\": \"true\"
  },
  \"proxyStatsMatcher\": {
    \"inclusion_prefixes\": [\"string\"],
    \"inclusion_regexps\": [\"string\"],
    \"inclusion_suffixes\": [\"string\"]
  },
  \"SidecarProxyInitResourceLimit\": {
    \"resourceCPULimit\": \"100 m\",
    \"resourceMemoryLimit\": \"60 Mi\"
  },
  \"SidecarProxyInitResourceRequest\": {
    \"resourceCPURequest\": \"60 m\",
    \"resourceMemoryRequest\": \"30 Mi\"
  },
```

```

"SidecarProxyResourceLimit": {
  "resourceCPULimit": "100 m",
  "resourceMemoryLimit": "60 Mi"
},
"SidecarProxyResourceRequest": {
  "resourceCPURequest": "60 m",
  "resourceMemoryRequest": "30 Mi"
},
"terminationDrainDuration": "5s"
},
"namespace": "istio-system",
"serviceMeshId": "e779bd8*****"
}

```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```

{"code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }

```

返回示例: 失败见 4.10

#### 4.5.3 修改网格注入配置

本接口修改网格注入设置。

接口 URL: /csm/v1/open/inject\_config/update

Content-Type: application/json

请求方式：POST

请求体参数说明：

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	是	命名空间
serviceMeshId	e779bd8*****	String	是	服务网格 id
enableIstiInjection	true	Boolean	否	是否开启注入

请求示例：

```
{ "enableIstiInjection": true, "namespace": "default", "serviceMeshId": "e779bd8*****" }
```

返回参数说明：

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例：成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例：失败见 4.10

## 4.6 管理 OPA 配置

### 4.6.1 关闭 OPA 注入

本接口用于关闭 opa 注入。

接口 URL：/csm/v1/open/opa/disable

Content-Type：application/json

请求方式：POST

请求体参数说明：

参数名	示例值	参数类型	是否必填	参数描述
serviceMeshId	e779bd8*****	String	是	服务网格 id

请求示例:

```
{
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

#### 4.6.2 开启 OPA 注入

本接口用于开启 OPA 注入。

接口 URL: /csm/v1/open/opa/enable

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
serviceMeshId	e779bd8*****	String	是	服务网格 id

请求示例:

```
{
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

#### 4.6.3 修改 OPA 注入范围

本接口用于指定哪些范围的服务需要开启 OPA 注入, 需要指定集群 id 和 namespace。

指定后, 相应集群和相应命名空间下新建的 pod 都会被进行 OPA 注入。

接口 URL: /csm/v1/open/open/opa/update

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	否	命名空间, 不传默认为 default。
serviceMeshId	e779bd8*****	String	是	服务网格 id
enableOpalInject	true	Boolean	否	开启 opa 注入开关, 默认为 false
clusterId	kjye*****	String	是	集群 id

请求示例:

```

{
  "enableOpalInject": true,
  "namespace": "default",
  "serviceMeshId": "e779bd8*****",
  "clusterId": "kjye*****"
}

```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```

{"code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }

```

#### 4.6.4 创建 OPA 策略

创建 OPA 策略。

接口 URL: /csm/v1/open/opa/policy/create

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	否	命名空间
serviceMeshId	e779bd8*****	String	是	服务网格 id
crContent	String	String	是	Opa policy

请求示例:

```
{
  "crContent": "string",
  "namespace": "string",
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

#### 4.6.5 删除 OPA 策略

删除 OPA 策略。

接口 URL: /csm/v1/open/opa/policy/delete

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	是	命名空间
serviceMeshId	e779bd8*****	String	是	服务网格 id
name	object-policy-dd	String	是	OPA 策略名称

请求示例:

```
{
  "name": "object-policy-dd",
  "namespace": "string",
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

#### 4.6.6 更新 OPA 策略

更新 OPA 策略。

接口 URL: /csm/v1/open/opa/policy/update

Content-Type: application/json

请求方式: POST

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
namespace	default	String	是	命名空间
serviceMeshId	e779bd8*****	String	是	服务网格 id
crContent	String	String	是	Opa policy

请求示例:

```
{
  "crContent": "string",
  "namespace": "string",
  "serviceMeshId": "e779bd8*****"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

返回示例: 成功

```
{ "code": "CSM_2000", "message": "SUCCESS", "returnObj": {}, "statusCode": 2000 }
```

返回示例: 失败见 4.10

#### 4.6.7 查询 OPA 策略详情

接口查询 OPA 策略详情, 入参为策略名, 策略名由创建时传入的 crContent 中指定。

接口 URL: /csm/v1/open/opa/policy/detail

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id
namespace	default	String	是	命名空间
name	object-policy-dd	String	是	策略名

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
metadata	Object	Object	OPA 策略元数据
spec	Object	Object	OPA 策略详细属性

表 metadata

参数名	示例值	参数类型	参数描述
creationTimestamp	String	String	创建时间
labels	Object	map[string]string	标签
name	String	String	OPA 策略名
namespace	String	String	命名空间
resourceVersion	String	String	资源版本

表 spec

参数名	示例值	参数类型	参数描述
policy	String	String	具体策略, rego 语法
workloadSelector	Object	Object	策略生效范围, 以 labels 为准

表 workloadSelector

参数名	示例值	参数类型	参数描述
labels	Object	map[string]string	标签

返回示例：成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {
    "metadata": {
      "creationTimestamp": "string",
      "labels": {
        "version": "string"
      },
      "name": "string",
      "namespace": "string",
      "resourceVersion": "string"
    },
    "spec": {
      "policy": "package istio.authz\nimport input.attributes.request.http
as http_request\ndefault allow = false",
      "workloadSelector": {
        "labels": {
          "version": "string"
        }
      }
    }
  },
  "statusCode": 2000
}
```

返回示例：失败见 4.10

#### 4.6.8 查询 OPA 策略列表

调用 opa/policy/list 接口查询 OPA 策略列表

接口 URL: /csm/v1/open/opa/policy/list

Content-Type: text/plain

请求方式: GET

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
service_mesh_id	e779bd8*****	String	是	服务网格 id
namespace	default	String	是	命名空间
keyword	String	String	否	搜索关键词
page_code	String	String	否	分页 code
page_size	10	Integer	否	分页大小

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj

参数名	示例值	参数类型	参数描述
List	Array	Array	pod 名称
pageCode	String	String	分页标志
remainingItemCount	Integer	Integer	list 请求返回的 remainingItemCount

			字段
--	--	--	----

表 List

参数名	示例值	参数类型	参数描述
creationTimestamp	String	String	创建时间
labels	Object	map[string]string	标签
name	String	String	OPA 策略名
namespace	String	String	命名空间
resourceVersion	String	String	资源版本

返回示例：成功

```
{
  "statusCode": 2000,
  "message": "SUCCESS",
  "code": "CSM_2000",
  "returnObj": {
    "list": [
      {
        "creationTimestamp": "2023-10-18T04:20:02Z",
        "label": {
          "version": "v1"
        },
        "name": "object-policy",
        "namespace": "default",
        "resourceVersion": "8899494"
      },
      {
        "creationTimestamp": "2023-10-18T02:01:17Z",
        "label": {
          "version": "v2"
        },
        "name": "reviews1",
```

```
"namespace": "default",
  "resourceVersion": "8833636"
}
]
}
}
```

返回示例：失败见 4.10

## 4.7 CSM 网关管理

### 4.7.1 更新 CSM 网关

本接口用于更新 CSM 网关。

接口 URL: /csm/v1/open/gateway/update

Content-Type: application/json

请求方式: POST

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
gatewayType	Ingress	String	是	网关类型，取值 Ingres: 入口网关 Egress: 出口网关
yamlStr	String	String	是	网关配置的 YAML 内容
serviceMeshId	e779bd8*****	String	是	服务网格 id

请求示例:

```
{
  "serviceMeshId": "d4a618d*****",
  "gatewayType": "Ingress",
  "yamlStr": "
  enabled: true
```

```
namespace: istio-system
name: my-gateway
label:
  key1: value1
  key2: value2
hub: my-hub"
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体,接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj 为空

返回示例: 成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {},
  "statusCode": 2000
}
```

返回示例: 失败见 4.10

#### 4.7.2 删除 CSM 网关

本接口用于删除 CSM 网关。

接口 URL: /csm/v1/open/gateway/delete

Content-Type: application/json

请求方式: POST

URI 参数说明:

参数名	示例值	参数类型	是否必填	参数描述
无请求参数 KEY/VALUE 类型				

请求体参数说明:

参数名	示例值	参数类型	是否必填	参数描述
gatewayType	Ingress	String	是	网关类型, 取值 Ingres: 入口网关 Egress: 出口网关
serviceMeshId	e779bd8*****	String	是	服务网格 id

请求示例:

```
{  
  "serviceMeshId": "d4a618d*****",  
  "gatewayType": "Ingress"  
}
```

返回参数说明:

参数名	示例值	参数类型	参数描述
code	CSM_2000	String	错误码
message	SUCCESS	String	错误提示信息
returnObj	Object	Object	API 返回数据体, 接口如果没有返回字段时也可能可能返回{}或[]
statusCode	2000	Integer	调用错误时返回的错误编码

表 returnObj 为空

返回示例: 成功

```
{
  "code": "CSM_2000",
  "message": "SUCCESS",
  "returnObj": {},
  "statusCode": 2000
}
```

返回示例：失败见 4.10

## 4.8 接口失败示例

所有接口失败返回均为：

```
{
  "statusCode":1013,
  "code":"CSM_1013",
  "message":"please check the request params"
}
```

其中 code 为 CSM\_开头的字符串，后面数字为具体错误码。

# 5 附录

## 5.1 返回码列表

错误码	错误码标识	错误描述
2000	SUCCESS	成功
5000	INTERNAL_SERVER_ERROR	未知的内部错误，稍后重试
1000	USER_NOT_LOGIN	用户未登陆
1001	FORBIDDEN	禁止访问，没有访问资源的权限
1002	DATA_EXIST	数据已存在，请勿重复创建
1003	DATA_NOT_EXIST	数据不存在
1004	OPERATION_TIMEOUT	操作超时
1005	OPERATION_TOO_FREQUENT	操作太频繁
1006	CACHE_ERROR	缓存数据库繁忙，稍后重试

1007	DB_OPERATE_ERROR	数据库繁忙，稍后重试
1008	ES_OPERATE_ERROR	es 数据库繁忙，稍后重试
1009	DATA_ERROR	数据异常（数据不一致，数据不符合预期，数据值异常等）
1010	SESSION_ERROR	会话验证失败，请先登录
1011	SYSTEM_ERROR	系统繁忙，稍后重试
1012	UNKNOWN_ERROR	系统繁忙，稍后重试
1013	REQUEST_PARAMS_ILLEGAL	参数不合法
1014	RPC_EXCEPTION_ERROR	组件控制台之间 RPC 返回异常
1015	RPC_TIME_OUT	RPC 超时
1016	APP_RPC_TIMEOUT	APP RPC 超时
1017	UPLOAD_PICTURE_FORMAT_ILLEGAL	文件上传类型仅支持 png,jpg,bmp,cif,gif
1018	MQ_ERROR	消息队列访问繁忙，稍后重试
1019	REDISSON_INIT_ERR	执行器服务初始化错误，需要将 RScheduled 执行器服务销毁方法设置为空，请更改服务密钥名称或删除密钥
1020	HTTP_ERROR	HTTP 请求失败
1021	ASYNC_ERROR	异步调用异常
1022	ASYNC_TIMEOUT	异步超时
1023	NACOS_INSTANCE_ERROR	Nacos 组件实例异常
1024	K8s_INSTANCE_ERROR	K8s 组件实例异常
1025	PAAS_GATEWAY_ERROR	Paas 网关异常