云搜索服务（Cloud Search Service）

API

文档版本	01
发布日期	2024-11-22

华为云计算技术有限公司

商标声明

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

目 录

1 使用前必读

概述

欢迎使用云搜索服务（Cloud Search Service，简称CSS）。云搜索服务为您提供托管的分布式搜索引擎服务，完全兼容开源Elasticsearch搜索引擎，支持结构化、非结构化文本的多条件检索、统计、报表。

本文档提供了云搜索服务API的描述、语法、参数说明及样例等内容。云搜索服务提供了集群管理接口、词库管理接口和快照管理接口，通过配合使用这些接口，您可以轻松地对云搜索服务进行相关操作，如创建、查询、删除、重启、扩容集群等。

在调用云搜索服务的API之前，请确保已经充分了解云搜索服务相关概念，详细信息请参见什么是云搜索服务。

支持的版本

云搜索服务中Elasticsearch搜索引擎目前支持7.6.2和7.10.2版本，OpenSearch搜索引擎目前支持1.3.6版本，Logstash目前支持7.10.0版本。

约束与限制

- 您能创建的CSS集群数量与配额有关系，如果您想查看服务配额、扩大配额，具体请参见服务配额。
- 更详细的限制请参见具体API说明。

终端节点

终端节点（Endpoint）即调用API的**请求地址**，不同服务不同区域的终端节点不同，您可以从地区和终端节点中查询所有服务的终端节点。

基本概念

- 账号

  用户注册时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用用户进行日常管理工作。

- 用户

  由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。

  在**API凭证**下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。

- 区域

  指云资源所在的物理位置，同一区域内可用区间内网互通，不同区域间内网不互通。通过在不同地区创建云资源，可以将应用程序设计的更接近特定客户的要求，或满足不同地区的法律或其他要求。

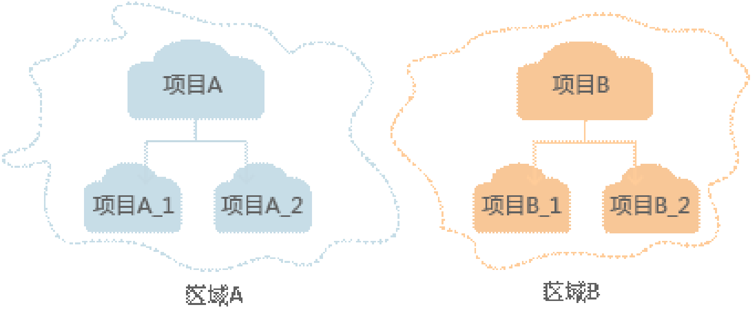
- 可用区

  一个可用区是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

- 项目

  区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中购买资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中资源，使得资源的权限控制更加精确。

  图 1-1 项目隔离模型

- Checkpoint：消费检查点。应用程序消费数据时，记录已消费数据的最新序列号作为检查点。当重新消费数据时，可根据此检查点继续消费。
- APP：应用程序标识符。当多个应用程序分别消费同一通道的数据时，为区分不同应用程序的消费检查点，使用APP作为标识。
- 企业项目

  企业项目是项目的升级版，针对企业不同项目间资源的分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。

  关于企业项目ID的获取及企业项目特性的详细信息，请参见《**企业管理服务用户指南**》。

2 API 概览

CSS提供的接口为符合RESTful API设计规范的自研接口。通过CSS的自研接口，您可以使用CSS的如表2-1所示的功能。

表 2-1 接口说明

类型	说明
集群管理接口	可以通过接口创建集群、扩容集群、查看集群详情、获取实例规格列表等。
词库管理接口	提供自定义词库功能，可以通过接口加载自定义库、查询词库状态或删除自定义词库。
Kibana公网访问接口	Kibana公网访问功能支持通过接口进行开启、关闭、修改等操作。
Logstash接口	可以通过接口创建Logstash类型集群的配置文件、启动pipeline迁移数据等操作。
日志管理接口	可以通过日志管理接口进行开启、关闭、修改、查询等操作。
公网访问接口	可以通过公网访问接口进行开启、关闭、修改等操作。
快照管理接口	快照功能可应用于数据备份与恢复，可以通过接口创建快照、恢复快照或者删除快照等。
终端节点接口	可以通过终端节点接口进行开启、关闭、获取连接、更新连接等操作。
参数配置接口	可以通过参数配置接口修改集群的参数配置。
负载均衡接口	可以通过负载均衡接口打开、关闭、获取负载均衡器信息。
智能运维接口	可以通过智能运维接口创建、删除、获取检测任务。

表 2-2 集群管理接口

API	说明
创建集群	该接口用于创建集群。
查询集群列表	该接口用于查询并显示集群列表以及集群的状态。
查询集群详情	该接口用于查询并显示单个集群详情。
删除集群	此接口用于删除集群。
按需集群转包周期	该接口用于按需集群转包周期集群。
修改集群名称	该接口用于修改集群名称。
修改密码	该接口用于修改集群密码。
重启集群	此接口用于重启集群，重启集群将导致业务中断。
扩容集群	该接口用于集群扩容实例（仅支持扩容elasticsearch实例）。
扩容实例的数量和存储容量	该接口用于集群扩容不同类型实例的个数以及存储容量。
变更规格	该接口用于变更集群规格。
获取实例规格列表	该接口用于查询并显示支持的实例规格对应的ID。
查询所有标签	该接口用于查询指定region下的所有标签集合。
查询指定集群的标签	该接口用于查询指定集群的标签信息。
添加指定集群标签	该接口用于给指定集群添加标签。
删除集群标签	此接口用于删除集群标签
批量添加或删除集群标签	该接口用于对集群批量添加或删除标签。
指定节点类型规格变更	修改集群规格。支持修改ess， ess-cold， ess-client，ess-master节点类型。
指定节点缩容	该接口可以对集群现有节点中指定节点进行缩容。
指定节点类型缩容	该接口用于集群缩容不同类型实例的个数以及存储容量。
节点替换	该接口用于替换失败节点。
切换安全组	该接口可以在集群创建成功后，修改安全组。
安全模式修改	该接口用于切换集群的安全模式。
添加独立master、client	该接口用于添加master或者client节点。
下载安全证书	该接口用于下载安全证书。
创建集群V2	该接口用于创建拥有多种不同节点类型（ess，ess-cold，ess-client，ess-master）组合的集群。

API	说明
重启集群V2	该接口可以用于重启当前集群拥有的全部节点类型，或部分节点类型组合的节点。
滚动重启	该接口会一个一个重启节点，在索引数量比较多的情况下耗时较长

表 2-3 词库管理接口

API	说明
加载自定义词库	该接口用于加载存放于OBS的自定义词库。
查询自定义词库状态	该接口用于查询自定义词库的加载状态。
删除自定义词库	该接口用于删除自定义词库。

表 2-4 Kibana 公网访问接口

API	说明
开启Kibana公网访问	该接口用于开启Kibana公网访问。
关闭Kibana公网访问	该接口用于关闭Kibana公网访问。
修改Kibana公网带宽	该接口用于修改Kibana公网带宽。
修改Kibana公网访问控制	该接口通过修改kibana白名单，修改kibana的访问权限。
关闭Kibana公网访问控制	该接口用于关闭Kibana公网访问控制。

表 2-5 Logstash 接口

API	说明
创建配置文件	该接口用于创建配置文件。
更新配置文件	该接口用于更新配置文件。
启动pipeline迁移数据	该接口用于启动pipeline迁移数据。
停止pipeline迁移数据	该接口用于停止pipeline迁移数据。
添加到自定义模板	该接口用于添加到自定义模板。
连通性测试	该接口用于连通性测试。
查询模板列表	该接口用于查询模板列表。

API	说明
查询配置文件列表	查询配置文件列表。
查询pipeline列表	该接口用于查询pipeline列表。
查询操作记录	该接口用于查询操作记录。
查询配置文件内容	该接口用于查询配置文件内容。
删除配置文件	删除配置文件。
删除自定义模板	该接口用于删除自定义模板。

表 2-6 日志管理接口

API	说明
开启日志功能	该接口用于开启日志管理功能。
关闭日志功能	该接口用于关闭日志管理功能。
查询作业列表	该接口用于查询具体某个集群的日志任务记录列表。
查询日志基础配置	该接口用于日志基础配置查询。
修改日志基础配置	该接口用于修改日志基础配置。
开启日志自动备份策略	该接口用于日志自动备份策略开启。
关闭日志自动备份策略	该接口用于日志自动备份策略关闭。
备份日志	该接口用于备份日志。
查询日志	该接口用于查询日志信息。

表 2-7 公网访问接口

API	说明
开启公网访问	该接口用于开启公网访问。
关闭公网访问	该接口用于关闭公网访问。
修改公网访问带宽	该接口用于修改公网访问带宽。
开启公网访问控制白名单	该接口用于开启公网访问控制白名单。
关闭公网访问控制白名单	该接口用于关闭公网访问控制白名单。

表 2-8 快照管理接口

API	说明
自动设置集群快照的基础配置（不推荐使用）	该接口用于自动设置集群快照的基础配置，包括配置OBS桶和IAM委托。
修改集群快照的基础配置	该接口用于修改集群快照的基础配置，可修改OBS桶和IAM委托。
手动创建快照	该接口用于手动创建一个快照。
恢复快照	该接口用于手动恢复一个快照。
删除快照	该接口用于删除快照。
设置自动创建快照策略	该接口用于设置自动创建快照，默认一天创建一个快照。
查询自动创建快照的策略	该接口用于查询集群自动创建快照的策略。
查询快照列表	该接口用于查询指定集群的快照列表。
停用快照功能	该接口用于停用快照功能。
开启自动创建快照功能	该接口用于打开自动备份功能。
关闭自动创建快照功能	该接口用于关闭自动备份功能

表 2-9 终端节点接口

API	说明
开启终端节点服务	该接口用于开启终端节点服务。
关闭终端节点服务	该接口用于关闭终端节点服务。
获取终端节点连接	该接口用于获取终端节点连接。
更新终端节点连接	该接口用于更新终端节点连接。
修改终端节点服务白名单	该接口用于修改终端节点服务访问控制。

表 2-10 参数配置接口

API	说明
修改参数配置	该接口用于修改参数配置。
获取参数配置任务列表	该接口可获取参数配置的任务列表。
获取参数配置列表	该接口用于获取当前集群现有的参数配置列表。

表 2-11 负载均衡

API	说明
查询集群支持的elbv3负载均衡器	展示查询集群支持的elbv3负载均衡器。
打开或关闭ES负载均衡器	该接口打开或关闭ES负载均衡器。
ES监听器配置	该接口用于es监听器配置。
获取该esELB的信息，以及页面需要展示健康检查状态	该接口用于获取该ES集群的ELB信息，以及页面需要展示健康检查状态。
更新ES监听器	该接口用于更新ES监听器。
查询证书列表	该接口用于查询证书列表。

表 2-12 智能运维

API	说明
获取智能运维任务列表及详情	该接口用于获取智能运维任务列表及详情。
创建一次集群检测任务	该接口用于创建一个集群检测任务。
删除一个检测任务记录	该接口用于删除一个检测任务记录。
获取智能运维告警可用的SMN主题	该接口用于获取智能运维告警可用的SMN主题。

3 如何调用 API

3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM服务的管理员创建IAM用户接口说明如何调用API。

您还可以通过这个视频教程了解如何构造请求调用API：**https://bbs.huaweicloud.com/videos/102987** 。

请求 URI

请求URI由如下部分组成。

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

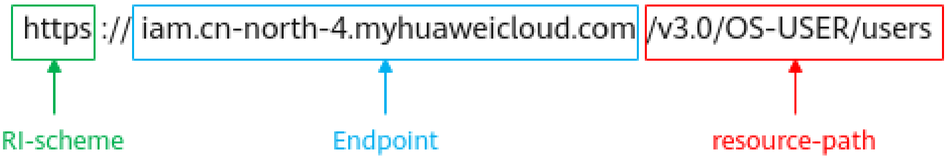
表 3-1 请求 URL

参数	说明
URI-scheme	传输请求的协议，当前所有API均采用**HTTPS**协议。
Endpoint	承载REST服务端点的服务器域名或IP，不同服务在不同区域的Endpoint不同，可以参考终端节点获取。例如IAM服务在“华北-北京一”区域的Endpoint为“iam.cn-north-1.myhuaweicloud.com”。
resource-path	资源路径，即API访问路径。从具体API的URI模块获取，例如“管理员创建IAM用户”接口的resource-path为“/v3.0/OS-USER/users”。
query-string	查询参数，可选，查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。

例如您需要创建IAM用户，由于IAM为全局服务，则使用任一区域的Endpoint（比如“华北-北京四”区域的Endpoint：“iam.cn-north-4.myhuaweicloud.com”），并在管理员创建IAM用户的URI部分找到resource-path（/v3.0/OS-USER/users），拼接起来如下所示。

```
https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
```

图 3-1 URI 示意图



说明

为查看方便，每个具体API的URI，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

- **GET**：请求服务器返回指定资源。
- **PUT**：请求服务器更新指定资源。
- **POST**：请求服务器新增资源或执行特殊操作。
- **DELETE**：请求服务器删除指定资源，如删除对象等。
- **HEAD**：请求服务器资源头部。
- **PATCH**：请求服务器更新资源的部分内容。当资源不存在的时候，PATCH可能会去创建一个新的资源。

在**管理员创建IAM用户**的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

详细的公共请求消息头字段请参见**表3-2**。

表 3-2 公共请求消息头

参数	是否必选	描述
Content-Type	是	消息体的类型（格式）。推荐用户使用默认值“application/json”。
X-Auth-Token	使用Token认证时必选	用户Token。是调用“获取用户Token”接口的响应值，该接口是唯一不需要认证的接口。请求响应成功后在响应消息头（Header）中包含的“X-Subject-Token”的值即为Token值。

参数	是否必选	描述
X-Project-Id	否	子项目ID，在多项目场景中使用。如果云服务资源创建在子项目中，AK/SK认证方式下，操作该资源的接口调用需要在请求消息头中携带X-Project-ID。
X-Sdk-Date	使用AK/SK认证时必选	请求的发生时间，当使用AK/SK方式认证时，使用SDK对请求进行签名的过程中会自动填充该字段。 AK/SK认证的详细说明请参见**认证鉴权**。 格式为(YYYYMMDD'T'HHMMSS'Z')，取值为当前系统的GMT时间。
Authorization	使用AK/SK认证时必选	签名认证信息。当使用AK/SK方式认证时，使用SDK对请求进行签名的过程中会自动填充该字段。 AK/SK认证的详细说明请参见**认证鉴权**。
X-Language	否	请求语言。

对于**管理员创建IAM用户**接口，使用AK/SK方式认证时，添加消息头后的请求如下所示。

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=****************, SignedHeaders=content-type;host;x-sdk-date,
Signature=****************
```

请求消息体

请求消息体通常以结构化格式发出，与请求消息头中Content-type对应，传递除请求消息头之外的内容。若请求消息体中参数支持中文，则中文字符必须为UTF-8编码，并在Content-type中声明字符编码方式，例如：Content-Type: application/json; charset=utf-8。

每个接口的请求消息体内容不同，也并不是每个接口都需要有请求消息体（或者说消息体为空），GET、DELETE操作类型的接口就不需要消息体，消息体具体内容需要根据具体接口而定。

对于**管理员创建IAM用户**接口，您可以从接口的请求部分看到所需的请求参数及参数说明，将消息体加入后的请求如下所示，其中加粗的字段需要根据实际值填写。

- **accountid**为IAM用户所属的账号ID。
- **username**为要创建的IAM用户名。
- **************为IAM用户的登录密码。

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users （中国站）
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users （国际站）
```

```
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=****************, SignedHeaders=content-type;host;x-sdk-date,
Signature=****************

{
    "user": {
        "domain_id": "accountid",
        "name": "username",
        "password": "***********",
        "description": "IAM User Description"
    }
 }
```

到此为止，一个API请求所需要的内容已经准备完成，您可以使用curl、Postman或直接编写代码等方式发送请求调用API。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key)加密调用请求。
- Token认证：通过Token认证调用请求。

AK/SK 认证

说明

- AK/SK签名认证方式仅支持消息体大小12MB以内，12MB以上的请求请使用Token认证。
- AK/SK既可以使用永久访问密钥中的AK/SK，也可以使用临时访问密钥中的AK/SK，但使用临时访问密钥的AK/SK时需要额外携带“X-Security-Token”字段，字段值为临时访问密钥的security_token。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。详细的签名方法和SDK使用方法请参见**API签名指南**。

须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

您也可以通过这个视频教程了解AK/SK认证的使用：**https://bbs.huaweicloud.com/videos/100697** 。

Token 认证

说明

- Token的有效期为24小时，需要使用同一个Token鉴权时，可以缓存起来，避免频繁调用。
- 使用Token前请确保Token离过期有足够的时间，防止调用API的过程中Token过期导致调用API失败。

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。

Token可通过调用**获取用户Token**接口获取，调用本服务API需要project级别的Token，即调用获取用户Token接口时，请求body中“auth.scope”的取值需要选择“project”，如下所示。

```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",     //用户名
                    "password": "********",     //登录密码
                    "domain": {
                        "name": "domainname"     //用户所属的账号名称
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxx"     //项目名称
            }
        }
    }
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFJ....”，则调用接口时将“X-Auth-Token: ABCDEFJ....”加到请求消息头即可，如下所示。

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

您还可以通过这个视频教程了解如何使用Token认证：**https://bbs.huaweicloud.com/videos/101333** 。

3.3 返回结果

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见**状态码**。

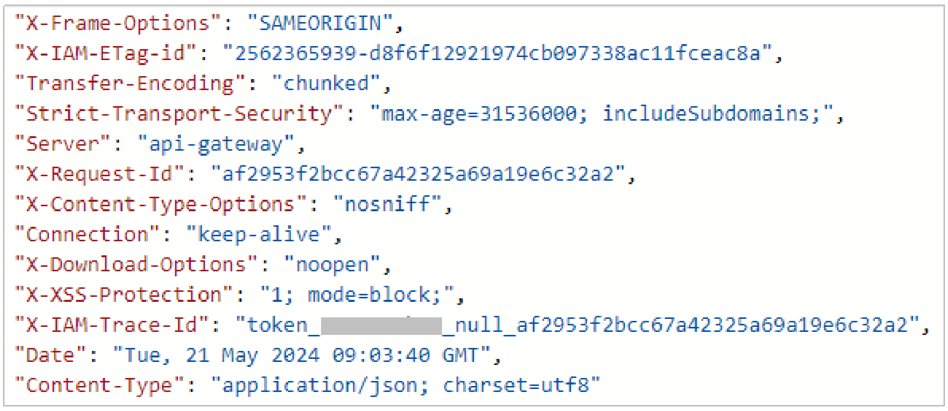
对于**管理员创建IAM用户**接口，如果调用后返回状态码为“201”，则表示请求成功。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于**管理员创建IAM用户**接口，返回如**图3-2**所示的消息头。

图 3-2 管理员创建 IAM 用户响应消息头



```
"X-Frame-Options": "SAMEORIGIN",
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fceac8a",
"Transfer-Encoding": "chunked",
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",
"Server": "api-gateway",
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",
"X-Content-Type-Options": "nosniff",
"Connection": "keep-alive",
"X-Download-Options": "noopen",
"X-XSS-Protection": "1; mode=block;",
"X-IAM-Trace-Id": "token_[redacted]_null_af2953f2bcc67a42325a69a19e6c32a2",
"Date": "Tue, 21 May 2024 09:03:40 GMT",
"Content-Type": "application/json; charset=utf8"
```

响应消息体

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于**管理员创建IAM用户**接口，返回如下消息体。篇幅有限，这里只展示部分内容。

```
{
  "user": {
    "id": "c131886aec...",
    "name": "IAMUser",
    "description": "IAM User Description",
    "areacode": "",
    "phone": "",
    "email": "***@***.com",
    "status": null,
    "enabled": true,
    "pwd_status": false,
    "access_mode": "default",
    "is_domain_owner": false,
    "xuser_id": "",
    "xuser_type": "",
    "password_expires_at": null,
    "create_time": "2024-05-21T09:03:41.000000",
    "domain_id": "d78cbac1..........",
    "xdomain_id": "30086000........",
    "xdomain_type": "",
    "default_project_id": null
  }
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
  "error_msg": "Request body is invalid.",
  "error_code": "IAM.0011"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

4 API

4.1 集群管理接口

4.1.1 创建集群

功能介绍

该接口用于创建集群。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters

表 4-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

请求参数

表 4-2 请求 Body 参数

参数	是否必选	参数类型	描述
cluster	是	**CreateClusterBody** object	集群对象。

表 **4-3** CreateClusterBody

参数	是否必选	参数类型	描述
instance	是	**CreateClusterInstanceBody** object	实例对象。
datastore	是	**CreateClusterDatastoreBody** object	数据搜索引擎类型。
name	是	String	集群名称。4～32个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。
instanceNum	是	Integer	集群实例个数，取值范围为1~32。
backupStrategy	否	**CreateClusterBackupStrategyBody** object	开启自动创建快照策略，默认关闭。 当backupStrategy参数配置不为空时，才会开启自动创建快照策略。
httpsEnable	否	Boolean	设置是否进行通信加密。取值范围为true或false。默认关闭通信加密功能。当httpsEnable设置为true时，authorityEnable字段需要设置为true。 ● true：表示集群进行通信加密。 ● false：表示集群不进行通信加密。 **说明** 此参数只有6.5.4及之后版本支持。
authorityEnable	否	Boolean	是否开启认证，取值范围为true或false。默认关闭认证功能。 ● true：表示集群开启认证。 ● false：表示集群不开启认证。 **说明** 此参数只有6.5.4及之后版本支持。

参数	是否必选	参数类型	描述
adminPwd	否	String	安全模式下集群管理员admin的密码，只有在创建集群时authorityEnable设置为true时需要设置此参数。 ● 管理员密码需要满足规则： – 可输入的字符串长度为8-32个字符。 – 密码至少包含大写字母，小写字母，数字和特殊字符中的三类，其中可输入的特殊字符为：~!@#$%^&*()-_=+\\\|[{}];:,<.>/?。 ● 安全集群的密码会进行弱口令校验，建议设置安全性高的密码。
enterprise_project_id	否	String	企业项目ID。创建集群时，给集群绑定企业项目ID。最大长度36个字符，带"-"连字符的UUID格式，或者是字符串"0"。"0"表示默认企业项目。
tags	否	Array of **CreateClusterTagsBody** objects	集群标签。
payInfo	否	**payInfoBody** object	包周期集群相关参数订购包周期集群时使用。
ipv6_enable	否	Boolean	集群是否开启自动分配IPv6地址，默认值为false。

表 4-4 CreateClusterInstanceBody

参数	是否必选	参数类型	描述
flavorRef	是	String	实例规格名称。可以使用**获取实例规格列表**的name属性确认当前拥有的规格信息。

参数	是否必选	参数类型	描述
volume	是	**CreateClusterInstanceVolumeBody** object	volume信息。当flavorRef选择的是本地盘规格时不需要填写，目前支持的本地盘规格有： ● ess.spec-i3small ● ess.spec-i3medium ● ess.spec-i3.8xlarge.8 ● ess.spec-ds.xlarge.8 ● ess.spec-ds.2xlarge.8 ● ess.spec-ds.4xlarge.8
nics	是	**CreateClusterInstanceNicsBody** object	子网信息。
availability_zone	是	String	可用区。需要指定可用区的名称（可用分区名称）。 默认指定单AZ。指定多AZ时，各个可用分区名称需要使用英文逗号（,）分隔，以“华北-北京四”为例，选择三AZ时，availability_zone取值为cn-north-4a,cn-north-4b,cn-north-4c。如果使用单AZ，availability_zone默认取值为空。 **说明** 说明 - 选择多AZ时，各个可用分区名称不能重复输入，并且要求节点个数大于等于AZ个数。 - 如果节点个数为AZ个数的倍数，节点将会均匀的分布到各个AZ。 - 如果节点个数不为AZ个数的倍数时，各个AZ分布的节点数量之差的绝对值小于等于1。 - 可用分区名称，请在地区和终端节点获取。

表 **4-5** CreateClusterInstanceVolumeBody

参数	是否必选	参数类型	描述
volume_type	是	String	卷类型。 ● COMMON：普通I/O。 ● HIGH：高I/O。 ● ULTRAHIGH：超高I/O。 ● ESSD：极速SSD。
size	是	Integer	卷大小，必须大于0且为4和10的公倍数，磁盘规格大小可以通过**获取实例规格列表**中diskrange属性获得。 单位：GB。

表 **4-6** CreateClusterInstanceNicsBody

参数	是否必选	参数类型	描述
vpcId	是	String	指定虚拟私有云ID，用于集群网络配置。
netId	是	String	子网ID(网络ID)。
securityGroupId	是	String	安全组ID。

表 **4-7** CreateClusterDatastoreBody

参数	是否必选	参数类型	描述
version	是	String	Elasticsearch/Logstash引擎版本号。详细请参考CSS**使用前必读**中支持的版本。
type	是	String	引擎类型，支持elasticsearch、logstash和opensearch。

表 4-8 CreateClusterBackupStrategyBody

参数	是否必选	参数类型	描述
period	是	String	每天创建快照的时刻，只支持整点，后面需加上时区，格式为“HH:mm z”，“HH:mm”表示整点时间，“z”表示时区。比如“00:00 GMT+08:00”、“01:00 GMT+08:00”等。 **说明** 默认值“00:00 GMT+08:00”，当frequency取值HOUR时，无需指定period值，默认从设置成功的下一整点创建。
prefix	是	String	自动创建的快照的前缀，需要用户自己手动输入。只能包含1~32位小写字母、数字、中划线或者下划线，并且以小写字母开头。
keepday	是	Integer	自定义设置快照保留的个数，范围是1～90。系统在半点时刻会自动删除超过保留个数的快照。过期删除策略只针对与当前自动创建快照策略相同执行频次的自动快照。
bucket	否	String	备份使用的OBS桶名称。
basePath	否	String	快照在OBS桶中的存放路径。
agency	否	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。 **说明** 如果bucket、basePath和agency三个参数同时为空，则系统会自动创建OBS桶和IAM代理（若创建失败，则需要手工配置正确的参数）。
frequency	否	String	自动创建快照的执行频次。 **说明** 默认值DAY，取值HOUR时，表示每小时整点执行一次；取值DAY时，表示每天执行一次；取值SUN、MON、TUE、WED、THU、FRI、SAT时，表示在每周对应时间执行一次。比如SUN表示每周日执行一次。保留快照个数与自动创建快照策略中设置的执行频次和索引有关。当执行频次间隔时间短或索引数据量大，保留自动快照可能会达不到设置的个数，请谨慎选择！

表 **4-9** CreateClusterTagsBody

参数	是否必选	参数类型	描述
key	是	String	集群标签的key值。可输入的字符串长度为1~36个字符。只能包含数字、字母、中划线"-"和下划线"_"。
value	是	String	集群标签的value值。可输入的字符串长度为0~43个字符。只能包含数字、字母、中划线"-"和下划线"_"。

表 **4-10** payInfoBody

参数	是否必选	参数类型	描述
payModel	是	Integer	订购周期类型。 ● 2: 包月。 ● 3: 包年。
period	是	Integer	订购周期数。 ● 若payModel为2，则有效值为1-9。 ● 若payModel为3，则有效值为1-3。
isAutoRenew	否	Integer	是否自动续订，为空时表示不自动续订。 ● 1: 自动续订。 ● 0：不自动续订（默认）。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

响应参数

状态码： 200

表 **4-11** 响应 Body 参数

参数	参数类型	描述
cluster	**CreateClusterClusterResponse** object	集群对象。若创建的是按需集群，则只返回cluster参数。
orderId	String	订单号。只有创建包周期集群时才会返回orderId参数。

表 **4-12** CreateClusterClusterResponse

参数	参数类型	描述
id	String	集群ID。
name	String	集群名称。

请求示例

- 创建包周期集群请求示例。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters

{
  "cluster" : {
    "instance" : {
      "flavorRef" : "ess.spec-4u16g",
      "volume" : {
        "volume_type" : "COMMON",
        "size" : 120
      },
      "nics" : {
        "vpcId" : "{VPC ID}",
        "netId" : "{NET ID}",
        "securityGroupId" : "{Security group ID}"
      },
      "availability_zone" : "{AZ CODE}"
    },
    "datastore" : {
      "version" : "{cluster-version}",
      "type" : "elasticsearch"
    },
    "name" : "cluster-name",
    "instanceNum" : 3,
    "backupStrategy" : {
      "period" : "16:00 GMT+08:00",
      "prefix" : "snapshot",
      "keepday" : 7,
      "frequency" : "DAY",
      "bucket" : "css-obs-backup",
      "basePath" : "css_repository/obs-path",
      "agency" : "css_obs_agency"
    },
    "httpsEnable" : true,
    "authorityEnable" : true,
    "adminPwd" : "{password}",
    "enterprise_project_id" : "0",
    "tags" : [ {
      "key" : "k1",
      "value" : "v1"
    }, {
      "key" : "k2",
      "value" : "v2"
    } ],
    "payInfo" : {
      "payModel" : 2,
      "period" : 1,
      "isAutoRenew" : 1,
      "isAutoPay" : 1
    }
  }
}
```

- 创建按需集群请求示例。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters

{
  "cluster" : {
    "instance" : {
      "flavorRef" : "ess.spec-4u16g",
      "volume" : {
        "volume_type" : "COMMON",
        "size" : 120
      },
      "nics" : {
        "vpcId" : "{VPC ID}",
        "netId" : "{NET ID}",
        "securityGroupId" : "{Security group ID}"
      },
      "availability_zone" : "{AZ CODE}"
    },
    "datastore" : {
      "version" : "{cluster-version}",
      "type" : "elasticsearch"
    },
    "name" : "cluster-name",
    "instanceNum" : 3,
    "backupStrategy" : {
      "period" : "16:00 GMT+08:00",
      "prefix" : "snapshot",
      "keepday" : 7,
      "frequency" : "DAY",
      "bucket" : "css-obs-backup",
      "basePath" : "css_repository/obs-path",
      "agency" : "css_obs_agency"
    },
    "httpsEnable" : true,
    "authorityEnable" : true,
    "adminPwd" : "{password}",
    "enterprise_project_id" : "0",
    "tags" : [ {
      "key" : "k1",
      "value" : "v1"
    }, {
      "key" : "k2",
      "value" : "v2"
    } ]
  }
}
```

响应示例

状态码： 200

请求已成功。

- 创建按需期集群响应示例。

```
{
  "cluster" : {
    "id" : "ef683016-871e-48bc-bf93-74a29d60d214",
    "name" : "ES-Test"
  }
}
```

- 创建包周期集群响应示例。

```
{
  "cluster" : {
    "id" : "ef683016-871e-48bc-bf93-74a29d60d214",
    "name" : "ES-Test"
  },
  "orderId" : "CS2208241141LY94J"
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 创建包周期集群请求示例。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateClusterRequest request = new CreateClusterRequest();
        CreateClusterReq body = new CreateClusterReq();
        PayInfoBody payInfoCluster = new PayInfoBody();
        payInfoCluster.withPayModel(2)
            .withPeriod(1)
            .withIsAutoRenew(1)
            .withIsAutoPay(1);
        List<CreateClusterTagsBody> listClusterTags = new ArrayList<>();
        listClusterTags.add(
            new CreateClusterTagsBody()
                .withKey("k1")
                .withValue("v1")
        );
        listClusterTags.add(
            new CreateClusterTagsBody()
                .withKey("k2")
                .withValue("v2")
        );
        CreateClusterBackupStrategyBody backupStrategyCluster = new
CreateClusterBackupStrategyBody();
        backupStrategyCluster.withPeriod("16:00 GMT+08:00")
            .withPrefix("snapshot")
            .withKeepday(7)
            .withBucket("css-obs-backup")
            .withBasePath("css_repository/obs-path")
            .withAgency("css_obs_agency");
        CreateClusterDatastoreBody datastoreCluster = new CreateClusterDatastoreBody();
        datastoreCluster.withVersion("{cluster-version}")
            .withType("elasticsearch");
        CreateClusterInstanceNicsBody nicsInstance = new CreateClusterInstanceNicsBody();
        nicsInstance.withVpcId("{VPC ID}")
            .withNetId("{NET ID}")
            .withSecurityGroupId("{Security group ID}");
        CreateClusterInstanceVolumeBody volumeInstance = new CreateClusterInstanceVolumeBody();
        volumeInstance.withVolumeType("COMMON")
            .withSize(120);
        CreateClusterInstanceBody instanceCluster = new CreateClusterInstanceBody();
        instanceCluster.withFlavorRef("ess.spec-4u16g")
            .withVolume(volumeInstance)
            .withNics(nicsInstance)
            .withAvailabilityZone("{AZ CODE}");
        CreateClusterBody clusterbody = new CreateClusterBody();
        clusterbody.withInstance(instanceCluster)
            .withDatastore(datastoreCluster)
            .withName("cluster-name")
            .withInstanceNum(3)
            .withBackupStrategy(backupStrategyCluster)
            .withHttpsEnable(true)
            .withAuthorityEnable(true)
            .withAdminPwd("{password}")
            .withEnterpriseProjectId("0")
            .withTags(listClusterTags)
            .withPayInfo(payInfoCluster);
        body.withCluster(clusterbody);
        request.withBody(body);
        try {
            CreateClusterResponse response = client.createCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
```

```
    }
}
```

- 创建按需集群请求示例。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateClusterRequest request = new CreateClusterRequest();
        CreateClusterReq body = new CreateClusterReq();
        List<CreateClusterTagsBody> listClusterTags = new ArrayList<>();
        listClusterTags.add(
            new CreateClusterTagsBody()
                .withKey("k1")
                .withValue("v1")
        );
        listClusterTags.add(
            new CreateClusterTagsBody()
                .withKey("k2")
                .withValue("v2")
        );
        CreateClusterBackupStrategyBody backupStrategyCluster = new CreateClusterBackupStrategyBody();
        backupStrategyCluster.withPeriod("16:00 GMT+08:00")
            .withPrefix("snapshot")
            .withKeepday(7)
            .withBucket("css-obs-backup")
            .withBasePath("css_repository/obs-path")
            .withAgency("css_obs_agency");
        CreateClusterDatastoreBody datastoreCluster = new CreateClusterDatastoreBody();
        datastoreCluster.withVersion("{cluster-version}")
            .withType("elasticsearch");
        CreateClusterInstanceNicsBody nicsInstance = new CreateClusterInstanceNicsBody();
        nicsInstance.withVpcId("{VPC ID}")
            .withNetId("{NET ID}")
            .withSecurityGroupId("{Security group ID}");
        CreateClusterInstanceVolumeBody volumeInstance = new CreateClusterInstanceVolumeBody();
        volumeInstance.withVolumeType("COMMON")
```

```
            .withSize(120);
        CreateClusterInstanceBody instanceCluster = new CreateClusterInstanceBody();
        instanceCluster.withFlavorRef("ess.spec-4u16g")
            .withVolume(volumeInstance)
            .withNics(nicsInstance)
            .withAvailabilityZone("{AZ CODE}");
        CreateClusterBody clusterbody = new CreateClusterBody();
        clusterbody.withInstance(instanceCluster)
            .withDatastore(datastoreCluster)
            .withName("cluster-name")
            .withInstanceNum(3)
            .withBackupStrategy(backupStrategyCluster)
            .withHttpsEnable(true)
            .withAuthorityEnable(true)
            .withAdminPwd("{password}")
            .withEnterpriseProjectId("0")
            .withTags(listClusterTags);
        body.withCluster(clusterbody);
        request.withBody(body);
        try {
            CreateClusterResponse response = client.createCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 创建包周期集群请求示例。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        payInfoCluster = PayInfoBody(
            pay_model=2,
            period=1,
            is_auto_renew=1,
            is_auto_pay=1
        )
        listTagsCluster = [
            CreateClusterTagsBody(
                key="k1",
                value="v1"
            ),
            CreateClusterTagsBody(
                key="k2",
                value="v2"
            )
        ]
        backupStrategyCluster = CreateClusterBackupStrategyBody(
            period="16:00 GMT+08:00",
            prefix="snapshot",
            keepday=7,
            bucket="css-obs-backup",
            base_path="css_repository/obs-path",
            agency="css_obs_agency"
        )
        datastoreCluster = CreateClusterDatastoreBody(
            version="{cluster-version}",
            type="elasticsearch"
        )
        nicsInstance = CreateClusterInstanceNicsBody(
            vpc_id="{VPC ID}",
            net_id="{NET ID}",
            security_group_id="{Security group ID}"
        )
        volumeInstance = CreateClusterInstanceVolumeBody(
            volume_type="COMMON",
            size=120
        )
        instanceCluster = CreateClusterInstanceBody(
            flavor_ref="ess.spec-4u16g",
            volume=volumeInstance,
            nics=nicsInstance,
            availability_zone="{AZ CODE}"
        )
        clusterbody = CreateClusterBody(
            instance=instanceCluster,
            datastore=datastoreCluster,
            name="cluster-name",
            instance_num=3,
            backup_strategy=backupStrategyCluster,
            https_enable=True,
            authority_enable=True,
            admin_pwd="{password}",
            enterprise_project_id="0",
            tags=listTagsCluster,
            pay_info=payInfoCluster
        )
        request.body = CreateClusterReq(
            cluster=clusterbody
        )
        response = client.create_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 创建按需集群请求示例。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        listTagsCluster = [
            CreateClusterTagsBody(
                key="k1",
                value="v1"
            ),
            CreateClusterTagsBody(
                key="k2",
                value="v2"
            )
        ]
        backupStrategyCluster = CreateClusterBackupStrategyBody(
            period="16:00 GMT+08:00",
            prefix="snapshot",
            keepday=7,
            bucket="css-obs-backup",
            base_path="css_repository/obs-path",
            agency="css_obs_agency"
        )
        datastoreCluster = CreateClusterDatastoreBody(
            version="{cluster-version}",
            type="elasticsearch"
        )
        nicsInstance = CreateClusterInstanceNicsBody(
            vpc_id="{VPC ID}",
            net_id="{NET ID}",
            security_group_id="{Security group ID}"
        )
        volumeInstance = CreateClusterInstanceVolumeBody(
            volume_type="COMMON",
            size=120
        )
        instanceCluster = CreateClusterInstanceBody(
            flavor_ref="ess.spec-4u16g",
            volume=volumeInstance,
            nics=nicsInstance,
            availability_zone="{AZ CODE}"
        )
        clusterbody = CreateClusterBody(
            instance=instanceCluster,
            datastore=datastoreCluster,
            name="cluster-name",
            instance_num=3,
            backup_strategy=backupStrategyCluster,
            https_enable=True,
            authority_enable=True,
            admin_pwd="{password}",
            enterprise_project_id="0",
            tags=listTagsCluster
        )
        request.body = CreateClusterReq(
            cluster=clusterbody
        )
        response = client.create_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 创建包周期集群请求示例。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{}
    isAutoRenewPayInfo:= int32(1)
    isAutoPayPayInfo:= int32(1)
    payInfoCluster := &model.PayInfoBody{
        PayModel: int32(2),
        Period: int32(1),
        IsAutoRenew: &isAutoRenewPayInfo,
        IsAutoPay: &isAutoPayPayInfo,
    }
    var listTagsCluster = []model.CreateClusterTagsBody{
        {
            Key: "k1",
            Value: "v1",
        },
        {
            Key: "k2",
            Value: "v2",
        },
    }
    bucketBackupStrategy:= "css-obs-backup"
    basePathBackupStrategy:= "css_repository/obs-path"
    agencyBackupStrategy:= "css_obs_agency"
    backupStrategyCluster := &model.CreateClusterBackupStrategyBody{
        Period: "16:00 GMT+08:00",
        Prefix: "snapshot",
        Keepday: int32(7),
        Bucket: &bucketBackupStrategy,
        BasePath: &basePathBackupStrategy,
        Agency: &agencyBackupStrategy,
    }
    datastoreCluster := &model.CreateClusterDatastoreBody{
        Version: "{cluster-version}",
        Type: "elasticsearch",
    }
    nicsInstance := &model.CreateClusterInstanceNicsBody{
        VpcId: "{VPC ID}",
        NetId: "{NET ID}",
        SecurityGroupId: "{Security group ID}",
    }
    volumeInstance := &model.CreateClusterInstanceVolumeBody{
        VolumeType: "COMMON",
        Size: int32(120),
    }
    availabilityZoneInstance:= "{AZ CODE}"
    instanceCluster := &model.CreateClusterInstanceBody{
        FlavorRef: "ess.spec-4u16g",
        Volume: volumeInstance,
        Nics: nicsInstance,
        AvailabilityZone: &availabilityZoneInstance,
    }
    httpsEnableCluster:= true
    authorityEnableCluster:= true
    adminPwdCluster:= "{password}"
    enterpriseProjectIdCluster:= "0"
    clusterbody := &model.CreateClusterBody{
        Instance: instanceCluster,
        Datastore: datastoreCluster,
        Name: "cluster-name",
        InstanceNum: int32(3),
        BackupStrategy: backupStrategyCluster,
        HttpsEnable: &httpsEnableCluster,
        AuthorityEnable: &authorityEnableCluster,
        AdminPwd: &adminPwdCluster,
        EnterpriseProjectId: &enterpriseProjectIdCluster,
        Tags: &listTagsCluster,
        PayInfo: payInfoCluster,
    }
    request.Body = &model.CreateClusterReq{
        Cluster: clusterbody,
    }
    response, err := client.CreateCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建按需集群请求示例。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{}
    var listTagsCluster = []model.CreateClusterTagsBody{
        {
            Key: "k1",
            Value: "v1",
        },
        {
            Key: "k2",
            Value: "v2",
        },
    }
    bucketBackupStrategy:= "css-obs-backup"
    basePathBackupStrategy:= "css_repository/obs-path"
    agencyBackupStrategy:= "css_obs_agency"
    backupStrategyCluster := &model.CreateClusterBackupStrategyBody{
        Period: "16:00 GMT+08:00",
        Prefix: "snapshot",
        Keepday: int32(7),
        Bucket: &bucketBackupStrategy,
        BasePath: &basePathBackupStrategy,
        Agency: &agencyBackupStrategy,
    }
    datastoreCluster := &model.CreateClusterDatastoreBody{
        Version: "{cluster-version}",
        Type: "elasticsearch",
    }
    nicsInstance := &model.CreateClusterInstanceNicsBody{
        VpcId: "{VPC ID}",
        NetId: "{NET ID}",
        SecurityGroupId: "{Security group ID}",
    }
    volumeInstance := &model.CreateClusterInstanceVolumeBody{
        VolumeType: "COMMON",
        Size: int32(120),
    }
    availabilityZoneInstance:= "{AZ CODE}"
    instanceCluster := &model.CreateClusterInstanceBody{
        FlavorRef: "ess.spec-4u16g",
        Volume: volumeInstance,
        Nics: nicsInstance,
        AvailabilityZone: &availabilityZoneInstance,
    }
    httpsEnableCluster:= true
    authorityEnableCluster:= true
    adminPwdCluster:= "{password}"
    enterpriseProjectIdCluster:= "0"
    clusterbody := &model.CreateClusterBody{
        Instance: instanceCluster,
        Datastore: datastoreCluster,
        Name: "cluster-name",
        InstanceNum: int32(3),
        BackupStrategy: backupStrategyCluster,
        HttpsEnable: &httpsEnableCluster,
        AuthorityEnable: &authorityEnableCluster,
        AdminPwd: &adminPwdCluster,
        EnterpriseProjectId: &enterpriseProjectIdCluster,
        Tags: &listTagsCluster,
    }
    request.Body = &model.CreateClusterReq{
        Cluster: clusterbody,
    }
    response, err := client.CreateCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.2 查询集群列表

功能介绍

该接口用于查询并显示集群列表以及集群的状态。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters

表 4-13 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。

表 4-14 Query 参数

参数	是否必选	参数类型	描述
start	否	Integer	指定查询起始值，默认值为1，即从第1个集群开始查询。
limit	否	Integer	指定查询个数，默认值为10，即一次查询10个集群信息。
datastoreType	否	String	集群引擎类型。

请求参数

无

响应参数

状态码： 200

表 4-15 响应 Body 参数

参数	参数类型	描述
totalSize	Integer	集群个数。
clusters	Array of **ClusterList** objects	集群对象列表。

表 **4-16** ClusterList

参数	参数类型	描述
datastore	**ClusterListDatastore** object	数据搜索引擎类型。
instances	Array of **ClusterListInstances** objects	节点对象列表。
publicKibanaResp	**publicKibanaRespBody** object	Kibana公网访问信息。
elbWhiteList	**elbWhiteListResp** object	弹性IP白名单列表。
updated	String	集群上次修改时间，格式为ISO8601： CCYY-MM-DDThh:mm:ss。
name	String	集群名称。
publicIp	String	公网IP信息。
created	String	集群创建时间，格式为ISO8601：CCYY-MM-DDThh:mm:ss。 返回的集群列表信息按照创建时间降序排序，即创建时间最新的集群排在最前。
id	String	集群ID。
status	String	集群状态值。 ● 100：创建中。 ● 200：可用。 ● 303：不可用，如创建失败。
endpoint	String	集群内网访问IPv4地址和端口号。
vpcId	String	VPC ID。
subnetId	String	子网ID。
securityGroupId	String	安全组ID。
bandwidthSize	Integer	公网带宽大小。单位：Mbit/s
httpsEnable	Boolean	通信加密状态。 ● false：未设置通信加密。 ● true：已设置通信加密。

参数	参数类型	描述
authorityEnable	Boolean	是否开启认证。 ● true：表示集群开启认证。 ● false：表示集群不开启认证。
diskEncrypted	Boolean	磁盘是否加密。 ● true：磁盘已加密。 ● false：磁盘未加密。
backupAvailable	Boolean	快照是否开启。 ● true: 快照开启状态。 ● false: 快照关闭状态。
actionProgress	Object	集群行为进度，显示创建或扩容进度的百分比等。CREATING表示创建的百分比。
actions	Array of strings	集群当前行为。REBOOTING表示重启、GROWING表示扩容、RESTORING表示恢复集群、SNAPSHOTTING表示创建快照等。
enterpriseProjectId	String	集群所属的企业项目ID。 如果集群所属用户没有开通企业项目，则不会返回该参数。
tags	Array of **ClusterListTags** objects	集群标签。
failedReason	**ClusterListFailedReasons** object	失败原因。如果集群处于正常状态，则不返回回该参数。
period	Boolean	是否为包周期集群。 ● "true" 表示是包周期计费的集群。 ● "false" 表示是按需计费的集群。
bandwidthResourceId	String	es公网访问的资源id。
ipv6Endpoint	String	集群内网访问IPv6地址和端口号。

表 **4-17** ClusterListDatastore

参数	参数类型	描述
type	String	引擎类型，支持elasticsearch，logstash。
version	String	CSS集群引擎版本号。详细请参考CSS**使用前必读**中支持的版本。

参数	参数类型	描述
supportSecuritymode	Boolean	是否支持安全模式。
subVersion	String	集群镜像发布版本号。
isEosCluster	Boolean	表示集群镜像发布版本是否EOS，true表示过期，false表示不过期。

表 4-18 ClusterListInstances

参数	参数类型	描述
status	String	节点状态值。 ● 100：创建中。 ● 200：可用。 ● 303：不可用，如创建失败。
type	String	当前节点的类型。
id	String	实例ID。
name	String	实例名字。
specCode	String	节点规格名称。
azCode	String	节点所属AZ信息。
ip	String	实例ip信息。
volume	**ClusterVolumeRsp** object	实例磁盘信息。
resourceId	String	该实例对应的资源Id。

表 4-19 ClusterVolumeRsp

参数	参数类型	描述
type	String	实例磁盘类型。
size	Integer	实例磁盘大小。
resourceIds	Array of strings	该实例拥有的磁盘对应的资源Id。

表 **4-20** publicKibanaRespBody

参数	参数类型	描述
eipSize	Integer	带宽大小。单位：Mbit/s
elbWhiteListResp	**kibanaElbWhiteListResp** object	Kibana公网访问信息。
publicKibanaIp	String	kibana访问IP。
bandwidthResourceId	String	当前kibana公网对应的资源Id。

表 **4-21** kibanaElbWhiteListResp

参数	参数类型	描述
enableWhiteList	Boolean	是否开启kibana访问控制。 ● true: 开启访问控制。 ● false: 关闭访问控制
whiteList	String	kibana公网访问白名单。

表 **4-22** elbWhiteListResp

参数	参数类型	描述
enableWhiteList	Boolean	是否开启公网访问控制。 ● true: 开启公网访问控制。 ● false: 关闭公网访问控制。
whiteList	String	公网访问白名单。

表 **4-23** ClusterListTags

参数	参数类型	描述
key	String	集群标签的key值。
value	String	集群标签的value值。

表 **4-24** ClusterListFailedReasons

参数	参数类型	描述
errorCode	String	错误码。 ● CSS.6000：表示集群创建失败。 ● CSS.6001：表示集群扩容失败。 ● CSS.6002：表示集群重启失败。 ● CSS.6004：表示集群节点创建失败。 ● CSS.6005：表示服务初始化失败。
errorMsg	String	详细错误信息。

请求示例

- 查询前2个集群

```
- 方法1：GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters?start=1&limit=2

- 方法2：GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters?limit=2
```

- 查询前10个集群

```
- 方法1：GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters?start=1&limit=10

- 方法2：GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters?limit=10
```

响应示例

状态码： 200

请求已成功。

```
{
  "totalSize" : 1,
  "clusters" : [ {
    "datastore" : {
      "type" : "elasticsearch",
      "version" : "7.10.2",
      "subVersion" : "7.10.2_24.3.0_0827",
      "isEosCluster" : false,
      "supportSecuritymode" : false
    },
    "instances" : [ {
      "status" : "200",
      "type" : "ess",
      "id" : "{INSTANCE_ID}",
      "name" : "css-8bc5-ess-esn-1-1",
      "specCode" : "ess.spec-4u8g",
      "azCode" : "{AZ_CODE}",
      "volume" : {
        "type" : "ULTRAHIGH",
        "size" : 40,
        "resourceIds" : [ "{RESOURCE_ID}" ]
      },
      "ip" : "192.168.0.122",
      "resourceId" : "{RESOURCE_ID}"
    } ],
    "publicKibanaResp" : {
      "eipSize" : 10,
      "publicKibanaIp" : "100.95.152.28:9200",
      "elbWhiteListResp" : null,
      "bandwidthResourceId" : "18bec13f-5cc1-4631-867f-33505d15be12"
    },
    "elbWhiteList" : {
      "whiteList" : "",
      "enableWhiteList" : false
    },
    "updated" : "2023-10-09T02:07:13",
    "name" : "css-8bc5",
    "publicIp" : "100.85.222.202",
    "created" : "2023-10-09T02:07:13",
    "id" : "{CLUSTER_ID}",
    "status" : "200",
    "endpoint" : "192.168.0.122:9200",
    "vpcId" : "{VPC_ID}",
    "subnetId" : "{SUBNET_ID}",
    "securityGroupId" : "{SECURITY_GROUP_ID}",
    "bandwidthResourceId" : "{BANDWIDTH_RESOURCE_ID}",
    "bandwidthSize" : 3,
    "httpsEnable" : true,
    "authorityEnable" : true,
    "diskEncrypted" : false,
    "backupAvailable" : false,
    "actionProgress" : { },
    "actions" : [ ],
    "enterpriseProjectId" : "0",
    "tags" : [ ],
    "period" : true
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListClustersDetailsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListClustersDetailsRequest request = new ListClustersDetailsRequest();
        try {
            ListClustersDetailsResponse response = client.listClustersDetails(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListClustersDetailsRequest()
        response = client.list_clusters_details(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListClustersDetailsRequest{}
    response, err := client.ListClustersDetails(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.3 查询集群详情

功能介绍

该接口用于查询并显示单个集群详情。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}

表 4-25 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

无

响应参数

状态码： 200

表 4-26 响应 Body 参数

参数	参数类型	描述
datastore	**ClusterDetailDatastore** object	数据搜索引擎类型。
instances	Array of **ClusterDetailInstances** objects	节点对象列表。
publicKibanaResp	**publicKibanaRespBody** object	Kibana公网访问信息。
elbWhiteList	**elbWhiteListResp** object	公网访问信息。
updated	String	集群上次修改时间，格式为ISO8601： CCYY-MM-DDThh:mm:ss。
name	String	集群名称。
publicIp	String	公网IP和端口信息。
created	String	集群创建时间，格式为ISO8601： CCYY-MM-DDThh:mm:ss。

参数	参数类型	描述
id	String	集群ID。
status	String	集群状态值。 ● 100：操作进行中，如创建中。 ● 200：可用。 ● 303：不可用，如创建失败。
endpoint	String	集群内网访问IPv4地址和端口号。
vpcId	String	VPC ID。
subnetId	String	子网ID。
securityGroupId	String	安全组ID。
vpcepIp	String	终端节点IP。
vpcepIpv6Address	String	终端节点IPv6。
bandwidthSize	Integer	公网带宽大小。单位：Mbit/s
httpsEnable	Boolean	通信加密状态。 ● false：未设置通信加密。 ● true：已设置通信加密。
diskEncrypted	Boolean	磁盘是否加密。 ● true：磁盘已加密。 ● false：磁盘未加密。
authorityEnable	Boolean	是否开启认证，取值范围为true或false。默认关闭认证功能。 ● true：表示集群开启认证。 ● false：表示集群不开启认证。
backupAvailable	Boolean	快照是否开启。 ● true: 快照开启状态。 ● false: 快照关闭状态。
actionProgress	Object	集群行为进度，显示创建或扩容进度的百分比。
actions	Array of strings	集群当前行为。REBOOTING表示重启、GROWING表示扩容、RESTORING表示恢复集群、SNAPSHOTTING表示创建快照等。
enterpriseProjectId	String	集群所属的企业项目ID。 如果集群所属用户没有开通企业项目，则不会返回该参数。

参数	参数类型	描述
tags	Array of **ClusterDetailTags** objects	集群标签。
failedReason	**ClusterDetailFailedReasons** object	失败原因。如果集群处于正常状态，则不返回该参数。
period	Boolean	是否为包周期集群。 ● "true" 表示是包周期计费的集群。 ● "false" 表示是按需计费的集群。
bandwidthResourceId	String	公网访问的资源id。
ipv6Endpoint	String	集群内网访问IPv6地址和端口号。
supportVpcep	Boolean	当前集群是否支持开启终端节点服务。

表 **4-27** ClusterDetailDatastore

参数	参数类型	描述
type	String	引擎类型，支持elasticsearch和logstash。
version	String	CSS集群引擎版本号。详细请参考CSS**使用前必读**中支持的版本。
supportSecuritymode	Boolean	支持安全模式
subVersion	String	集群镜像发布版本号。
isEosCluster	Boolean	表示集群镜像发布版本是否EOS，true表示过期，false表示不过期。

表 **4-28** ClusterDetailInstances

参数	参数类型	描述
status	String	节点状态值。 ● 100：操作进行中，如创建中。 ● 200：可用。 ● 303：不可用，如创建失败。
resourceId	String	该实例对应的资源Id。
type	String	当前节点的类型。

参数	参数类型	描述
id	String	实例ID。
name	String	实例名字。
specCode	String	节点规格名称。
azCode	String	节点所属AZ信息。
privateIp	String	实例ip信息。
volume	**ShowClusterVolumeRsp** object	实例磁盘信息。

表 **4-29** ShowClusterVolumeRsp

参数	参数类型	描述
type	String	实例磁盘类型。
size	Integer	实例磁盘大小。
resourceIds	Array of strings	该实例拥有的磁盘对应的资源Id。

表 **4-30** publicKibanaRespBody

参数	参数类型	描述
eipSize	Integer	带宽大小。单位：Mbit/s
elbWhiteListResp	**kibanaElbWhiteListResp** object	Kibana公网访问信息。
publicKibanaIp	String	kibana访问IP。
bandwidthResourceId	String	当前kibana公网对应的资源Id。

表 **4-31** kibanaElbWhiteListResp

参数	参数类型	描述
enableWhiteList	Boolean	是否开启kibana访问控制。 ● true: 开启访问控制。 ● false: 关闭访问控制

参数	参数类型	描述
whiteList	String	kibana公网访问白名单。

表 **4-32** elbWhiteListResp

参数	参数类型	描述
enableWhiteList	Boolean	是否开启公网访问控制。 ● true: 开启公网访问控制。 ● false: 关闭公网访问控制。
whiteList	String	公网访问白名单。

表 **4-33** ClusterDetailTags

参数	参数类型	描述
key	String	集群标签的key值。
value	String	集群标签的value值。

表 **4-34** ClusterDetailFailedReasons

参数	参数类型	描述
errorCode	String	错误码。 ● CSS.6000：表示集群创建失败。 ● CSS.6001：表示集群扩容失败。 ● CSS.6002：表示集群重启失败。 ● CSS.6004：表示集群节点创建失败。 ● CSS.6005：表示服务初始化失败。
errorMsg	String	详细错误信息。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "datastore" : {
    "type" : "elasticsearch",
    "version" : "7.10.2",
    "subVersion" : "7.10.2_24.3.0_0827",
    "isEosCluster" : false,
    "supportSecuritymode" : false
  },
  "instances" : [ {
    "status" : "200",
    "type" : "ess",
    "id" : "{INSTANCE_ID}",
    "name" : "css-8bc5-ess-esn-1-1",
    "specCode" : "ess.spec-4u8g",
    "azCode" : "{AZ_CODE}",
    "volume" : {
      "type" : "ULTRAHIGH",
      "size" : 40,
      "resourceIds" : [ "{RESOURCE_ID}" ]
    },
    "privateIp" : "192.168.0.122",
    "resourceId" : "{RESOURCE_ID}"
  } ],
  "publicKibanaResp" : {
    "eipSize" : 10,
    "publicKibanaIp" : "100.95.152.28",
    "elbWhiteListResp" : null,
    "bandwidthResourceId" : "18bec13f-5cc1-4631-867f-33505d15be12"
  },
  "elbWhiteList" : {
    "whiteList" : "",
    "enableWhiteList" : false
  },
  "updated" : "2023-10-09T02:07:13",
  "name" : "css-8bc5",
  "publicIp" : "100.85.222.202",
  "created" : "2023-10-09T02:07:13",
  "id" : "{CLUSTER_ID}",
  "status" : "200",
  "endpoint" : "192.168.0.122:9200",
  "vpcId" : "{VPC_ID}",
  "subnetId" : "{SUBNET_ID}",
  "securityGroupId" : "{SECURITY_GROUP_ID}",
  "bandwidthResourceId" : "{BANDWIDTH_RESOURCE_ID}",
  "bandwidthSize" : 3,
  "httpsEnable" : true,
  "authorityEnable" : true,
  "diskEncrypted" : false,
  "backupAvailable" : false,
  "actionProgress" : { },
  "actions" : [ ],
  "enterpriseProjectId" : "0",
  "tags" : [ ],
  "period" : true,
  "supportVpcep" : true
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowClusterDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowClusterDetailRequest request = new ShowClusterDetailRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowClusterDetailResponse response = client.showClusterDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterDetailRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_cluster_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterDetailRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowClusterDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.4 删除集群

功能介绍

此接口用于删除集群。集群删除将释放此集群的所有资源，包括客户数据。如果需要保留客户集群数据，建议在删除集群前先创建快照。

说明

此接口亦可用于包年/包月集群退订。公安冻结的集群不能删除。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}

表 4-35 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定删除集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteClusterRequest request = new DeleteClusterRequest();
        request.withClusterId("{cluster_id}");
        try {
            DeleteClusterResponse response = client.deleteCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.delete_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteClusterRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.DeleteCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.5 按需集群转包周期

功能介绍

该接口用于按需集群转包周期集群。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/cluster/{cluster_id}/period

表 **4-36** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待转包周期的集群ID。

请求参数

表 **4-37** 请求 Body 参数

参数	是否必选	参数类型	描述
periodType	是	Integer	订购周期类型。 ● 2：包月。 ● 3：包年。
periodNum	是	Integer	订购周期数。 ● 若选择包月（参数范围：1-9）。 ● 若选择包年（参数范围：1-3）。
isAutoRenew	否	Integer	是否自动续订，为空时表示不自动续订 ● 1：自动续订。 ● 0：不自动续订（默认）。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

参数	是否必选	参数类型	描述
consoleURL	否	String	云服务ConsoleURL。 订购订单支付完成后，客户可以通过此URL跳转到云服务Console页面查看信息。（仅手动支付时涉及）。

响应参数

状态码： 200

表 **4-38** 响应 Body 参数

参数	参数类型	描述
orderId	String	订单ID（此订单类型为“新购”）。

请求示例

修改集群的计费模式，由按需计费模式转为包周期计费模式。

```
{
  "periodType" : 2,
  "periodNum" : 1,
  "isAutoRenew" : 1,
  "isAutoPay" : 0,
  "consoleURL" : "https://console.xxx.huawei.com/elasticsearch-pre/?locale=zh-cn&region=xxx#/es/management/elasticsearch"
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "orderId" : "CS2112281600EBHKR"
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改集群的计费模式，由按需计费模式转为包周期计费模式。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateOndemandClusterToPeriodSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateOndemandClusterToPeriodRequest request = new UpdateOndemandClusterToPeriodRequest();
        request.withClusterId("{cluster_id}");
        PeriodReq body = new PeriodReq();
        body.withConsoleURL("https://console.xxx.huawei.com/elasticsearch-pre/?locale=zh-
cn&region=xxx#/es/management/elasticsearch");
        body.withIsAutoPay(0);
        body.withIsAutoRenew(1);
        body.withPeriodNum(1);
        body.withPeriodType(2);
        request.withBody(body);
        try {
            UpdateOndemandClusterToPeriodResponse response = 
client.updateOndemandClusterToPeriod(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改集群的计费模式，由按需计费模式转为包周期计费模式。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

```
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateOndemandClusterToPeriodRequest()
        request.cluster_id = "{cluster_id}"
        request.body = PeriodReq(
            console_url="https://console.xxx.huawei.com/elasticsearch-pre/?locale=zh-cn&region=xxx#/es/
management/elasticsearch",
            is_auto_pay=0,
            is_auto_renew=1,
            period_num=1,
            period_type=2
        )
        response = client.update_ondemand_cluster_to_period(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改集群的计费模式，由按需计费模式转为包周期计费模式。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
    request := &model.UpdateOndemandClusterToPeriodRequest{}
    request.ClusterId = "{cluster_id}"
    consoleURLPeriodReq:= "https://console.xxx.huawei.com/elasticsearch-pre/?locale=zh-cn&region=xxx#/es/
management/elasticsearch"
    isAutoPayPeriodReq:= int32(0)
    isAutoRenewPeriodReq:= int32(1)
    request.Body = &model.PeriodReq{
        ConsoleURL: &consoleURLPeriodReq,
        IsAutoPay: &isAutoPayPeriodReq,
        IsAutoRenew: &isAutoRenewPeriodReq,
        PeriodNum: int32(1),
        PeriodType: int32(2),
    }
    response, err := client.UpdateOndemandClusterToPeriod(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	身份认证失败。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

错误码

请参见**错误码**。

4.1.6 修改集群名称

功能介绍

该接口用于修改集群名称。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/changename

表 4-39 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改集群名称的集群ID。

请求参数

表 4-40 请求 Body 参数

参数	是否必选	参数类型	描述
displayName	是	String	修改后集群名称。

响应参数

无

请求示例

修改当前集群的展示名称。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
changename

{
  "displayName" : "ES-Test-new"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群的展示名称。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateClusterNameSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateClusterNameRequest request = new UpdateClusterNameRequest();
        request.withClusterId("{cluster_id}");
        UpdateClusterNameReq body = new UpdateClusterNameReq();
        body.withDisplayName("ES-Test-new");
        request.withBody(body);
        try {
            UpdateClusterNameResponse response = client.updateClusterName(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群的展示名称。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterNameRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdateClusterNameReq(
            display_name="ES-Test-new"
        )
        response = client.update_cluster_name(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群的展示名称。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateClusterNameRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.UpdateClusterNameReq{
        DisplayName: "ES-Test-new",
    }
    response, err := client.UpdateClusterName(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.7 修改密码

功能介绍

该接口用于修改集群密码。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/password/reset

表 4-41 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待修改集群密码的集群ID。

请求参数

表 **4-42** 请求 Body 参数

参数	是否必选	参数类型	描述
newpassword	是	String	安全模式下集群管理员admin的密码，只有在创建集群时authorityEnable设置为true时该接口可用。 ● 参数范围：8~32个字符。 ● 参数要求：密码至少包含大写字母，小写字母，数字、特殊字符四类中的三类，其中可输入的特殊字符为：~!@#$%&*()-_=|[{}];:,<.>/?

响应参数

无

请求示例

修改当前集群的管理员密码。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
password/reset

{
  "newpassword" : "xxxxxx"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群的管理员密码。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;
```

```
public class ResetPasswordSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ResetPasswordRequest request = new ResetPasswordRequest();
        request.withClusterId("{cluster_id}");
        ResetPasswordReq body = new ResetPasswordReq();
        body.withNewpassword("xxxxxx");
        request.withBody(body);
        try {
            ResetPasswordResponse response = client.resetPassword(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群的管理员密码。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ResetPasswordRequest()
        request.cluster_id = "{cluster_id}"
        request.body = ResetPasswordReq(
            newpassword="xxxxxx"
        )
        response = client.reset_password(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群的管理员密码。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ResetPasswordRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.ResetPasswordReq{
        Newpassword: "xxxxxx",
    }
    response, err := client.ResetPassword(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.8 重启集群

功能介绍

此接口用于重启集群，重启集群将导致业务中断。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/restart

表 4-43 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定重启集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class RestartClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        RestartClusterRequest request = new RestartClusterRequest();
        request.withClusterId("{cluster_id}");
        try {
            RestartClusterResponse response = client.restartCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RestartClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.restart_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RestartClusterRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.RestartCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.9 扩容集群

功能介绍

该接口用于集群扩容实例（仅支持扩容elasticsearch实例）。只扩容普通节点，且只针对要扩容的集群实例不存在特殊节点（Master、Client、冷数据节点）的情况。集群扩容实例的数量和存储容量，请参考**扩容实例的数量和存储容量**。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/extend

表 4-44 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待扩容的集群ID。

请求参数

表 4-45 请求 Body 参数

参数	是否必选	参数类型	描述
grow	是	**ExtendClusterGrowReq** object	集群扩容请求详细描述。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 4-46 ExtendClusterGrowReq

参数	是否必选	参数类型	描述
modifySize	是	Integer	扩容实例个数。 集群已有实例个数和扩容实例个数总和不能超过32。

响应参数

无

请求示例

修改当前集群的实例个数。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/extend

{
  "grow" : {
    "modifySize" : 4
  },
  "isAutoPay" : 1
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群的实例个数。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateExtendClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateExtendClusterRequest request = new UpdateExtendClusterRequest();
        request.withClusterId("{cluster_id}");
        ExtendClusterReq body = new ExtendClusterReq();
        ExtendClusterGrowReq growbody = new ExtendClusterGrowReq();
        growbody.withModifySize(4);
        body.withIsAutoPay(1);
        body.withGrow(growbody);
        request.withBody(body);
        try {
            UpdateExtendClusterResponse response = client.updateExtendCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群的实例个数。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateExtendClusterRequest()
        request.cluster_id = "{cluster_id}"
        growbody = ExtendClusterGrowReq(
            modify_size=4
        )
        request.body = ExtendClusterReq(
            is_auto_pay=1,
            grow=growbody
        )
        response = client.update_extend_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群的实例个数。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateExtendClusterRequest{}
    request.ClusterId = "{cluster_id}"
    growbody := &model.ExtendClusterGrowReq{
        ModifySize: int32(4),
    }
    isAutoPayExtendClusterReq:= int32(1)
    request.Body = &model.ExtendClusterReq{
        IsAutoPay: &isAutoPayExtendClusterReq,
        Grow: growbody,
    }
    response, err := client.UpdateExtendCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。

状态码	描述
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.10 扩容实例的数量和存储容量

功能介绍

该接口用于集群扩容不同类型实例的个数以及存储容量。已经存在独立Master、Client、冷数据节点的集群使用该接口扩容。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/role_extend

表 4-47 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待扩容的集群ID。

请求参数

表 4-48 请求 Body 参数

参数	是否必选	参数类型	描述
grow	是	Array of **RoleExtendGrowReq** objects	集群扩容请求详细描述。

参数	是否必选	参数类型	描述
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 4-49 RoleExtendGrowReq

参数	是否必选	参数类型	描述
type	是	String	扩容实例类型。取值为ess、ess-master或ess-client，可以选择其中一个或多个之间的组合但不可以重复选择。 ess-master、ess-client节点只支持增加实例个数。
nodesize	是	Integer	扩容实例个数。集群已有实例个数和增加实例个数总和不能超过32。若无需扩容该参数将该参数设置为0即可。
disksize	是	Integer	扩容实例存储容量。集群原实例存储容量和扩容实例存储容量之和不能超过创建集群时对应默认实例存储容量上限。若无需扩容该参数将该参数设置为0即可。当集群为包周期集群时不支持同时修改节点个数和磁盘容量。 单位：GB。 ● ess节点、ess-cold节点扩容步长为20。 ● ess-master节点、ess-client节点不允许扩容存储。

响应参数

状态码： 200

表 4-50 响应 Body 参数

参数	参数类型	描述
id	String	集群ID。

请求示例

修改当前集群的实例个数与存储容量。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
role_extend

{
  "grow" : [ {
    "type" : "ess-master",
    "nodesize" : 2,
    "disksize" : 0
  }, {
    "type" : "ess",
    "nodesize" : 0,
    "disksize" : 40
  }, {
    "type" : "ess-client",
    "nodesize" : 1,
    "disksize" : 0
  } ],
  "isAutoPay" : 1
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "id" : "4f3deec3-efa8-4598-bf91-560aad1377a3"
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群的实例个数与存储容量。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateExtendInstanceStorageSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateExtendInstanceStorageRequest request = new UpdateExtendInstanceStorageRequest();
        request.withClusterId("{cluster_id}");
        RoleExtendReq body = new RoleExtendReq();
        List<RoleExtendGrowReq> listbodyGrow = new ArrayList<>();
        listbodyGrow.add(
            new RoleExtendGrowReq()
                .withType("ess-master")
                .withNodesize(2)
                .withDisksize(0)
        );
        listbodyGrow.add(
            new RoleExtendGrowReq()
                .withType("ess")
                .withNodesize(0)
                .withDisksize(40)
        );
        listbodyGrow.add(
            new RoleExtendGrowReq()
                .withType("ess-client")
                .withNodesize(1)
                .withDisksize(0)
        );
        body.withIsAutoPay(1);
        body.withGrow(listbodyGrow);
        request.withBody(body);
        try {
            UpdateExtendInstanceStorageResponse response = client.updateExtendInstanceStorage(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群的实例个数与存储容量。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateExtendInstanceStorageRequest()
        request.cluster_id = "{cluster_id}"
        listGrowbody = [
            RoleExtendGrowReq(
                type="ess-master",
                nodesize=2,
                disksize=0
            ),
            RoleExtendGrowReq(
                type="ess",
                nodesize=0,
                disksize=40
            ),
            RoleExtendGrowReq(
                type="ess-client",
                nodesize=1,
                disksize=0
            )
        ]
        request.body = RoleExtendReq(
            is_auto_pay=1,
            grow=listGrowbody
        )
        response = client.update_extend_instance_storage(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群的实例个数与存储容量。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateExtendInstanceStorageRequest{}
    request.ClusterId = "{cluster_id}"
    var listGrowbody = []model.RoleExtendGrowReq{
        {
            Type: "ess-master",
            Nodesize: int32(2),
            Disksize: int32(0),
        },
        {
            Type: "ess",
            Nodesize: int32(0),
            Disksize: int32(40),
        },
        {
            Type: "ess-client",
            Nodesize: int32(1),
            Disksize: int32(0),
        },
    }
    isAutoPayRoleExtendReq:= int32(1)
    request.Body = &model.RoleExtendReq{
        IsAutoPay: &isAutoPayRoleExtendReq,
        Grow: listGrowbody,
    }
    response, err := client.UpdateExtendInstanceStorage(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.11 变更规格

功能介绍

该接口用于变更集群规格。只支持变更ess节点类型。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/flavor

表 4-51 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改规格的集群ID。

请求参数

表 4-52 请求 Body 参数

参数	是否必选	参数类型	描述
newFlavorId	是	String	operationType为vm时，newFlavorId为变更后节点规格ID。 该参数通过 该参数通过获取实例规格列表接口获取，根据name属性所需要的规格，选择对应的flavor_id。 仅支持同一个Elasticsearch引擎版本下的节点规格变更。 operationType为volume时，newFlavorId为变更后磁盘类型，支持如下类型： ● COMMON: 普通I/O ● HIGH: 高I/O ● ULTRAHIGH: 超高I/O ● ESSD: 极速SSD
operationType	否	String	操作类型。可选值：vm和volume，默认为vm。 ● vm: 修改节点规格 ● volume: 修改磁盘规格
needCheckReplica	否	Boolean	是否需要检查副本，取值范围为true或false。默认开启校验。 ● true: 开启副本校验。 ● false: 忽略副本校验。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

参数	是否必选	参数类型	描述
needCheckClusterStatus	否	Boolean	描述：是否需要检查集群状态，取值范围为true或false。默认开启校验。 ● true: 开启集群状态校验。 ● false: 忽略集群状态校验。
clusterLoadCheck	否	Boolean	描述：是否需要检查集群负载，取值范围为true或false。默认开启校验。 ● true: 开启集群负载校验。 ● false: 忽略集群负载校验。

响应参数

无

请求示例

-
  ```
  POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/flavor

  {
    "needCheckReplica" : false,
    "newFlavorId" : "35b060a4-f152-48ce-8773-36559ceb81f2",
    "isAutoPay" : 1,
    "needCheckClusterStatus" : true
  }
  ```

-
  ```
  POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/flavor

  {
    "needCheckReplica": false,
    "newFlavorId": "ULTRAHIGH",
    "operationType":   "volume",
    "isAutoPay": 1,
    "clusterLoadCheck": true
  }
  ```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

-
  ```
  package com.huaweicloud.sdk.test;

  import com.huaweicloud.sdk.core.auth.ICredential;
  import com.huaweicloud.sdk.core.auth.BasicCredentials;
  import com.huaweicloud.sdk.core.exception.ConnectionException;
  ```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateFlavorSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateFlavorRequest request = new UpdateFlavorRequest();
        request.withClusterId("{cluster_id}");
        UpdateFlavorReq body = new UpdateFlavorReq();
        body.withIsAutoPay(1);
        body.withNewFlavorId("35b060a4-f152-48ce-8773-36559ceb81f2");
        body.withNeedCheckReplica(false);
        request.withBody(body);
        try {
            UpdateFlavorResponse response = client.updateFlavor(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

-
  ```
  package com.huaweicloud.sdk.test;

  import com.huaweicloud.sdk.core.auth.ICredential;
  import com.huaweicloud.sdk.core.auth.BasicCredentials;
  import com.huaweicloud.sdk.core.exception.ConnectionException;
  import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
  import com.huaweicloud.sdk.core.exception.ServiceResponseException;
  import com.huaweicloud.sdk.css.v1.region.CssRegion;
  import com.huaweicloud.sdk.css.v1.*;
  import com.huaweicloud.sdk.css.v1.model.*;


  public class UpdateFlavorSolution {

      public static void main(String[] args) {
          // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
  ```

```
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateFlavorRequest request = new UpdateFlavorRequest();
        request.withClusterId("{cluster_id}");
        UpdateFlavorReq body = new UpdateFlavorReq();
        body.withIsAutoPay(1);
        body.withNewFlavorId("ULTRAHIGH");
        body.withNeedCheckReplica(false);
        request.withBody(body);
        try {
            UpdateFlavorResponse response = client.updateFlavor(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

-
```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()
    try:
        request = UpdateFlavorRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdateFlavorReq(
            is_auto_pay=1,
            new_flavor_id="35b060a4-f152-48ce-8773-36559ceb81f2",
            need_check_replica=False
        )
        response = client.update_flavor(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

-
  ```
  # coding: utf-8

  import os
  from huaweicloudsdkcore.auth.credentials import BasicCredentials
  from huaweicloudsdkcss.v1.region.css_region import CssRegion
  from huaweicloudsdkcore.exceptions import exceptions
  from huaweicloudsdkcss.v1 import *

  if __name__ == "__main__":
      # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
      # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
      ak = os.environ["CLOUD_SDK_AK"]
      sk = os.environ["CLOUD_SDK_SK"]
      projectId = "{project_id}"

      credentials = BasicCredentials(ak, sk, projectId)

      client = CssClient.new_builder() \
          .with_credentials(credentials) \
          .with_region(CssRegion.value_of("<YOUR REGION>")) \
          .build()

      try:
          request = UpdateFlavorRequest()
          request.cluster_id = "{cluster_id}"
          request.body = UpdateFlavorReq(
              is_auto_pay=1,
              new_flavor_id="ULTRAHIGH",
              need_check_replica=False
          )
          response = client.update_flavor(request)
          print(response)
      except exceptions.ClientRequestException as e:
          print(e.status_code)
          print(e.request_id)
          print(e.error_code)
          print(e.error_msg)
  ```

Go

-
  ```
  package main

  import (
      "fmt"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
      css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
      region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
  )
  ```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateFlavorRequest{}
    request.ClusterId = "{cluster_id}"
    isAutoPayUpdateFlavorReq:= int32(1)
    needCheckReplicaUpdateFlavorReq:= false
    request.Body = &model.UpdateFlavorReq{
        IsAutoPay: &isAutoPayUpdateFlavorReq,
        NewFlavorId: "35b060a4-f152-48ce-8773-36559ceb81f2",
        NeedCheckReplica: &needCheckReplicaUpdateFlavorReq,
    }
    response, err := client.UpdateFlavor(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

-
  ```
  package main

  import (
      "fmt"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
      css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
      region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
  )

  func main() {
      // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
      // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
      ak := os.Getenv("CLOUD_SDK_AK")
      sk := os.Getenv("CLOUD_SDK_SK")
      projectId := "{project_id}"

      auth := basic.NewCredentialsBuilder().
          WithAk(ak).
          WithSk(sk).
          WithProjectId(projectId).
          Build()

      client := css.NewCssClient(
          css.CssClientBuilder().
              WithRegion(region.ValueOf("<YOUR REGION>")).
  ```

```
            WithCredential(auth).
            Build())

    request := &model.UpdateFlavorRequest{}
    request.ClusterId = "{cluster_id}"
    isAutoPayUpdateFlavorReq:= int32(1)
    needCheckReplicaUpdateFlavorReq:= false
    request.Body = &model.UpdateFlavorReq{
        IsAutoPay: &isAutoPayUpdateFlavorReq,
        NewFlavorId: "ULTRAHIGH",
        NeedCheckReplica: &needCheckReplicaUpdateFlavorReq,
    }
    response, err := client.UpdateFlavor(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.12 获取实例规格列表

功能介绍

该接口用于查询并显示支持的实例规格对应的ID。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/es-flavors

表 4-53 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

请求参数

无

响应参数

状态码： 200

表 4-54 响应 Body 参数

参数	参数类型	描述
versions	Array of **EsflavorsVersionsResp** objects	引擎版本列表。

表 4-55 EsflavorsVersionsResp

参数	参数类型	描述
version	String	Elasticsearch引擎版本号。详细请参考CSS**使用前必读**中支持的版本。
flavors	Array of **EsflavorsVersionsFlavorsResp** objects	规格列表。
type	String	实例类型，包括为ess、ess-cold、ess-master和ess-client。

表 4-56 EsflavorsVersionsFlavorsResp

参数	参数类型	描述
cpu	Integer	实例的CPU核数。

参数	参数类型	描述
ram	Integer	实例的内存大小。单位GB。
name	String	规格名称。
region	String	可用region。
diskrange	String	实例的硬盘容量范围。
availableAZ	String	可用区。
flavor_id	String	规格对应的ID。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "versions" : [ {
    "flavors" : [ {
      "cpu" : 4,
      "ram" : 32,
      "name" : "ess.spec-4u32g",
      "region" : "xx-xxx-xx",
      "diskrange" : "40,2560",
      "availableAZ" : "xx-xxx-xx,xx-xxx-xx",
      "flavor_id" : "2d8daf1b-873f-4c2e-a7b9-2f9cbcf2f213"
    }, {
      "cpu" : 8,
      "ram" : 64,
      "name" : "ess.spec-8u64g",
      "region" : "xx-xxx-xx",
      "diskrange" : "80,5120",
      "availableAZ" : "xx-xxx-xx,xx-xxx-xx",
      "flavor_id" : "b3d33ec6-d58a-40f0-aa51-4f671ce64b2a"
    }, {
      "cpu" : 16,
      "ram" : 128,
      "name" : "ess.spec-16u128g",
      "region" : "xx-xxx-xx",
      "diskrange" : "160,10240",
      "availableAZ" : "xx-xxx-xx,xx-xxx-xx",
      "flavor_id" : "f74419ca-bc91-4558-b4e2-90eeefb37c6e"
    } ],
    "type" : "ess",
    "version" : "x.x.x"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListFlavorsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListFlavorsRequest request = new ListFlavorsRequest();
        try {
            ListFlavorsResponse response = client.listFlavors(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListFlavorsRequest()
        response = client.list_flavors(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListFlavorsRequest{}
    response, err := client.ListFlavors(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.13 查询所有标签

功能介绍

该接口用于查询指定region下的所有标签集合。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/{resource_type}/tags

表 4-57 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
resource_type	是	String	资源类型，当前固定值为“css-cluster”，表示是集群类型。

请求参数

无

响应参数

状态码： 200

表 **4-58** 响应 Body 参数

参数	参数类型	描述
tags	Array of **ShowAllTagsTagsResp** objects	集群的标签列表。

表 **4-59** ShowAllTagsTagsResp

参数	参数类型	描述
key	String	集群标签的key值。
values	Array of strings	集群标签的value值列表。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  }, {
    "key" : "key2",
    "values" : [ "value1", "value2" ]
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListClustersTagsSolution {
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListClustersTagsRequest request = new ListClustersTagsRequest();
        request.withResourceType("{resource_type}");
        try {
            ListClustersTagsResponse response = client.listClustersTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListClustersTagsRequest()
        request.resource_type = "{resource_type}"
        response = client.list_clusters_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListClustersTagsRequest{}
    request.ResourceType = "{resource_type}"
    response, err := client.ListClustersTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。

状态码	描述
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见错误码。

4.1.14 查询指定集群的标签

功能介绍

该接口用于查询指定集群的标签信息。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/{resource_type}/{cluster_id}/tags

表 4-60 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定待查询的集群ID。
resource_type	是	String	资源类型，当前固定值为“css-cluster”，表示是集群类型。

请求参数

无

响应参数

状态码： 200

表 **4-61** 响应 Body 参数

参数	参数类型	描述
tags	Array of **ShowTagsTagsResp** objects	集群标签列表。

表 **4-62** ShowTagsTagsResp

参数	参数类型	描述
key	String	集群标签的key值。
value	String	集群标签的value值。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value3"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowClusterTagSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowClusterTagRequest request = new ShowClusterTagRequest();
        request.withClusterId("{cluster_id}");
        request.withResourceType("{resource_type}");
        try {
            ShowClusterTagResponse response = client.showClusterTag(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterTagRequest()
        request.cluster_id = "{cluster_id}"
        request.resource_type = "{resource_type}"
        response = client.show_cluster_tag(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterTagRequest{}
    request.ClusterId = "{cluster_id}"
    request.ResourceType = "{resource_type}"
    response, err := client.ShowClusterTag(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。

状态码	描述
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.15 添加指定集群标签

功能介绍

该接口用于给指定集群添加标签。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/{resource_type}/{cluster_id}/tags

表 4-63 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待添加标签的集群ID。
resource_type	是	String	资源类型，当前固定值为“css-cluster”，表示是集群类型。

请求参数

表 4-64 请求 Body 参数

参数	是否必选	参数类型	描述
tag	是	**Tag** object	待添加的标签对象。

表 **4-65** Tag

参数	是否必选	参数类型	描述
key	是	String	标签名。
value	是	String	标签值。

响应参数

无

请求示例

创建集群标签。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/css-cluster/4f3deec3-efa8-4598-bf91-560aad1377a3/tags

{
  "tag" : {
    "key" : "K1",
    "value" : "V1"
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

创建集群标签。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateClustersTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateClustersTagsRequest request = new CreateClustersTagsRequest();
        request.withClusterId("{cluster_id}");
        request.withResourceType("{resource_type}");
        TagReq body = new TagReq();
        Tag tagbody = new Tag();
        tagbody.withKey("K1")
            .withValue("V1");
        body.withTag(tagbody);
        request.withBody(body);
        try {
            CreateClustersTagsResponse response = client.createClustersTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

创建集群标签。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClustersTagsRequest()
        request.cluster_id = "{cluster_id}"
        request.resource_type = "{resource_type}"
        tagbody = Tag(
            key="K1",
            value="V1"
        )
        request.body = TagReq(
            tag=tagbody
        )
        response = client.create_clusters_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建集群标签。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClustersTagsRequest{}
    request.ClusterId = "{cluster_id}"
    request.ResourceType = "{resource_type}"
    tagbody := &model.Tag{
        Key: "K1",
        Value: "V1",
    }
    request.Body = &model.TagReq{
        Tag: tagbody,
    }
    response, err := client.CreateClustersTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.16 删除集群标签

功能介绍

此接口用于删除集群标签。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/{resource_type}/{cluster_id}/tags/{key}

表 4-66 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定删除标签的集群ID。
resource_type	是	String	资源类型，当前固定值为“css-cluster”，表示是集群类型。

参数	是否必选	参数类型	描述
key	是	String	需要删除的标签名。如果标签名为中文需要经过url编码再调用该接口。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteClustersTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteClustersTagsRequest request = new DeleteClustersTagsRequest();
        request.withClusterId("{cluster_id}");
        request.withResourceType("{resource_type}");
        request.withKey("{key}");
        try {
            DeleteClustersTagsResponse response = client.deleteClustersTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteClustersTagsRequest()
        request.cluster_id = "{cluster_id}"
        request.resource_type = "{resource_type}"
        request.key = "{key}"
        response = client.delete_clusters_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteClustersTagsRequest{}
    request.ClusterId = "{cluster_id}"
    request.ResourceType = "{resource_type}"
    request.Key = "{key}"
    response, err := client.DeleteClustersTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.17 批量添加或删除集群标签

功能介绍

该接口用于对集群批量添加或删除标签。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/{resource_type}/{cluster_id}/tags/action

表 4-67 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定批量添加标签的集群ID。
resource_type	是	String	资源类型，当前固定值为“css-cluster”，表示是集群类型。

请求参数

表 4-68 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	操作类型。通过该属性标识当前所需的操作类型。 ● create：批量添加标签。 ● delete：批量删除标签。
tags	是	Array of **Tag** objects	标签列表。
sysTags	否	Array of **SysTags** objects	系统标签列表。 ● key固定为：_sys_enterprise_project_id。 ● value是UUID或0,value为0表示默认企业项目。

表 **4-69** Tag

参数	是否必选	参数类型	描述
key	是	String	标签名。
value	是	String	标签值。

表 **4-70** SysTags

参数	是否必选	参数类型	描述
key	是	String	标签名。
value	是	String	标签值。

响应参数

无

请求示例

批量创建或删除集群标签。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/css-cluster/4f3deec3-efa8-4598-bf91-560aad1377a3/
tags/action

{
  "action" : "create",
  "tags" : [ {
    "key" : "K1",
    "value" : "V1"
  }, {
    "key" : "K2",
    "value" : "V2"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

批量创建或删除集群标签。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateBatchClustersTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateBatchClustersTagsRequest request = new UpdateBatchClustersTagsRequest();
        request.withClusterId("{cluster_id}");
        request.withResourceType("{resource_type}");
        BatchAddOrDeleteTagOnClusterReq body = new BatchAddOrDeleteTagOnClusterReq();
        List<Tag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new Tag()
                .withKey("K1")
                .withValue("V1")
        );
        listbodyTags.add(
            new Tag()
                .withKey("K2")
                .withValue("V2")
        );
        body.withTags(listbodyTags);
        body.withAction("create");
        request.withBody(body);
        try {
            UpdateBatchClustersTagsResponse response = client.updateBatchClustersTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

批量创建或删除集群标签。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateBatchClustersTagsRequest()
        request.cluster_id = "{cluster_id}"
        request.resource_type = "{resource_type}"
        listTagsbody = [
            Tag(
                key="K1",
                value="V1"
            ),
            Tag(
                key="K2",
                value="V2"
            )
        ]
        request.body = BatchAddOrDeleteTagOnClusterReq(
            tags=listTagsbody,
            action="create"
        )
        response = client.update_batch_clusters_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

批量创建或删除集群标签。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateBatchClustersTagsRequest{}
    request.ClusterId = "{cluster_id}"
    request.ResourceType = "{resource_type}"
    var listTagsbody = []model.Tag{
        {
            Key: "K1",
            Value: "V1",
        },
        {
            Key: "K2",
            Value: "V2",
        },
    }
    request.Body = &model.BatchAddOrDeleteTagOnClusterReq{
        Tags: listTagsbody,
        Action: "create",
    }
    response, err := client.UpdateBatchClustersTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.18 指定节点缩容

功能介绍

该接口可以对集群现有节点中指定节点进行缩容。包周期类型的集群不支持通过api进行指定节点缩容操作。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/node/offline

表 4-71 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待缩容的集群ID。

请求参数

表 4-72 请求 Body 参数

参数	是否必选	参数类型	描述
migrate_data	否	String	是否迁移数据。默认为"true"。 ● "true"：迁移数据。 ● "false"：不迁移数据。
shrinkNodes	是	Array of strings	需要缩容的节点ID。 通过**查询集群详情**获取instances中的id属性。

响应参数

无

请求示例

缩容集群，通过指定节点方式缩容。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/node/offline

{
  "shrinkNodes" : [ "2077bdf3-b90d-412e-b460-635b9b159c11" ],
  "migrate_data" : "true"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

缩容集群，通过指定节点方式缩容。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateShrinkNodesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateShrinkNodesRequest request = new UpdateShrinkNodesRequest();
        request.withClusterId("{cluster_id}");
        ShrinkNodesReq body = new ShrinkNodesReq();
        List<String> listbodyShrinkNodes = new ArrayList<>();
        listbodyShrinkNodes.add("2077bdf3-b90d-412e-b460-635b9b159c11");
        body.withShrinkNodes(listbodyShrinkNodes);
        body.withMigrateData("true");
        request.withBody(body);
        try {
            UpdateShrinkNodesResponse response = client.updateShrinkNodes(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

缩容集群，通过指定节点方式缩容。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateShrinkNodesRequest()
        request.cluster_id = "{cluster_id}"
        listShrinkNodesbody = [
            "2077bdf3-b90d-412e-b460-635b9b159c11"
        ]
        request.body = ShrinkNodesReq(
            shrink_nodes=listShrinkNodesbody,
            migrate_data="true"
        )
        response = client.update_shrink_nodes(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

缩容集群，通过指定节点方式缩容。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateShrinkNodesRequest{}
    request.ClusterId = "{cluster_id}"
    var listShrinkNodesbody = []string{
        "2077bdf3-b90d-412e-b460-635b9b159c11",
    }
    migrateDataShrinkNodesReq:= "true"
    request.Body = &model.ShrinkNodesReq{
        ShrinkNodes: listShrinkNodesbody,
        MigrateData: &migrateDataShrinkNodesReq,
    }
    response, err := client.UpdateShrinkNodes(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.19 指定节点类型规格变更

功能介绍

修改集群规格。支持修改:

- ess： 数据节点。
- ess-cold: 冷数据节点。
- ess-client: Client节点。
- ess-master: Master节点。
- lgs：Lgs节点。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/{types}/flavor

表 4-73 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改规格的集群ID。
types	是	String	指定待更改的集群节点类型。 ● ess：数据节点。 ● ess-cold：冷数据节点。 ● ess-client：Client节点。 ● ess-master：Master节点。 ● lgs：Lgs节点。

请求参数

表 4-74 请求 Body 参数

参数	是否必选	参数类型	描述
newFlavorId	是	String	operationType为vm时，newFlavorId为变更后节点规格ID。 该参数通过获取实例规格列表接口获取，根据name属性所需要的规格，选择对应的flavor_id。 仅支持同一个Elasticsearch引擎版本下的节点规格变更。 operationType为volume时，newFlavorId为变更后磁盘类型，支持如下类型： ● COMMON: 普通I/O ● HIGH: 高I/O ● ULTRAHIGH: 超高I/O ● ESSD: 极速SSD
operationType	否	String	操作类型。可选值：vm和volume，默认为vm。 ● vm: 修改节点规格 ● volume: 修改磁盘规格
needCheckReplica	否	Boolean	是否需要检查副本，取值范围为true或false。默认开启校验。 ● true: 开启副本校验。 ● false: 忽略副本校验。 **说明** Master和Client节点不是数据节点，因此不需要进行副本校验。即使选择true，也不会进行副本校验。

参数	是否必选	参数类型	描述
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。
needCheckClusterStatus	否	Boolean	是否需要检查集群状态，取值范围为true或false。默认开启校验。 ● true: 开启集群状态校验。 ● false: 忽略集群状态校验。
clusterLoadCheck	否	Boolean	是否需要检查集群负载，取值范围为true或false。默认开启校验。 ● true: 开启集群负载校验。 ● false: 忽略集群负载校验。

响应参数

无

请求示例

-
  ```
  POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ess/flavor

  {
    "needCheckReplica" : false,
    "newFlavorId" : "35b060a4-f152-48ce-8773-36559ceb81f2",
    "isAutoPay" : 1,
    "needCheckClusterStatus" : true
  }
  ```

-
  ```
  POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ess/flavor

  {
    "needCheckReplica": false,
    "newFlavorId": "ULTRAHIGH",
  ```

```
  "operationType"：  "volume",
  "isAutoPay": 1,
  "clusterLoadCheck": true
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

-
```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateFlavorByTypeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateFlavorByTypeRequest request = new UpdateFlavorByTypeRequest();
        request.withClusterId("{cluster_id}");
        request.withTypes("{types}");
        UpdateFlavorByTypeReq body = new UpdateFlavorByTypeReq();
        body.withIsAutoPay(1);
        body.withNewFlavorId("35b060a4-f152-48ce-8773-36559ceb81f2");
        body.withNeedCheckReplica(false);
        request.withBody(body);
        try {
            UpdateFlavorByTypeResponse response = client.updateFlavorByType(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

-
  ```
  package com.huaweicloud.sdk.test;

  import com.huaweicloud.sdk.core.auth.ICredential;
  import com.huaweicloud.sdk.core.auth.BasicCredentials;
  import com.huaweicloud.sdk.core.exception.ConnectionException;
  import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
  import com.huaweicloud.sdk.core.exception.ServiceResponseException;
  import com.huaweicloud.sdk.css.v1.region.CssRegion;
  import com.huaweicloud.sdk.css.v1.*;
  import com.huaweicloud.sdk.css.v1.model.*;


  public class UpdateFlavorByTypeSolution {

      public static void main(String[] args) {
          // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
          // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
          String ak = System.getenv("CLOUD_SDK_AK");
          String sk = System.getenv("CLOUD_SDK_SK");
          String projectId = "{project_id}";

          ICredential auth = new BasicCredentials()
                  .withProjectId(projectId)
                  .withAk(ak)
                  .withSk(sk);

          CssClient client = CssClient.newBuilder()
                  .withCredential(auth)
                  .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                  .build();
          UpdateFlavorByTypeRequest request = new UpdateFlavorByTypeRequest();
          request.withClusterId("{cluster_id}");
          request.withTypes("{types}");
          UpdateFlavorByTypeReq body = new UpdateFlavorByTypeReq();
          body.withIsAutoPay(1);
          body.withNewFlavorId("ULTRAHIGH");
          body.withNeedCheckReplica(false);
          request.withBody(body);
          try {
              UpdateFlavorByTypeResponse response = client.updateFlavorByType(request);
              System.out.println(response.toString());
          } catch (ConnectionException e) {
              e.printStackTrace();
          } catch (RequestTimeoutException e) {
              e.printStackTrace();
          } catch (ServiceResponseException e) {
              e.printStackTrace();
              System.out.println(e.getHttpStatusCode());
              System.out.println(e.getRequestId());
              System.out.println(e.getErrorCode());
              System.out.println(e.getErrorMsg());
          }
      }
  }
  ```

Python

-
  ```
  # coding: utf-8
  ```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateFlavorByTypeRequest()
        request.cluster_id = "{cluster_id}"
        request.types = "{types}"
        request.body = UpdateFlavorByTypeReq(
            is_auto_pay=1,
            new_flavor_id="35b060a4-f152-48ce-8773-36559ceb81f2",
            need_check_replica=False
        )
        response = client.update_flavor_by_type(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

-
  ```
  # coding: utf-8

  import os
  from huaweicloudsdkcore.auth.credentials import BasicCredentials
  from huaweicloudsdkcss.v1.region.css_region import CssRegion
  from huaweicloudsdkcore.exceptions import exceptions
  from huaweicloudsdkcss.v1 import *

  if __name__ == "__main__":
      # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
  security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
  environment variables and decrypted during use to ensure security.
      # In this example, AK and SK are stored in environment variables for authentication. Before
  running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
  environment
      ak = os.environ["CLOUD_SDK_AK"]
      sk = os.environ["CLOUD_SDK_SK"]
      projectId = "{project_id}"

      credentials = BasicCredentials(ak, sk, projectId)

      client = CssClient.new_builder() \
          .with_credentials(credentials) \
          .with_region(CssRegion.value_of("<YOUR REGION>")) \
          .build()

      try:
          request = UpdateFlavorByTypeRequest()
          request.cluster_id = "{cluster_id}"
  ```

```
        request.types = "{types}"
        request.body = UpdateFlavorByTypeReq(
            is_auto_pay=1,
            new_flavor_id="ULTRAHIGH",
            need_check_replica=False
        )
        response = client.update_flavor_by_type(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

-
  ```
  package main

  import (
      "fmt"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
      css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
      region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
  )

  func main() {
      // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
      // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
      ak := os.Getenv("CLOUD_SDK_AK")
      sk := os.Getenv("CLOUD_SDK_SK")
      projectId := "{project_id}"

      auth := basic.NewCredentialsBuilder().
          WithAk(ak).
          WithSk(sk).
          WithProjectId(projectId).
          Build()

      client := css.NewCssClient(
          css.CssClientBuilder().
              WithRegion(region.ValueOf("<YOUR REGION>")).
              WithCredential(auth).
              Build())

      request := &model.UpdateFlavorByTypeRequest{}
      request.ClusterId = "{cluster_id}"
      request.Types = "{types}"
      isAutoPayUpdateFlavorByTypeReq:= int32(1)
      needCheckReplicaUpdateFlavorByTypeReq:= false
      request.Body = &model.UpdateFlavorByTypeReq{
          IsAutoPay: &isAutoPayUpdateFlavorByTypeReq,
          NewFlavorId: "35b060a4-f152-48ce-8773-36559ceb81f2",
          NeedCheckReplica: &needCheckReplicaUpdateFlavorByTypeReq,
      }
      response, err := client.UpdateFlavorByType(request)
      if err == nil {
          fmt.Printf("%+v\n", response)
      } else {
          fmt.Println(err)
      }
  }
  ```

-
  ```
  package main

  import (
  ```

```
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateFlavorByTypeRequest{}
    request.ClusterId = "{cluster_id}"
    request.Types = "{types}"
    isAutoPayUpdateFlavorByTypeReq:= int32(1)
    needCheckReplicaUpdateFlavorByTypeReq:= false
    request.Body = &model.UpdateFlavorByTypeReq{
        IsAutoPay: &isAutoPayUpdateFlavorByTypeReq,
        NewFlavorId: "ULTRAHIGH",
        NeedCheckReplica: &needCheckReplicaUpdateFlavorByTypeReq,
    }
    response, err := client.UpdateFlavorByType(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.20 指定节点类型缩容

功能介绍

该接口用于集群对不同类型实例的个数以及存储容量进行缩容。包周期类型的集群不支持通过api进行指定节点类型缩容操作。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/extend/{project_id}/clusters/{cluster_id}/role/shrink

表 4-75 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待缩容的集群ID。

请求参数

表 4-76 请求 Body 参数

参数	是否必选	参数类型	描述
shrink	是	Array of **ShrinkNodeReq** objects	需要缩容的节点类型和数量集合。

表 4-77 ShrinkNodeReq

参数	是否必选	参数类型	描述
reducedNodeNum	是	Integer	需要缩容的节点数量。 ● 对节点缩容后，请确保每个节点类型在每个AZ中的数量至少为1。 ● 关于跨AZ的集群，在不同AZ中同类型节点个数的差值要小于等于1。 ● 关于没有Master节点的集群，每次缩容的数据节点个数(包含冷数据节点和其他类型节点)要小于当前数据节点总数的一半，缩容后的数据节点个数要大于索引的最大副本个数。 ● 关于有Master节点的集群，每次缩容的Master节点个数要小于当前Master节点总数的一半，缩容后的Master节点个数必须是奇数且不小于3。
type	是	String	指定节点类型。 ● ess：数据节点。 ● ess-cold：冷数据节点。 ● ess-client：Client节点。 ● ess-master：Master节点。

响应参数

无

请求示例

缩容集群，通过指定当前节点类型缩容。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/role/
shrink

{
  "shrink" : [ {
    "type" : "ess",
    "reducedNodeNum" : 1
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

缩容集群，通过指定当前节点类型缩容。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateShrinkClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateShrinkClusterRequest request = new UpdateShrinkClusterRequest();
        request.withClusterId("{cluster_id}");
        ShrinkClusterReq body = new ShrinkClusterReq();
        List<ShrinkNodeReq> listbodyShrink = new ArrayList<>();
        listbodyShrink.add(
            new ShrinkNodeReq()
                .withReducedNodeNum(1)
                .withType("ess")
        );
        body.withShrink(listbodyShrink);
        request.withBody(body);
        try {
            UpdateShrinkClusterResponse response = client.updateShrinkCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

缩容集群，通过指定当前节点类型缩容。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateShrinkClusterRequest()
        request.cluster_id = "{cluster_id}"
        listShrinkbody = [
            ShrinkNodeReq(
                reduced_node_num=1,
                type="ess"
            )
        ]
        request.body = ShrinkClusterReq(
            shrink=listShrinkbody
        )
        response = client.update_shrink_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

缩容集群，通过指定当前节点类型缩容。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateShrinkClusterRequest{}
    request.ClusterId = "{cluster_id}"
    var listShrinkbody = []model.ShrinkNodeReq{
        {
            ReducedNodeNum: int32(1),
            Type: "ess",
        },
    }
    request.Body = &model.ShrinkClusterReq{
        Shrink: listShrinkbody,
    }
    response, err := client.UpdateShrinkCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.1.21 下载安全证书

功能介绍

该接口用于下载安全证书。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/cer/download

表 4-78 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。

请求参数

无

响应参数

状态码： 200

表 4-79 响应 Body 参数

参数	参数类型	描述
-	File	文件流

请求示例

无

响应示例

状态码： 200

请求已成功，返回文件流。

```
"Certificate:xxxx"
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DownloadCertSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DownloadCertRequest request = new DownloadCertRequest();
        try {
            DownloadCertResponse response = client.downloadCert(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DownloadCertRequest()
        response = client.download_cert(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DownloadCertRequest{}
    response, err := client.DownloadCert(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功，返回文件流。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.22 节点替换

功能介绍

该接口用于替换失败节点。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/instance/{instance_id}/replace

表 4-80 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定替换集群ID。
instance_id	是	String	指定替换节点ID。

表 **4-81** Query 参数

参数	是否必选	参数类型	描述
migrateData	否	String	是否迁移数据。默认为"true"。 ● "true"：迁移数据。 ● "false"：不迁移数据。
agency	否	String	委托名称，委托给CSS服务。

请求参数

无

响应参数

无

请求示例

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/instance/
43e63449-339c-4280-a6e9-da36b0685995/replace?migrateData=true
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
```

```
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceRequest request = new UpdateInstanceRequest();
        request.withClusterId("{cluster_id}");
        request.withInstanceId("{instance_id}");
        try {
            UpdateInstanceResponse response = client.updateInstance(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceRequest()
        request.cluster_id = "{cluster_id}"
        request.instance_id = "{instance_id}"
        response = client.update_instance(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateInstanceRequest{}
    request.ClusterId = "{cluster_id}"
    request.InstanceId = "{instance_id}"
    response, err := client.UpdateInstance(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见错误码。

4.1.23 安全模式修改

功能介绍

该接口用于切换集群的安全模式。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/mode/change

表 4-82 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定修改的集群ID。

请求参数

表 4-83 请求 Body 参数

参数	是否必选	参数类型	描述
authorityEnable	是	Boolean	是否开启安全模式。 ● true: 开启安全模式。 ● false: 关闭安全模式。 默认为：true。
adminPwd	否	String	安全模式下集群密码。
httpsEnable	是	Boolean	是否开启HTTPS。 ● true: 开启HTTPS。 ● false: 关闭HTTPS。 默认为：true。

响应参数

无

请求示例

修改当前集群安全模式。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/mode/
change

{
  "authorityEnable" : true,
  "adminPwd" : "admin@1234",
  "httpsEnable" : true
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群安全模式。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ChangeModeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeModeRequest request = new ChangeModeRequest();
        request.withClusterId("{cluster_id}");
        ChangeModeReq body = new ChangeModeReq();
        body.withHttpsEnable(true);
        body.withAdminPwd("admin@1234");
        body.withAuthorityEnable(true);
        request.withBody(body);
        try {
            ChangeModeResponse response = client.changeMode(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群安全模式。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ChangeModeRequest()
        request.cluster_id = "{cluster_id}"
        request.body = ChangeModeReq(
            https_enable=True,
            admin_pwd="admin@1234",
            authority_enable=True
        )
        response = client.change_mode(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群安全模式。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeModeRequest{}
    request.ClusterId = "{cluster_id}"
    adminPwdChangeModeReq:= "admin@1234"
    request.Body = &model.ChangeModeReq{
        HttpsEnable: true,
        AdminPwd: &adminPwdChangeModeReq,
        AuthorityEnable: true,
    }
    response, err := client.ChangeMode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
404	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。

错误码

请参见错误码。

4.1.24 添加独立 master、client

功能介绍

由于集群数据面业务的增长或者不确定性，很难在一开始就能够把集群的规模形态想明白，该接口能够在非独立master和client的集群上面独立master、client角色。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/type/{type}/independent

表 4-84 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定需要独立master或client的集群ID。
type	是	String	指定待新增独立节点类型。 ● ess-master：Master节点。 ● ess-client：Client节点。

请求参数

表 4-85 请求 Body 参数

参数	是否必选	参数类型	描述
type	是	IndependentTypeReq object	独立master或client请求体参数。

参数	是否必选	参数类型	描述
is_auto_pay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 - 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 - 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 **4-86** IndependentTypeReq

参数	是否必选	参数类型	描述
flavor_ref	是	String	规格id，该参数通过获取实例规格列表接口获取，根据集群版本选择所需要的规格id
node_size	是	Integer	要独立节点个数。 ● 如果路径参数type取值为“ess-master”即新增独立master节点，节点个数必须为大于等于三且小于等于10的奇数。 ● 如果路径参数type取值为“ess-client”即新增独立client节点，节点个数要求大于等于1小于等于32。
volume_type	是	String	节点存储类型：取值为ULTRAHIGH，COMMON，HIGH。

响应参数

状态码： 200

表 **4-87** 响应 Body 参数

参数	参数类型	描述
id	String	集群ID。

请求示例

添加独立master、client。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/type/ess-client/independent

{
  "type" : {
    "flavor_ref" : "d9dc06ae-b9c4-4ef4-acd8-953ef4205e27",
    "node_size" : 3,
    "volume_type" : "COMMON"
  }
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "id" : "320afa24-ff2a-4f44-8460-6ba95e512ad4"
}
```

SDK 代码示例

SDK代码示例如下。

Java

添加独立master、client。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class AddIndependentNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        AddIndependentNodeRequest request = new AddIndependentNodeRequest();
        request.withClusterId("{cluster_id}");
        request.withType("{type}");
        IndependentReq body = new IndependentReq();
        IndependentBodyReq typebody = new IndependentBodyReq();
        typebody.withFlavorRef("d9dc06ae-b9c4-4ef4-acd8-953ef4205e27")
            .withNodeSize(3)
            .withVolumeType("COMMON");
        body.withType(typebody);
        request.withBody(body);
        try {
            AddIndependentNodeResponse response = client.addIndependentNode(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

添加独立master、client。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddIndependentNodeRequest()
        request.cluster_id = "{cluster_id}"
        request.type = "{type}"
        typebody = IndependentBodyReq(
            flavor_ref="d9dc06ae-b9c4-4ef4-acd8-953ef4205e27",
            node_size=3,
            volume_type="COMMON"
        )
        request.body = IndependentReq(
            type=typebody
        )
        response = client.add_independent_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

添加独立master、client。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddIndependentNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.Type = "{type}"
    typebody := &model.IndependentBodyReq{
        FlavorRef: "d9dc06ae-b9c4-4ef4-acd8-953ef4205e27",
        NodeSize: int32(3),
        VolumeType: "COMMON",
    }
    request.Body = &model.IndependentReq{
        Type: typebody,
    }
    response, err := client.AddIndependentNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

```
  }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	表明服务端能被请求访问到，但是不能理解用户的请求。

错误码

请参见**错误码**。

4.1.25 集群内核升级

功能介绍

该接口用于将低版本的ES升级到高版本或同版本ES。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/inst-type/{inst_type}/image/upgrade

表 4-88 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待升级的集群ID。
inst_type	是	String	指定待升级的节点类型，当前仅支持all。

请求参数

表 **4-89** 请求 Body 参数

参数	是否必选	参数类型	描述
target_image_id	是	String	目标镜像版本ID。
upgrade_type	是	String	升级类型。 ● same：同版本。 ● cross：跨版本。 ● crossEngine：跨引擎。
indices_backup_check	是	Boolean	是否进行备份校验。 ● true：进行校验。 ● false：不进行校验。
agency	是	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。
cluster_load_check	否	Boolean	是否校验负载。默认为true。 ● true：进行校验。 ● false：不进行校验。

响应参数

无

请求示例

更新集群版本。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/inst-type/all/image/upgrade

{
  "target_image_id" : "{target_image_id}",
  "upgrade_type" : "same",
  "indices_backup_check" : true,
  "agency" : "css-test-agency",
  "cluster_load_check" : true
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

更新集群版本。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpgradeCoreSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpgradeCoreRequest request = new UpgradeCoreRequest();
        request.withClusterId("{cluster_id}");
        request.withInstType("{inst_type}");
        UpgradingTheKernelBody body = new UpgradingTheKernelBody();
        body.withAgency("css-test-agency");
        body.withIndicesBackupCheck(true);
        body.withUpgradeType(UpgradingTheKernelBody.UpgradeTypeEnum.fromValue("same"));
        body.withTargetImageId("{target_image_id}");
        request.withBody(body);
        try {
            UpgradeCoreResponse response = client.upgradeCore(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新集群版本。

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeCoreRequest()
        request.cluster_id = "{cluster_id}"
        request.inst_type = "{inst_type}"
        request.body = UpgradingTheKernelBody(
            agency="css-test-agency",
            indices_backup_check=True,
            upgrade_type="same",
            target_image_id="{target_image_id}"
        )
        response = client.upgrade_core(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新集群版本。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpgradeCoreRequest{}
    request.ClusterId = "{cluster_id}"
    request.InstType = "{inst_type}"
    request.Body = &model.UpgradingTheKernelBody{
        Agency: "css-test-agency",
        IndicesBackupCheck: true,
        UpgradeType: model.GetUpgradingTheKernelBodyUpgradeTypeEnum().SAME,
        TargetImageId: "{target_image_id}",
    }
    response, err := client.UpgradeCore(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.26 获取目标镜像 ID

功能介绍

该接口用于获取当前集群的可升级目标镜像ID。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/target/{upgrade_type}/images

表 4-90 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	待升级的集群的ID。
upgrade_type	是	String	升级目标版本类型： ● same：相同版本。 ● cross：跨版本。 ● cross-engine：跨引擎。

请求参数

无

响应参数

状态码： 200

表 4-91 响应 Body 参数

参数	参数类型	描述
needUploadUpgradePlugin	Boolean	是否需要上传升级后版本的插件。
imageInfoList	Array of **GetTargetImageIdDetail** objects	镜像详情信息。

表 4-92 GetTargetImageIdDetail

参数	参数类型	描述
id	String	可以升级的目标镜像ID。
displayName	String	可以升级的目标镜像名称。

参数	参数类型	描述
imageDesc	String	镜像描述信息。
datastoreType	String	镜像引擎类型。
datastoreVersion	String	镜像引擎版本。
priority	Integer	优先级。

请求示例

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/target/cross/images

{
  "needUploadUpgradePlugin" : false,
  "imageInfoList" : [ ]
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "needUploadUpgradePlugin" : false,
  "imageInfoList" : [ ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListImagesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListImagesRequest request = new ListImagesRequest();
        request.withClusterId("{cluster_id}");
        request.withUpgradeType("{upgrade_type}");
        try {
            ListImagesResponse response = client.listImages(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListImagesRequest()
        request.cluster_id = "{cluster_id}"
        request.upgrade_type = "{upgrade_type}"
        response = client.list_images(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListImagesRequest{}
    request.ClusterId = "{cluster_id}"
    request.UpgradeType = "{upgrade_type}"
    response, err := client.ListImages(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见错误码。

4.1.27 获取升级详情信息

功能介绍

由于升级过程时间较长，该接口可以展示当前升级（切换AZ）节点的各个阶段信息。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/upgrade/detail

表 4-93 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	待升级的集群的ID。

表 4-94 Query 参数

参数	是否必选	参数类型	描述
start	否	Integer	偏移量。 偏移量为一个大于0小于终端节点服务总个数的整数，表示从偏移量后面的终端节点服务开始查询。
limit	否	Integer	查询返回终端节点服务的连接列表限制每页个数，即每页返回的个数。
action_mode	否	String	查询升级行为。 ● 查询集群版本升级详情：不填写该参数。 ● 查询切换AZ详情：当前仅支持AZ_MIGRATION。

请求参数

无

响应参数

状态码： 200

表 4-95 响应 Body 参数

参数	参数类型	描述
totalSize	Integer	下发执行接口次数。
detailList	Array of **GetUpgradeDetailInfo** objects	升级任务详情。

表 4-96 GetUpgradeDetailInfo

参数	参数类型	描述
id	String	任务ID。
startTime	String	升级开始时间。
endTime	String	升级结束时间。
status	String	任务状态。 ● RUNNING：升级中。 ● SUCCESS：升级成功。 ● FAILED：升级失败。 ● PARTIAL_FAILED：部分升级失败。
agencyName	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。
imageInfo	**GetTargetImageIdDetail** object	镜像详情信息。
totalNodes	String	所有需要升级的节点名称集合。
completedNodes	String	所有升级完成的节点名称集合。
currentNodeName	String	当前正在升级的节点名称。
executeTimes	String	重试次数。

参数	参数类型	描述
migrateParam	String	集群当前升级的行为。当有query参数时显示该返回值。
finalAzInfoMap	String	集群升级预期结果。当有query参数时显示该返回值。
currentNodeDetail	Array of **CurrentNodeDetail** objects	当前正在升级节点任务详情。

表 **4-97** GetTargetImageIdDetail

参数	参数类型	描述
id	String	可以升级的目标镜像ID。
displayName	String	可以升级的目标镜像名称。
imageDesc	String	镜像描述信息。
datastoreType	String	镜像引擎类型。
datastoreVersion	String	镜像引擎版本。
priority	Integer	优先级。

表 **4-98** CurrentNodeDetail

参数	参数类型	描述
order	Integer	升级任务序号。
name	String	升级任务名称。
status	String	当前任务状态。
desc	String	当前任务描述。
beginTime	String	当前任务开始时间。
endTime	String	当前任务结束时间。

请求示例

- 集群升级请求示例

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
upgrade/detail
```

- 集群升级请求示例

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
upgrade/detail?action_mode=AZ_MIGRATION
```

响应示例

状态码： 200

请求已成功。

- 集群镜像升级响应示例。

```
{
  "totalSize" : 1,
  "detailList" : [ {
    "id" : "b7ac4c5b-3bda-4feb-a303-eb80f4bce986",
    "startTime" : "2023-01-05T02:23:39",
    "endTime" : "",
    "status" : "RUNNING",
    "agencyName" : "css_test_agency",
    "imageInfo" : {
      "id" : "439b5d30-5968-45df-b088-d030a858522d",
      "displayName" : "7.10.2_22.5.1_1230",
      "imageDesc" : "The latest image of version 7.10.2 will be officially released on December 30, 
2022. the stability of the cluster has been optimized. It is recommended that you upgrade to this 
version.",
      "datastoreType" : "elasticsearch",
      "datastoreVersion" : "7.10.2",
      "priority" : 16
    },
    "totalNodes" : "css-test-ess-esn-1-1,css-test-ess-esn-2-1,css-test-ess-esn-3-1",
    "completedNodes" : "",
    "currentNodeName" : "css-test-ess-esn-1-1",
    "executeTimes" : "1",
    "currentNodeDetail" : [ {
      "order" : 0,
      "name" : "Data migration",
      "status" : "SUCCESS",
      "desc" : "Data is migrated from a node to other nodes in the cluster by running the exclude 
command.",
      "beginTime" : "2023-01-05T02:23:42",
      "endTime" : "2023-01-05T02:29:51"
    }, {
      "order" : 12,
      "name" : "Task status update",
      "status" : "WAITING",
      "desc" : "The cluster task status is refreshed. If there are no nodes to be replaced, the task is 
marked as completed. Otherwise, another node starts to be replaced.",
      "beginTime" : "",
      "endTime" : ""
    } ]
  } ]
}
```

- 切换AZ响应示例。

```
{
  "totalSize" : 1,
  "detailList" : [ {
    "id" : "8ebe958b-b8c6-4939-b5a7-85aee9468888",
    "startTime" : "2022-12-29T08:32:29",
    "endTime" : "",
    "status" : "RUNNING",
    "agencyName" : "css_test_agency",
    "totalNodes" : "css-test-ess-esn-2-1,css-test-ess-esn-1-1,css-test-ess-esn-3-1",
    "completedNodes" : "css-test-ess-esn-2-1",
    "currentNodeName" : "css-test-ess-esn-1-1",
    "executeTimes" : "1",
    "migrateParam" : "{\"instType\":\"ess\",\"migrateType\":\"az_migrate\",\"sourceAz\":\"cn-north-4a
\",\"targetAz\":\"cn-north-4c\"}",
    "finalAzInfoMap" : "{\"cn-north-4c\":\"css-test-ess-esn-2-1,css-test-ess-esn-1-1,css-test-ess-
esn-3-1\"}",
    "currentNodeDetail" : [ {
      "order" : 0,
      "name" : "Data migration",
      "status" : "SUCCESS",
      "desc" : "Data is migrated from a node to other nodes in the cluster by running the exclude 
command.",
      "beginTime" : "2022-12-29T08:41:05",
      "endTime" : "2022-12-29T08:41:23"
    }, {
      "order" : 12,
      "name" : "Task status update",
      "status" : "RUNNING",
      "desc" : "The cluster task status is refreshed. If there are no nodes to be replaced, the task is 
marked as completed. Otherwise, another node starts to be replaced.",
      "beginTime" : "2023-01-04T06:53:42",
      "endTime" : ""
    } ]
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpgradeDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpgradeDetailRequest request = new UpgradeDetailRequest();
        request.withClusterId("{cluster_id}");
        try {
            UpgradeDetailResponse response = client.upgradeDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeDetailRequest()
        request.cluster_id = "{cluster_id}"
        response = client.upgrade_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpgradeDetailRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.UpgradeDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.28 重试升级失败任务

功能介绍

由于升级过程时间较长，可能由于网络等原因导致升级失败，可以通过该接口重试该任务或终止该任务的影响。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/upgrade/{action_id}/retry

表 **4-99** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	待重试的集群ID。
action_id	是	String	待重试的任务ID。

表 **4-100** Query 参数

参数	是否必选	参数类型	描述
retry_mode	否	String	当该参数不为空时，终止该任务的影响。当前仅支持abort。

请求参数

无

响应参数

无

请求示例

- 升级任务重试请求示例。

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
upgrade/bcdb711c-a7f0-4007-b8ee-9f13c05f8326/retry
```

- 升级任务终止请求示例。

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
upgrade/bcdb711c-a7f0-4007-b8ee-9f13c05f8326/retry?retry_mode=abort
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class RetryUpgradeTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        RetryUpgradeTaskRequest request = new RetryUpgradeTaskRequest();
        request.withClusterId("{cluster_id}");
        request.withActionId("{action_id}");
        try {
            RetryUpgradeTaskResponse response = client.retryUpgradeTask(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RetryUpgradeTaskRequest()
        request.cluster_id = "{cluster_id}"
        request.action_id = "{action_id}"
        response = client.retry_upgrade_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RetryUpgradeTaskRequest{}
    request.ClusterId = "{cluster_id}"
    request.ActionId = "{action_id}"
    response, err := client.RetryUpgradeTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.29 切换安全组

功能介绍

该接口可以在集群创建成功后，修改安全组。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/sg/change

表 4-101 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	待修改安全组的集群ID。

请求参数

表 **4-102** 请求 Body 参数

参数	是否必选	参数类型	描述
security_group_ids	是	String	期望安全组的ID。

响应参数

无

请求示例

修改当前集群所属安全组。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/sg/change

{
  "security_group_ids" : "b1038649-1f77-4ae9-b64d-9af56e422652"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改当前集群所属安全组。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ChangeSecurityGroupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeSecurityGroupRequest request = new ChangeSecurityGroupRequest();
        request.withClusterId("{cluster_id}");
        ChangeSecurityGroupReq body = new ChangeSecurityGroupReq();
        body.withSecurityGroupIds("b1038649-1f77-4ae9-b64d-9af56e422652");
        request.withBody(body);
        try {
            ChangeSecurityGroupResponse response = client.changeSecurityGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改当前集群所属安全组。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ChangeSecurityGroupRequest()
        request.cluster_id = "{cluster_id}"
        request.body = ChangeSecurityGroupReq(
            security_group_ids="b1038649-1f77-4ae9-b64d-9af56e422652"
        )
        response = client.change_security_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改当前集群所属安全组。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeSecurityGroupRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.ChangeSecurityGroupReq{
        SecurityGroupIds: "b1038649-1f77-4ae9-b64d-9af56e422652",
    }
    response, err := client.ChangeSecurityGroup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。

状态码	描述
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见错误码。

4.1.30 切换集群实例 AZ

功能介绍

该接口通过指定节点类型切换AZ。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/inst-type/{inst_type}/azmigrate

表 4-103 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	待切换AZ的集群ID。
inst_type	是	String	待切换AZ的节点类型。支持: ● all：所有节点类型。 ● ess： 数据节点。 ● ess-cold: 冷数据节点。 ● ess-client: Client节点。 ● ess-master: Master节点。

请求参数

表 4-104 请求 Body 参数

参数	是否必选	参数类型	描述
source_az	是	String	节点当前所在AZ。
target_az	是	String	期望节点最终分布AZ。
migrate_type	是	String	AZ迁移方式： ● multi_az_change：高可用改造。 ● az_migrate： AZ平移
agency	是	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。
indices_backup_check	否	Boolean	是否进行全量索引快照备份检测。 true：进行全量索引快照备份检测。 false：不进行全量索引快照备份检测。

响应参数

无

请求示例

切换当前集群AZ。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/inst-type/all/azmigrate

{
  "source_az" : "cn-north-4c",
  "target_az" : "cn-north-4a",
  "migrate_type" : "az_migrate",
  "agency" : "css-test-agency"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

切换当前集群AZ。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateAzByInstanceTypeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateAzByInstanceTypeRequest request = new UpdateAzByInstanceTypeRequest();
        request.withClusterId("{cluster_id}");
        request.withInstType(UpdateAzByInstanceTypeRequest.InstTypeEnum.fromValue("{inst_type}"));
        UpdateAzByInstanceTypeReq body = new UpdateAzByInstanceTypeReq();
        body.withAgency("css-test-agency");
        body.withMigrateType("az_migrate");
        body.withTargetAz("cn-north-4a");
        body.withSourceAz("cn-north-4c");
        request.withBody(body);
        try {
            UpdateAzByInstanceTypeResponse response = client.updateAzByInstanceType(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

切换当前集群AZ。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateAzByInstanceTypeRequest()
        request.cluster_id = "{cluster_id}"
        request.inst_type = "{inst_type}"
        request.body = UpdateAzByInstanceTypeReq(
            agency="css-test-agency",
            migrate_type="az_migrate",
            target_az="cn-north-4a",
            source_az="cn-north-4c"
        )
        response = client.update_az_by_instance_type(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

切换当前集群AZ。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
```

```
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAzByInstanceTypeRequest{}
    request.ClusterId = "{cluster_id}"
    request.InstType = model.GetUpdateAzByInstanceTypeRequestInstTypeEnum().INST_TYPE
    request.Body = &model.UpdateAzByInstanceTypeReq{
        Agency: "css-test-agency",
        MigrateType: "az_migrate",
        TargetAz: "cn-north-4a",
        SourceAz: "cn-north-4c",
    }
    response, err := client.UpdateAzByInstanceType(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.1.31 创建集群 V2

功能介绍

该接口用于创建拥有多种不同节点类型（ess，ess-cold，ess-client，ess-master）组合的集群。

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/clusters

表 4-105 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

请求参数

表 4-106 请求 Body 参数

参数	是否必选	参数类型	描述
cluster	是	**CreateClusterBody** object	集群对象。

表 4-107 CreateClusterBody

参数	是否必选	参数类型	描述
name	是	String	集群名称。4~32个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。
backupStrategy	否	**CreateClusterBackupStrategyBody** object	开启自动创建快照策略。 当backupStrategy参数配置不为空时，才会开启自动创建快照策略。
roles	是	Array of **CreateClusterRolesBody** objects	集群信息。
nics	是	**CreateClusterInstanceNicsBody** object	子网信息。
enterprise_project_id	否	String	企业项目ID。创建集群时，给集群绑定企业项目ID。最大长度36个字符，带"-"连字符的UUID格式，或者是字符串"0"。"0"表示默认企业项目。

参数	是否必选	参数类型	描述
tags	否	Array of **CreateClusterTagsBody** objects	集群标签。
availability_zone	是	String	可用区。需要指定可用区的名称（可用分区名称）。 默认指定单AZ。指定多AZ时，各个可用分区名称需要使用英文逗号（,）分隔，以“华北-北京四”为例，选择三AZ时，availability_zone取值为cn-north-4a,cn-north-4b,cn-north-4c。如果使用单AZ，availability_zone默认取值为空。 **说明** 说明 - 选择多AZ时，各个可用分区名称不能重复输入，并且要求节点个数大于等于AZ个数。 - 如果节点个数为AZ个数的倍数，节点将会均匀的分布到各个AZ。 - 如果节点个数不为AZ个数的倍数时，各个AZ分布的节点数量之差的绝对值小于等于1。 - 可用分区名称，请在地区和终端节点获取。
datastore	是	**CreateClusterDatastoreBody** object	引擎类型，支持elasticsearch、opensearch和logstash。
authorityEnable	否	Boolean	是否开启认证，取值范围为true或false。默认关闭认证功能。 ● true：表示集群开启认证。 ● false：表示集群不开启认证。 **说明** 此参数只有6.5.4及之后版本支持。
httpsEnable	否	Boolean	设置是否进行通信加密。取值范围为true或false。默认关闭通信加密功能。当httpsEnable设置为true时，authorityEnable字段需要设置为true。- true：表示集群进行通信加密。- false：表示集群不进行通信加密。>此参数只有6.5.4及之后版本支持。

参数	是否必选	参数类型	描述
adminPwd	否	String	安全模式下集群管理员admin的密码，只有在创建集群时authorityEnable设置为true时需要设置此参数。 ● 参数范围：8~32个字符。 ● 参数要求：密码至少包含大写字母，小写字母，数字、特殊字符四类中的三类，其中可输入的特殊字符为：~!@#$%&*()-_=|[{}];:,<.>/?
publicIPReq	否	**CreateClusterPublicIpReq** object	公网访问信息。只有在httpsEnable设置为true时该参数配置生效。
loadBalance	否	**CreateClusterLoadBalance** object	终端节点服务信息。
publicKibanaReq	否	**CreateClusterPublicKibanaReq** object	Kibana公网访问信息。只有在authorityEnable设置为true时该参数配置生效。
payInfo	否	**PayInfoBody** object	包周期集群相关参数订购包周期集群时使用。
ipv6_enable	否	Boolean	集群是否开启自动分配IPv6地址，默认值为false。

表 4-108 CreateClusterBackupStrategyBody

参数	是否必选	参数类型	描述
period	是	String	每天创建快照的时刻，只支持整点，后面需加上时区，格式为“HH:mm z”，“HH:mm”表示整点时间，“z”表示时区。比如“00:00 GMT+08:00”、“01:00 GMT+08:00”等。 **说明** 默认值“00:00 GMT+08:00”，当frequency取值HOUR时，无需指定period值，默认从设置成功的下一整点创建。

参数	是否必选	参数类型	描述
frequency	否	String	自动创建快照的执行频次。 **说明** 默认值DAY，取值HOUR时，表示每小时整点执行一次；取值DAY时，表示每天执行一次；取值SUN、MON、TUE、WED、THU、FRI、SAT时，表示在每周对应时间执行一次。比如SUN表示每周日执行一次。保留快照个数与自动创建快照策略中设置的执行频次和索引有关。当执行频次间隔时间短或索引数据量大，保留自动快照可能会达不到设置的个数，请谨慎选择！
prefix	是	String	自动创建的快照的前缀，需要用户自己手动输入。只能包含1~32位小写字母、数字、中划线或者下划线，并且以小写字母开头。
keepday	是	Integer	自定义设置快照保留的个数，范围是1～90。系统在半点时刻会自动删除超过保留个数的快照。过期删除策略只针对与当前自动创建快照策略相同执行频次的自动快照。
bucket	否	String	备份使用的OBS桶名称。
basePath	否	String	快照在OBS桶中的存放路径。
agency	否	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。 **说明** 如果bucket、basePath和agency三个参数同时为空，则系统会自动创建OBS桶和IAM代理（若创建失败，则需要手工配置正确的参数）。

表 4-109 CreateClusterRolesBody

参数	是否必选	参数类型	描述
flavorRef	是	String	实例规格名称。可以使用**获取实例规格列表**的name属性确认当前拥有的规格信息。

参数	是否必选	参数类型	描述
volume	是	CreateClusterInstanceVolumeBody object	volume信息。当flavorRef选择的是本地盘规格时不需要填写，本地盘规格可以通过获取实例规格列表中diskrange属性获得。目前支持的本地盘规格有： ● ess.spec-i3small ● ess.spec-i3medium ● ess.spec-i3.8xlarge.8 ● ess.spec-ds.xlarge.8 ● ess.spec-ds.2xlarge.8 ● ess.spec-ds.4xlarge.8
type	是	String	● ess-master对应Master节点。 ● ess-client对应Clinet节点。 ● ess-cold对应冷数据节点。 ● ess对应数据节点。 ● lgs对应Logstash节点。
instanceNum	是	Integer	实例个数。 ● ess节点，选择范围：1~32个节点数量。 – 若同时选择ess和ess-master时，可以选择1~200个节点数量。 – 若同时选择ess和ess-client时，可以选择1~32个节点数量。 – 若同时选择ess和ess-cold时，可以选择1~32个节点数量。 ● ess-master节点，选择范围：3~10内的奇数个节点数量。 ● ess-client节点，选择范围：1~32个节点数量。 ● ess-cold节点，选择范围：1~32个节点数量。

表 **4-110** CreateClusterInstanceVolumeBody

参数	是否必选	参数类型	描述
volume_type	是	String	卷类型。 ● COMMON：普通I/O。 ● HIGH：高I/O。 ● ULTRAHIGH：超高I/O。 ● ESSD：极速SSD。
size	是	Integer	卷大小，必须为大于0且为4和10的公倍数，磁盘规格大小可以通过获取实例规格列表中diskrange属性获得。 单位：GB。 **说明** ess-master节点和ess-client节点默认大小为40G，且不可更改。

表 **4-111** CreateClusterInstanceNicsBody

参数	是否必选	参数类型	描述
vpcId	是	String	指定虚拟私有云ID，用于集群网络配置。
netId	是	String	子网ID(网络ID)。
securityGroupId	是	String	安全组ID。

表 **4-112** CreateClusterTagsBody

参数	是否必选	参数类型	描述
key	是	String	集群标签的key值。可输入的字符串长度为1~36个字符。只能包含数字、字母、中划线"-"和下划线"_"。
value	是	String	集群标签的value值。可输入的字符串长度为0~43个字符。只能包含数字、字母、中划线"-"和下划线"_"。

表 **4-113** CreateClusterDatastoreBody

参数	是否必选	参数类型	描述
version	是	String	CSS集群引擎版本号。详细请参考CSS使用前必读中支持的版本。
type	是	String	引擎类型，支持elasticsearch、opensearch和logstash。

表 **4-114** CreateClusterPublicIpReq

参数	是否必选	参数类型	描述
eip	是	CreateClusterPublicEip object	公网带宽信息。
elbWhiteListReq	是	CreateClusterElbWhiteList object	弹性IP白名单。
publicBindType	是	String	是否自动绑定弹性公网IP。当前仅支持auto_assign为自动分配参数。
eipId	否	String	弹性公网IP的ID。

表 **4-115** CreateClusterPublicEip

参数	是否必选	参数类型	描述
bandWidth	是	CreateClusterPublicEipSize object	设置公网带宽大小。

表 **4-116** CreateClusterPublicEipSize

参数	是否必选	参数类型	描述
size	是	Integer	带宽大小。

表 **4-117** CreateClusterElbWhiteList

参数	是否必选	参数类型	描述
enableWhiteList	是	Boolean	是否开启公网访问控制。
whiteList	否	String	公网访问控制白名单。需要添加白名单的网段或ip，以逗号隔开，不可重复。

表 **4-118** CreateClusterLoadBalance

参数	是否必选	参数类型	描述
endpointWithDnsName	是	Boolean	是否开启内网域名。 ● true： 开启内网域名。 ● false： 关闭内网域名。
vpcPermissions	否	Array of strings	访问控制。
professionVpcep	否	Boolean	创建专业型终端节点。 ● true：开启。 ● false：不开启。
dualstackEnable	否	Boolean	是否开启IPv4/IPv6双栈网络，仅支持在创建专业型终端节点时开启双栈网络，且集群的VPC子网支持IPv6。 ● true：开启。 ● false：不开启。

表 **4-119** CreateClusterPublicKibanaReq

参数	是否必选	参数类型	描述
eipSize	是	Integer	带宽大小。
elbWhiteList	是	**CreateClusterPublicKibanaElbWhiteList** object	kibana白名单信息。

表 **4-120** CreateClusterPublicKibanaElbWhiteList

参数	是否必选	参数类型	描述
whiteList	是	String	白名单。需要添加白名单的网段或ip，以逗号隔开，不可重复。
enableWhiteList	是	Boolean	是否开启kibana访问控制。

表 **4-121** PayInfoBody

参数	是否必选	参数类型	描述
payModel	是	Integer	订购周期类型。 ● 2：包月。 ● 3：包年。
period	是	Integer	订购周期数。 ● 若payModel为2，则有效值为1-9。 ● 若payModel为3，则有效值为1-3。
isAutoRenew	否	Integer	是否自动续订，为空时表示不自动续订。 ● 1：自动续订。 ● 0：不自动续订（默认）。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

响应参数

状态码： 200

表 **4-122** 响应 Body 参数

参数	参数类型	描述
cluster	**CreateClusterResp** object	集群对象。若创建的是按需集群，则只返回cluster参数。
orderId	String	订单号。只有创建包周期集群时才会返回orderId参数。

表 **4-123** CreateClusterResp

参数	参数类型	描述
id	String	集群ID。
name	String	集群名称。

请求示例

- 创建包周期集群请求示例。

```
POST /v2.0/6204a5bd270343b5885144cf9c8c158d/clusters

{
  "cluster" : {
    "name" : "css-cluster",
    "backupStrategy" : {
      "period" : "16:00 GMT+08:00",
      "prefix" : "snapshot",
      "keepday" : 7,
      "frequency" : "DAY",
      "bucket" : "OBS-obsName",
      "basePath" : "css_repository/obs-path",
      "agency" : "css_obs_agency"
    },
    "roles" : [ {
      "flavorRef" : "ess.spec-4u8g",
      "volume" : {
        "volume_type" : "COMMON",
        "size" : 100
      },
      "type" : "ess",
      "instanceNum" : 1
    } ],
    "nics" : {
      "vpcId" : "{VPC ID}",
      "netId" : "{NET ID}",
      "securityGroupId" : "{Security group ID}"
    },
    "enterprise_project_id" : 0,
    "tags" : [ {
      "key" : "k1",
      "value" : "v1"
    }, {
      "key" : "k2",
      "value" : "v2"
    } ],
    "availability_zone" : "{Az code}",
    "datastore" : {
      "version" : "{cluster-version}",
      "type" : "elasticsearch"
    },
    "authorityEnable" : true,
    "httpsEnable" : true,
    "adminPwd" : "{password}",
    "publicIPReq" : {
      "eip" : {
        "bandWidth" : {
          "size" : 5
        }
      },
      "elbWhiteListReq" : {
        "enableWhiteList" : true,
        "whiteList" : "127.0.0.1"
      },
      "publicBindType" : "auto_assign",
      "eipId" : null
    },
    "loadBalance" : {
      "endpointWithDnsName" : false,
      "vpcPermissions" : [ "{Account ID}" ]
    },
    "publicKibanaReq" : {
      "eipSize" : 5,
      "elbWhiteList" : {
        "whiteList" : "127.0.0.1",
        "enableWhiteList" : true
      }
    },
    "payInfo" : {
      "payModel" : 2,
      "period" : 1,
      "isAutoRenew" : 1,
      "isAutoPay" : null
    }
  }
}
```

- 创建按需集群请求示例。

```
{
  "cluster" : {
    "name" : "css-cluster",
    "backupStrategy" : {
      "period" : "16:00 GMT+08:00",
      "prefix" : "snapshot",
      "keepday" : 7,
      "frequency" : "DAY",
      "bucket" : "OBS-obsName",
      "basePath" : "css_repository/obs-path",
      "agency" : "css_obs_agency"
    },
    "roles" : [ {
      "flavorRef" : "ess.spec-4u8g",
      "volume" : {
        "volume_type" : "COMMON",
        "size" : 100
      },
      "type" : "ess",
      "instanceNum" : 1
    } ],
    "nics" : {
      "vpcId" : "{VPC ID}",
      "netId" : "{NET ID}",
      "securityGroupId" : "{Security group ID}"
    },
    "enterprise_project_id" : 0,
    "tags" : [ {
      "key" : "k1",
      "value" : "v1"
    }, {
      "key" : "k2",
      "value" : "v2"
    } ],
    "availability_zone" : "{Az code}",
    "datastore" : {
      "version" : "{cluster-version}",
      "type" : "elasticsearch"
    },
    "authorityEnable" : true,
    "httpsEnable" : true,
    "adminPwd" : "{password}",
    "publicIPReq" : {
      "eip" : {
        "bandWidth" : {
          "size" : 5
        }
      },
      "elbWhiteListReq" : {
        "enableWhiteList" : true,
        "whiteList" : "127.0.0.1"
      },
      "publicBindType" : "auto_assign",
      "eipId" : null
    },
    "loadBalance" : {
      "endpointWithDnsName" : false,
      "vpcPermissions" : [ "{Account ID}" ]
    },
    "publicKibanaReq" : {
      "eipSize" : 5,
      "elbWhiteList" : {
        "whiteList" : "127.0.0.1",
        "enableWhiteList" : true
      }
    }
  }
}
```

响应示例

状态码：200

请求已成功

- 创建按需期集群响应示例。

```
{
  "cluster" : {
    "id" : "ef683016-871e-48bc-bf93-74a29d60d214",
    "name" : "ES-Test"
  }
}
```

- 创建包周期集群响应示例。

```
{
  "orderId" : "CS2208241141LY94J",
  "cluster" : {
    "id" : "ef683016-871e-48bc-bf93-74a29d60d214",
    "name" : "ES-Test"
  }
}
```

状态码

状态码	描述
200	请求已成功
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.32 重启集群 V2

功能介绍

该接口可以用于重启当前集群拥有的全部节点类型，或部分节点类型组合的节点。

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/clusters/{cluster_id}/restart

表 4-124 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定重启集群ID。

请求参数

表 4-125 请求 Body 参数

参数	是否必选	参数类型	描述
type	是	String	操作角色。参数范围： ● node ● role
value	是	String	操作参数。参数说明： ● 当操作角色为node时，value为节点ID,通过查询集群详情获取instances中的id属性。 ● 当操作角色为role时，value为节点类型(ess、ess-master、ess-client、ess-cold)的多种不同组合。

响应参数

无

请求示例

重启当前集群拥有的全部节点类型，或部分节点类型组合的节点。

```
POST /v2.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/restart

{
  "type" : "role",
  "value" : "ess"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.1.33 滚动重启

功能介绍

该接口会一个一个重启节点，在索引数量比较多的情况下耗时较长

说明

仅当集群的节点数量（含Master节点、Client节点和冷数据节点）大于3时，才支持滚动重启

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/clusters/{cluster_id}/rolling_restart

表 4-126 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定重启集群ID。

请求参数

表 4-127 请求 Body 参数

参数	是否必选	参数类型	描述
type	是	String	操作角色。type的值只能为role。

参数	是否必选	参数类型	描述
value	是	String	实例类型（选择实例类型时至少需要一个数据节点），多个类型使用逗号隔开。例如: ● ess-master对应Master节点。 ● ess-client对应Client节点。 ● ess-cold对应冷数据节点。 ● ess对应数据节点。 ● all对应所有节点。

响应参数

无

请求示例

重启节点。

```
POST /v2.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
rolling_restart

{
  "type" : "role",
  "value" : "ess"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.2 词库管理接口

4.2.1 加载自定义词库

功能介绍

该接口用于加载存放于OBS的自定义词库。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/thesaurus

表 4-128 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定配置自定义词库的集群ID。

请求参数

表 4-129 请求 Body 参数

参数	是否必选	参数类型	描述
bucketName	是	String	词库文件存放的OBS桶（桶类型必须为标准存储或者低频存储，不支持归档存储）。
mainObject	否	String	主词词库文件对象，必须为UTF-8无BOM编码的文本文件，一行一个分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。

参数	是否必选	参数类型	描述
stopObject	否	String	停词词库文件对象，必须为UTF-8无BOM编码的文本文件，一行一个分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。
synonymObject	否	String	同义词词库文件，必须为UTF-8无BOM编码的文本文件，一行一组分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。
static_main_object	否	String	静态主词词库文件，必须为UTF-8无BOM编码的文本文件，一行一组分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。仅支持此词库功能上线后的新集群。
static_stop_object	否	String	静态停词词库文件，必须为UTF-8无BOM编码的文本文件，一行一组分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。仅支持此词库功能上线后的新集群。
extra_main_object	否	String	Extra主词词库文件，必须为UTF-8无BOM编码的文本文件，一行一组分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。仅支持此词库功能上线后的新集群。

参数	是否必选	参数类型	描述
extra_stop_object	否	String	Extra停词词库文件，必须为UTF-8无BOM编码的文本文件，一行一组分词，文件大小最大支持100M。 7个词库参数至少修改一个词库。注：参数传递""空字符串为清空此词库，不传或传递null为不修改。仅支持此词库功能上线后的新集群。

响应参数

无

请求示例

开启并配置词库信息。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
thesaurus

{
  "bucketName" : "test-bucket",
  "mainObject" : "word/main.txt",
  "stopObject" : "word/stop.txt",
  "synonymObject" : "word/synonym.txt",
  "static_main_object" : "word/staticMain.txt",
  "static_stop_object" : "word/staticStop.txt",
  "extra_main_object" : "word/extraMain.txt",
  "extra_stop_object" : "word/extraStop.txt"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启并配置词库信息。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;
```

```
public class CreateLoadIkThesaurusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateLoadIkThesaurusRequest request = new CreateLoadIkThesaurusRequest();
        request.withClusterId("{cluster_id}");
        LoadCustomThesaurusReq body = new LoadCustomThesaurusReq();
        body.withSynonymObject("word/synonym.txt");
        body.withStopObject("word/stop.txt");
        body.withMainObject("word/main.txt");
        body.withBucketName("test-bucket");
        request.withBody(body);
        try {
            CreateLoadIkThesaurusResponse response = client.createLoadIkThesaurus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启并配置词库信息。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLoadIkThesaurusRequest()
        request.cluster_id = "{cluster_id}"
        request.body = LoadCustomThesaurusReq(
            synonym_object="word/synonym.txt",
            stop_object="word/stop.txt",
            main_object="word/main.txt",
            bucket_name="test-bucket"
        )
        response = client.create_load_ik_thesaurus(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启并配置词库信息。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateLoadIkThesaurusRequest{}
    request.ClusterId = "{cluster_id}"
    synonymObjectLoadCustomThesaurusReq:= "word/synonym.txt"
    stopObjectLoadCustomThesaurusReq:= "word/stop.txt"
    mainObjectLoadCustomThesaurusReq:= "word/main.txt"
    request.Body = &model.LoadCustomThesaurusReq{
        SynonymObject: &synonymObjectLoadCustomThesaurusReq,
        StopObject: &stopObjectLoadCustomThesaurusReq,
        MainObject: &mainObjectLoadCustomThesaurusReq,
        BucketName: "test-bucket",
    }
    response, err := client.CreateLoadIkThesaurus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	表明服务端能被请求访问到，但是不能理解用户的请求。

错误码

请参见**错误码**。

4.2.2 查询自定义词库状态

功能介绍

该接口用于查询自定义词库的加载状态。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/thesaurus

表 4-130 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定需查询词库状态的集群ID。

请求参数

无

响应参数

状态码： 200

表 4-131 响应 Body 参数

参数	参数类型	描述
status	String	加载状态。 ● Loaded表示加载成功。 ● Loading表示正在加载中。 ● Failed表示加载失败。
bucket	String	最近一次更新存放词库文件的OBS桶。
mainObj	String	主词词库文件对象。
stopObj	String	停词词库文件对象。
synonymObj	String	同义词词库文件对象。
staticMainObj	String	静态主词词库文件对象
staticStopObj	String	静态主词词库文件对象
extraMainObj	String	Extra主词词库库文件对象
extraStopObj	String	Extra停词词库对象
updateTime	String	词库最近更新时间。
updateDetails	String	更新详情。
clusterId	String	指定配置自定义词库的集群ID。
operateStatus	String	操作状态。
id	String	词库的ID。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "status" : "Loaded",
  "operateStatus" : "success",
  "id" : "e766bd5a-79b0-4d1a-8402-fdeb017a36d1",
  "bucket" : "test-bucket",
  "mainObj" : "word/main.txt",
  "stopObj" : "word/stop.txt",
  "synonymObj" : "word/synonym.txt",
  "staticMainObj" : "Unused",
  "staticStopObj" : "Unused",
  "extraMainObj" : "Unused",
  "extraStopObj" : "Unused",
  "updateTime" : 1521184757000,
  "updateDetails" : "allinstancesareloadedsuccessfully.",
  "clusterId" : "ea244205-d641-45d9-9dcb-ab2236bcd07e"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowIkThesaurusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowIkThesaurusRequest request = new ShowIkThesaurusRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowIkThesaurusResponse response = client.showIkThesaurus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowIkThesaurusRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_ik_thesaurus(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowIkThesaurusRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowIkThesaurus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
500	表明服务端能被请求访问到，但是不能理解用户的请求。

错误码

请参见**错误码**。

4.2.3 删除自定义词库

功能介绍

该接口用于清空自定义词库（主词词库、停词词库、同义词词库）。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/thesaurus

表 **4-132** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

参数	是否必选	参数类型	描述
cluster_id	是	String	指定要删除自定义词库的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteIkThesaurusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteIkThesaurusRequest request = new DeleteIkThesaurusRequest();
        request.withClusterId("{cluster_id}");
        try {
            DeleteIkThesaurusResponse response = client.deleteIkThesaurus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteIkThesaurusRequest()
        request.cluster_id = "{cluster_id}"
        response = client.delete_ik_thesaurus(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteIkThesaurusRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.DeleteIkThesaurus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	表明服务端能被请求访问到，但是不能理解用户的请求。

错误码

请参见**错误码**。

4.3 Kibana 公网访问接口

4.3.1 开启 Kibana 公网访问

功能介绍

该接口用于开启Kibana公网访问。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/publickibana/open

表 4-133 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待开启kibana公网访问的集群ID。

请求参数

表 4-134 请求 Body 参数

参数	是否必选	参数类型	描述
eipSize	是	Integer	带宽。单位：Mbit/s
elbWhiteList	是	**StartKibanaPublicReqElbWhitelist** object	elb白名单。

参数	是否必选	参数类型	描述
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 **4-135** StartKibanaPublicReqElbWhitelist

参数	是否必选	参数类型	描述
enableWhiteList	是	Boolean	是否开启白名单。 ● true：开启白名单。 ● false：关闭白名单。
whiteList	是	String	白名单。

响应参数

无

请求示例

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
publickibana/open

{
  "eipSize" : 5,
  "elbWhiteList" : {
    "enableWhiteList" : true,
    "whiteList" : "192.168.0.xx"
  },
  "isAutoPay" : 1
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartKibanaPublicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartKibanaPublicRequest request = new StartKibanaPublicRequest();
        request.withClusterId("{cluster_id}");
        StartKibanaPublicReq body = new StartKibanaPublicReq();
        StartKibanaPublicReqElbWhitelist elbWhiteListbody = new StartKibanaPublicReqElbWhitelist();
        elbWhiteListbody.withEnableWhiteList(true)
            .withWhiteList("192.168.0.xx");
        body.withIsAutoPay(1);
        body.withElbWhiteList(elbWhiteListbody);
        body.withEipSize(5);
        request.withBody(body);
        try {
            StartKibanaPublicResponse response = client.startKibanaPublic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartKibanaPublicRequest()
        request.cluster_id = "{cluster_id}"
        elbWhiteListbody = StartKibanaPublicReqElbWhitelist(
            enable_white_list=True,
            white_list="192.168.0.xx"
        )
        request.body = StartKibanaPublicReq(
            is_auto_pay=1,
            elb_white_list=elbWhiteListbody,
            eip_size=5
        )
        response = client.start_kibana_public(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartKibanaPublicRequest{}
    request.ClusterId = "{cluster_id}"
    elbWhiteListbody := &model.StartKibanaPublicReqElbWhitelist{
        EnableWhiteList: true,
        WhiteList: "192.168.0.xx",
    }
    isAutoPayStartKibanaPublicReq:= int32(1)
    request.Body = &model.StartKibanaPublicReq{
        IsAutoPay: &isAutoPayStartKibanaPublicReq,
        ElbWhiteList: elbWhiteListbody,
        EipSize: int32(5),
    }
    response, err := client.StartKibanaPublic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.3.2 关闭 Kibana 公网访问

功能介绍

该接口用于关闭Kibana公网访问。包周期类型集群不支持通过api进行关闭Kibana公网访问。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/publickibana/close

表 **4-136** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待关闭的kibana公网访问的集群ID。

请求参数

表 **4-137** 请求 Body 参数

参数	是否必选	参数类型	描述
eipSize	否	Integer	带宽。单位：Mbit/s
elbWhiteList	否	**StartKibanaPublicReqElbWhitelist** object	elb白名单。

表 **4-138** StartKibanaPublicReqElbWhitelist

参数	是否必选	参数类型	描述
enableWhiteList	是	Boolean	是否开启白名单。 ● true：开启白名单。 ● false：关闭白名单。
whiteList	是	String	白名单。

响应参数

无

请求示例

关闭Kibana公网访问。

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
publickibana/close

{
  "eipSize" : 5,
  "elbWhiteList" : {
    "enableWhiteList" : true,
    "whiteList" : "192.168.0.xx"
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

关闭Kibana公网访问。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateCloseKibanaSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateCloseKibanaRequest request = new UpdateCloseKibanaRequest();
        request.withClusterId("{cluster_id}");
        CloseKibanaPublicReq body = new CloseKibanaPublicReq();
        StartKibanaPublicReqElbWhitelist elbWhiteListbody = new StartKibanaPublicReqElbWhitelist();
        elbWhiteListbody.withEnableWhiteList(true)
            .withWhiteList("192.168.0.xx");
        body.withElbWhiteList(elbWhiteListbody);
        body.withEipSize(5);
        request.withBody(body);
        try {
            UpdateCloseKibanaResponse response = client.updateCloseKibana(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

关闭Kibana公网访问。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateCloseKibanaRequest()
        request.cluster_id = "{cluster_id}"
        elbWhiteListbody = StartKibanaPublicReqElbWhitelist(
            enable_white_list=True,
            white_list="192.168.0.xx"
        )
        request.body = CloseKibanaPublicReq(
            elb_white_list=elbWhiteListbody,
            eip_size=5
        )
        response = client.update_close_kibana(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

关闭Kibana公网访问。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateCloseKibanaRequest{}
    request.ClusterId = "{cluster_id}"
    elbWhiteListbody := &model.StartKibanaPublicReqElbWhitelist{
        EnableWhiteList: true,
        WhiteList: "192.168.0.xx",
    }
    eipSizeCloseKibanaPublicReq:= int32(5)
    request.Body = &model.CloseKibanaPublicReq{
        ElbWhiteList: elbWhiteListbody,
        EipSize: &eipSizeCloseKibanaPublicReq,
    }
    response, err := client.UpdateCloseKibana(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.3.3 修改 Kibana 公网带宽

功能介绍

该接口用于修改Kibana公网带宽。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/publickibana/bandwidth

表 4-139 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待修改kibana公网带宽的集群ID。

请求参数

表 **4-140** 请求 Body 参数

参数	是否必选	参数类型	描述
bandWidth	是	**UpdatePublicKibanaBandwidthReqBandWidth** object	带宽。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 **4-141** UpdatePublicKibanaBandwidthReqBandWidth

参数	是否必选	参数类型	描述
size	是	Integer	修改后的带宽大小。

响应参数

无

请求示例

修改Kibana公网带宽。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
publickibana/bandwidth

{
  "bandWidth" : {
    "size" : 5
  },
  "isAutoPay" : 1
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改Kibana公网带宽。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateAlterKibanaSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateAlterKibanaRequest request = new UpdateAlterKibanaRequest();
        request.withClusterId("{cluster_id}");
        UpdatePublicKibanaBandwidthReq body = new UpdatePublicKibanaBandwidthReq();
        UpdatePublicKibanaBandwidthReqBandWidth bandWidthbody = new
UpdatePublicKibanaBandwidthReqBandWidth();
        bandWidthbody.withSize(5);
        body.withIsAutoPay(1);
        body.withBandWidth(bandWidthbody);
        request.withBody(body);
        try {
            UpdateAlterKibanaResponse response = client.updateAlterKibana(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改Kibana公网带宽。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateAlterKibanaRequest()
        request.cluster_id = "{cluster_id}"
        bandWidthbody = UpdatePublicKibanaBandwidthReqBandWidth(
            size=5
        )
        request.body = UpdatePublicKibanaBandwidthReq(
            is_auto_pay=1,
            band_width=bandWidthbody
        )
        response = client.update_alter_kibana(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改Kibana公网带宽。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAlterKibanaRequest{}
    request.ClusterId = "{cluster_id}"
    bandWidthbody := &model.UpdatePublicKibanaBandwidthReqBandWidth{
        Size: int32(5),
    }
    isAutoPayUpdatePublicKibanaBandwidthReq:= int32(1)
    request.Body = &model.UpdatePublicKibanaBandwidthReq{
        IsAutoPay: &isAutoPayUpdatePublicKibanaBandwidthReq,
        BandWidth: bandWidthbody,
    }
    response, err := client.UpdateAlterKibana(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.3.4 修改 Kibana 公网访问控制

功能介绍

该接口通过修改kibana白名单，修改kibana的访问权限。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/publickibana/whitelist/update

表 4-142 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定修改kibana的访问权限的集群ID。

请求参数

表 4-143 请求 Body 参数

参数	是否必选	参数类型	描述
whiteList	是	String	允许kibana公网访问的白名单ip或网段，以逗号隔开，不可重复。

响应参数

无

请求示例

修改后的白名单，即最新的白名单

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/5c77b71c-5b35-4f50-8984-76387e42451a/
publickibana/whitelist/update

{
  "whiteList" : "192.168.0.21"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改后的白名单，即最新的白名单

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdatePublicKibanaWhitelistSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdatePublicKibanaWhitelistRequest request = new UpdatePublicKibanaWhitelistRequest();
        request.withClusterId("{cluster_id}");
        UpdatePublicKibanaWhitelistReq body = new UpdatePublicKibanaWhitelistReq();
        body.withWhiteList("192.168.0.21");
        request.withBody(body);
        try {
            UpdatePublicKibanaWhitelistResponse response = client.updatePublicKibanaWhitelist(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改后的白名单，即最新的白名单

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdatePublicKibanaWhitelistRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdatePublicKibanaWhitelistReq(
            white_list="192.168.0.21"
        )
        response = client.update_public_kibana_whitelist(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改后的白名单，即最新的白名单

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePublicKibanaWhitelistRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.UpdatePublicKibanaWhitelistReq{
        WhiteList: "192.168.0.21",
    }
    response, err := client.UpdatePublicKibanaWhitelist(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.3.5 关闭 Kibana 公网访问控制

功能介绍

该接口用于关闭Kibana公网访问控制。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/publickibana/whitelist/close

表 4-144 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定关闭Kibana公网访问控制的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopPublicKibanaWhitelistSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopPublicKibanaWhitelistRequest request = new StopPublicKibanaWhitelistRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopPublicKibanaWhitelistResponse response = client.stopPublicKibanaWhitelist(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopPublicKibanaWhitelistRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_public_kibana_whitelist(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopPublicKibanaWhitelistRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopPublicKibanaWhitelist(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4 Logstash 接口

4.4.1 创建配置文件

功能介绍

该接口用于创建配置文件。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/submit

表 4-145 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定创建配置文件的集群ID。

请求参数

表 **4-146** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。4～32个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。
confContent	是	String	配置文件内容。
setting	是	**setting** object	配置实体对象。
sensitive_words	否	Array of strings	敏感字符替换，输入需要隐藏的敏感字串列表。配置隐藏字符串列表后，在返回的配置内容中，会将所有在列表中的字串隐藏为***（列表最大支持20条，单个字串最大长度512字节）。

表 **4-147** setting

参数	是否必选	参数类型	描述
workers	否	Integer	并行执行管道的Filters+Outputs阶段的工作线程数，默认值为CPU核数。
batchSize	否	Integer	单个工作线程在尝试执行其Filters和Outputs之前将从inputs收集的最大事件数，该值较大通常更有效，但会增加内存开销，默认为125。
batchDelayMs	否	Integer	每个event被pipeline调度等待的最小时间。 单位毫秒。
queueType	是	String	用于事件缓冲的内部队列模型。memory为基于内存的传统队列，persisted为基于磁盘的ACKed持久化队列，默认值为memory。
queueCheckPointWrites	否	Integer	如果使用持久化队列，则表示强制执行检查点之前写入的最大事件数，默认值为1024。
queueMaxBytesMb	否	Integer	如果使用持久化队列，则表示持久化队列的总容量（以兆字节MB为单位），确保磁盘的容量大于该值，默认值为1024。

响应参数

无

请求示例

配置实体对象。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
lgsconf/submit

{
  "name" : "conf1",
  "confContent" : "xxxxxxxx",
  "setting" : {
    "workers" : 2,
    "batchSize" : 125,
    "batchDelayMs" : 50,
    "queueType" : "persisted",
    "queueCheckPointWrites" : 1024,
    "queueMaxBytesMb" : 1024
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

配置实体对象。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateCnfSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);
        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateCnfRequest request = new CreateCnfRequest();
        request.withClusterId("{cluster_id}");
        CreateCnfReq body = new CreateCnfReq();
        Setting settingbody = new Setting();
        settingbody.withWorkers(2)
            .withBatchSize(125)
            .withBatchDelayMs(50)
            .withQueueType("persisted")
            .withQueueCheckPointWrites(1024)
            .withQueueMaxBytesMb(1024);
        body.withSetting(settingbody);
        body.withConfContent("xxxxxxxx");
        body.withName("conf1");
        request.withBody(body);
        try {
            CreateCnfResponse response = client.createCnf(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

配置实体对象。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateCnfRequest()
        request.cluster_id = "{cluster_id}"
        settingbody = Setting(
            workers=2,
            batch_size=125,
            batch_delay_ms=50,
            queue_type="persisted",
            queue_check_point_writes=1024,
            queue_max_bytes_mb=1024
        )
        request.body = CreateCnfReq(
            setting=settingbody,
            conf_content="xxxxxxxx",
            name="conf1"
        )
        response = client.create_cnf(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

配置实体对象。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateCnfRequest{}
    request.ClusterId = "{cluster_id}"
    workersSetting:= int32(2)
    batchSizeSetting:= int32(125)
    batchDelayMsSetting:= int32(50)
    queueCheckPointWritesSetting:= int32(1024)
    queueMaxBytesMbSetting:= int32(1024)
    settingbody := &model.Setting{
        Workers: &workersSetting,
        BatchSize: &batchSizeSetting,
        BatchDelayMs: &batchDelayMsSetting,
        QueueType: "persisted",
        QueueCheckPointWrites: &queueCheckPointWritesSetting,
        QueueMaxBytesMb: &queueMaxBytesMbSetting,
    }
    request.Body = &model.CreateCnfReq{
        Setting: settingbody,
        ConfContent: "xxxxxxxx",
        Name: "conf1",
    }
    response, err := client.CreateCnf(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.2 更新配置文件

功能介绍

该接口用于更新配置文件。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/update

表 **4-148** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定更新配置文件的集群ID。

请求参数

表 **4-149** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。4～32个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。
confContent	是	String	配置文件内容。
setting	是	**setting** object	配置实体对象。
sensitive_words	否	Array of strings	敏感字符替换，输入需要隐藏的敏感字串列表。配置隐藏字符串列表后，在返回的配置内容中，会将所有在列表中的字串隐藏为***（列表最大支持20条，单个字串最大长度512字节）。

表 **4-150** setting

参数	是否必选	参数类型	描述
workers	否	Integer	并行执行管道的Filters+Outputs阶段的工作线程数，默认值为CPU核数。
batchSize	否	Integer	单个工作线程在尝试执行其Filters和Outputs之前将从inputs收集的最大事件数，该值较大通常更有效，但会增加内存开销，默认为125。
batchDelayMs	否	Integer	每个event被pipeline调度等待的最小时间。 单位毫秒。
queueType	是	String	用于事件缓冲的内部队列模型。memory为基于内存的传统队列，persisted为基于磁盘的ACKed持久化队列，默认值为memory。

参数	是否必选	参数类型	描述
queueCheckPointWrites	否	Integer	如果使用持久化队列，则表示强制执行检查点之前写入的最大事件数，默认值为1024。
queueMaxBytesMb	否	Integer	如果使用持久化队列，则表示持久化队列的总容量（以兆字节MB为单位），确保磁盘的容量大于该值，默认值为1024。

响应参数

无

请求示例

更新配置文件。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
lgsconf/update

{
  "name" : "conf1",
  "confContent" : "xxxxxxxx",
  "setting" : {
    "workers" : 2,
    "batchSize" : 125,
    "batchDelayMs" : 50,
    "queueType" : "persisted",
    "queueCheckPointWrites" : 1024,
    "queueMaxBytesMb" : 1024
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

更新配置文件。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;
```

```
public class UpdateCnfSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateCnfRequest request = new UpdateCnfRequest();
        request.withClusterId("{cluster_id}");
        CreateCnfReq body = new CreateCnfReq();
        Setting settingbody = new Setting();
        settingbody.withWorkers(2)
            .withBatchSize(125)
            .withBatchDelayMs(50)
            .withQueueType("persisted")
            .withQueueCheckPointWrites(1024)
            .withQueueMaxBytesMb(1024);
        body.withSetting(settingbody);
        body.withConfContent("xxxxxxxx");
        body.withName("conf1");
        request.withBody(body);
        try {
            UpdateCnfResponse response = client.updateCnf(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新配置文件。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateCnfRequest()
        request.cluster_id = "{cluster_id}"
        settingbody = Setting(
            workers=2,
            batch_size=125,
            batch_delay_ms=50,
            queue_type="persisted",
            queue_check_point_writes=1024,
            queue_max_bytes_mb=1024
        )
        request.body = CreateCnfReq(
            setting=settingbody,
            conf_content="xxxxxxxx",
            name="conf1"
        )
        response = client.update_cnf(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新配置文件。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
```

```
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateCnfRequest{}
    request.ClusterId = "{cluster_id}"
    workersSetting:= int32(2)
    batchSizeSetting:= int32(125)
    batchDelayMsSetting:= int32(50)
    queueCheckPointWritesSetting:= int32(1024)
    queueMaxBytesMbSetting:= int32(1024)
    settingbody := &model.Setting{
        Workers: &workersSetting,
        BatchSize: &batchSizeSetting,
        BatchDelayMs: &batchDelayMsSetting,
        QueueType: "persisted",
        QueueCheckPointWrites: &queueCheckPointWritesSetting,
        QueueMaxBytesMb: &queueMaxBytesMbSetting,
    }
    request.Body = &model.CreateCnfReq{
        Setting: settingbody,
        ConfContent: "xxxxxxxx",
        Name: "conf1",
    }
    response, err := client.UpdateCnf(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.4.3 启动 pipeline 迁移数据

功能介绍

该接口用于启动pipeline迁移数据。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/start

表 4-151 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定开启pipeline的集群ID。

请求参数

表 4-152 请求 Body 参数

参数	是否必选	参数类型	描述
keepAlive	否	Boolean	是否保持常驻。- true：开启保持常驻。- false：关闭保持常驻。开启“保持常驻”适用于需要长期运行的业务。开启“保持常驻”以后，将会在每个节点上面配置一个守护进程，当logstash服务出现故障的时候，会主动拉起并修复。“保持常驻”不适用于短期运行的业务，因为多次主动拉起logstash服务会导致数据迁移重复。
names	是	Array of strings	配置文件名称。

响应参数

无

请求示例

启动pipeline迁移数据。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/lgsconf/start

{
  "keepAlive" : false,
  "names" : [ "confName1", "confName2", "confName3" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

启动pipeline迁移数据。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class StartPipelineSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartPipelineRequest request = new StartPipelineRequest();
        request.withClusterId("{cluster_id}");
        StartPipelineReq body = new StartPipelineReq();
        List<String> listbodyNames = new ArrayList<>();
        listbodyNames.add("confName1");
        listbodyNames.add("confName2");
        listbodyNames.add("confName3");
        body.withNames(listbodyNames);
        body.withKeepAlive(false);
        request.withBody(body);
        try {
            StartPipelineResponse response = client.startPipeline(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

启动pipeline迁移数据。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartPipelineRequest()
        request.cluster_id = "{cluster_id}"
        listNamesbody = [
            "confName1",
            "confName2",
            "confName3"
        ]
        request.body = StartPipelineReq(
            names=listNamesbody,
            keep_alive=False
        )
        response = client.start_pipeline(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

启动pipeline迁移数据。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartPipelineRequest{}
    request.ClusterId = "{cluster_id}"
    var listNamesbody = []string{
        "confName1",
        "confName2",
        "confName3",
    }
    keepAliveStartPipelineReq:= false
    request.Body = &model.StartPipelineReq{
        Names: listNamesbody,
        KeepAlive: &keepAliveStartPipelineReq,
    }
    response, err := client.StartPipeline(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.4 停止 pipeline 迁移数据

功能介绍

该接口用于停止pipeline迁移数据。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/stop

表 4-153 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定停止pipeline的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopPipelineSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopPipelineRequest request = new StopPipelineRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopPipelineResponse response = client.stopPipeline(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopPipelineRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_pipeline(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopPipelineRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopPipeline(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.5 添加到自定义模板

功能介绍

该接口用于添加到自定义模板。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/favorite

表 **4-154** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定添加自定义模板的集群ID。

请求参数

表 **4-155** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。
template	是	**AddFavoriteReqTemplate** object	自定义模板实体对象。

表 **4-156** AddFavoriteReqTemplate

参数	是否必选	参数类型	描述
templateName	是	String	自定义模板名称。
desc	否	String	模板描述。

响应参数

无

请求示例

配置自定义模板。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
lgsconf/favorite

{
  "name" : "conf1",
  "template" : {
    "templateName" : "template1",
    "desc" : "自定义模板"
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

配置自定义模板。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class AddFavoriteSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        AddFavoriteRequest request = new AddFavoriteRequest();
        request.withClusterId("{cluster_id}");
        AddFavoriteReq body = new AddFavoriteReq();
        AddFavoriteReqTemplate templatebody = new AddFavoriteReqTemplate();
        templatebody.withTemplateName("template1")
            .withDesc("自定义模板");
        body.withTemplate(templatebody);
        body.withName("conf1");
        request.withBody(body);
        try {
            AddFavoriteResponse response = client.addFavorite(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

配置自定义模板。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddFavoriteRequest()
        request.cluster_id = "{cluster_id}"
        templatebody = AddFavoriteReqTemplate(
            template_name="template1",
            desc="自定义模板"
        )
        request.body = AddFavoriteReq(
            template=templatebody,
            name="conf1"
        )
        response = client.add_favorite(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

配置自定义模板。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddFavoriteRequest{}
    request.ClusterId = "{cluster_id}"
    descTemplate:= "自定义模板"
    templatebody := &model.AddFavoriteReqTemplate{
        TemplateName: "template1",
        Desc: &descTemplate,
    }
    request.Body = &model.AddFavoriteReq{
        Template: templatebody,
        Name: "conf1",
    }
    response, err := client.AddFavorite(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.6 连通性测试

功能介绍

该接口用于连通性测试。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/checkconnection

表 4-157 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待测试的集群ID。

请求参数

表 4-158 请求 Body 参数

参数	是否必选	参数类型	描述
addressAndPorts	是	Array of **AddressAndPorts** objects	地址和端口列表。

表 4-159 AddressAndPorts

参数	是否必选	参数类型	描述
address	是	String	IP地址或域名。
port	否	Integer	端口号。

响应参数

状态码： 200

表 **4-160** 响应 Body 参数

参数	参数类型	描述
result	Array of **result** objects	连通性测试结果。

表 **4-161** result

参数	参数类型	描述
address	String	IP地址或域名。
port	Integer	端口号。
status	Integer	测试结果。 ● 1：表示连接成功。 ● 0：表示地址不可达。 ● 2：表示端口不可达。 ● 3：表示域名无法解析。 ● -2：表示位置错误。

请求示例

配置端口测试连通性。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
checkconnection

{
  "addressAndPorts" : [ {
    "address" : "10.0.0.83",
    "port" : "9200"
  } ]
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "result" : [ {
    "address" : "10.0.0.83",
    "port" : "9200",
    "status" : 1
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

配置端口测试连通性。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class StartConnectivityTestSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartConnectivityTestRequest request = new StartConnectivityTestRequest();
        request.withClusterId("{cluster_id}");
        StartConnectivityTestReq body = new StartConnectivityTestReq();
        List<AddressAndPorts> listbodyAddressAndPorts = new ArrayList<>();
        listbodyAddressAndPorts.add(
            new AddressAndPorts()
                .withAddress("10.0.0.83")
                .withPort(9200)
        );
        body.withAddressAndPorts(listbodyAddressAndPorts);
        request.withBody(body);
        try {
            StartConnectivityTestResponse response = client.startConnectivityTest(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

配置端口测试连通性。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartConnectivityTestRequest()
        request.cluster_id = "{cluster_id}"
        listAddressAndPortsbody = [
            AddressAndPorts(
                address="10.0.0.83",
                port=9200
            )
        ]
        request.body = StartConnectivityTestReq(
            address_and_ports=listAddressAndPortsbody
        )
        response = client.start_connectivity_test(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

配置端口测试连通性。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartConnectivityTestRequest{}
    request.ClusterId = "{cluster_id}"
    portAddressAndPorts:= int32(9200)
    var listAddressAndPortsbody = []model.AddressAndPorts{
        {
            Address: "10.0.0.83",
            Port: &portAddressAndPorts,
        },
    }
    request.Body = &model.StartConnectivityTestReq{
        AddressAndPorts: listAddressAndPortsbody,
    }
    response, err := client.StartConnectivityTest(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.7 查询模板列表

功能介绍

该接口用于查询模板列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/lgsconf/template

表 4-162 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

表 4-163 Query 参数

参数	是否必选	参数类型	描述
type	否	String	模板类型。custom为自定义模板，system为系统模板。不指定查询模板类型默认查找自定义模板和系统模板。

请求参数

无

响应参数

状态码： 200

表 4-164 响应 Body 参数

参数	参数类型	描述
systemTemplates	Array of **systemTemplates** objects	系统模板列表。
customTemplates	Array of **customTemplates** objects	自定义模板列表。

表 **4-165** systemTemplates

参数	参数类型	描述
id	String	配置文件id。
name	String	配置文件名称。
confContent	String	配置文件内容。
desc	String	描述。

表 **4-166** customTemplates

参数	参数类型	描述
id	String	配置文件id。
name	String	配置文件名称。
confContent	String	配置文件内容。
desc	String	描述。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "systemTemplates" : [ {
    "id" : "a03b1791-6dc6-453d-961d-88939adc82aa",
    "name" : "elasticsearch",
    "confContent" : "xxxxxxxx",
    "desc" : "logstash config for es"
  } ],
  "customTemplates" : [ {
    "id" : "2e16fbba-49bf-4690-93af-3f89025c6523",
    "name" : "es_to_https_es",
    "confContent" : "xxxxxxxx",
    "desc" : "es到安全集群es"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListTemplatesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListTemplatesRequest request = new ListTemplatesRequest();
        try {
            ListTemplatesResponse response = client.listTemplates(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTemplatesRequest()
        response = client.list_templates(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListTemplatesRequest{}
    response, err := client.ListTemplates(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.8 查询配置文件列表

功能介绍

该接口用于查询配置文件列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/listconfs

表 4-167 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

无

响应参数

状态码： 200

表 **4-168** 响应 Body 参数

参数	参数类型	描述
confs	Array of **confs** objects	配置文件列表。

表 **4-169** confs

参数	参数类型	描述
name	String	配置文件名称。
status	String	配置文件状态。
confContent	String	配置文件内容。
setting	**setting** object	具体的配置内容对象。
updateAt	String	更新时间。

表 **4-170** setting

参数	参数类型	描述
workers	Integer	并行执行管道的Filters+Outputs阶段的工作线程数，默认值为CPU核数。
batchSize	Integer	单个工作线程在尝试执行其Filters和Outputs之前将从inputs收集的最大事件数，该值较大通常更有效，但会增加内存开销，默认为125。
batchDelayMs	Integer	每个event被pipeline调度等待的最小时间。 单位毫秒。
queueType	String	用于事件缓冲的内部队列模型。memory为基于内存的传统队列，persisted为基于磁盘的ACKed持久化队列，默认值为memory。
queueCheckPointWrites	Integer	如果使用持久化队列，则表示强制执行检查点之前写入的最大事件数，默认值为1024。
queueMaxBytesMb	Integer	如果使用持久化队列，则表示持久化队列的总容量（以兆字节MB为单位），确保磁盘的容量大于该值，默认值为1024。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "confs" : [ {
    "name" : "conf1",
    "status" : "available",
    "confContent" : "xxxxxxxx",
    "setting" : {
      "workers" : 2,
      "batchSize" : 125,
      "batchDelayMs" : 50,
      "queueType" : "persisted",
      "queueCheckPointWrites" : 1024,
      "queueMaxBytesMb" : 1024
    },
    "updateAt" : "2021-09-29T02:44:44"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListConfsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListConfsRequest request = new ListConfsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListConfsResponse response = client.listConfs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListConfsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_confs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListConfsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListConfs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.9 查询 pipeline 列表

功能介绍

该接口用于查询pipeline列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/listpipelines

表 4-171 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

无

响应参数

状态码： 200

表 4-172 响应 Body 参数

参数	参数类型	描述
pipelines	Array of **pipelines** objects	pipeline列表。

表 4-173 pipelines

参数	参数类型	描述
name	String	配置文件名称。
status	String	pipeline状态。
keepAlive	Boolean	是否开启常驻。
events	String	事件只有在“工作中”状态才可以实时查看（需要手动刷新），“已停止”状态请到output端查看迁移数据量。
updateAt	String	更新时间。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "pipelines" : [ {
    "name" : "conf1",
    "status" : "stopped",
    "keepAlive" : false,
    "events" : "xxxxxxxx",
    "updateAt" : "2021-09-29T02:44:44"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListPipelinesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPipelinesRequest request = new ListPipelinesRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListPipelinesResponse response = client.listPipelines(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPipelinesRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_pipelines(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPipelinesRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListPipelines(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.10 查询操作记录

功能介绍

该接口用于查询操作记录。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/listactions

表 4-174 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

无

响应参数

状态码： 200

表 4-175 响应 Body 参数

参数	参数类型	描述
actions	Array of **actions** objects	操作记录列表。

表 4-176 actions

参数	参数类型	描述
id	String	操作记录id。
actionType	String	操作类型。
confContent	String	配置文件内容。
status	String	操作状态。
updateAt	String	更新时间，格式为ISO8601：CCYY-MM-DDThh:mm:ss。
errorMsg	String	错误信息。当操作状态为success时该字段为null。
message	String	内容。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "actions" : [ {
    "id" : "e487cb83-1b30-435d-a11a-def463bbe9aa",
    "actionType" : "stop",
    "confContent" : "xxxxxxxx",
    "status" : "success",
    "updateAt" : "2021-09-29T02:44:44",
    "errorMsg" : "xxxxxxxx",
    "message" : {
      "keepAlive" : "0",
      "confNames" : "conf1"
    }
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListActionsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListActionsRequest request = new ListActionsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListActionsResponse response = client.listActions(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListActionsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_actions(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListActionsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListActions(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.11 查询配置文件内容

功能介绍

该接口用于查询配置文件内容。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/confdetail

表 4-177 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

表 4-178 Query 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。

请求参数

无

响应参数

状态码： 200

表 4-179 响应 Body 参数

参数	参数类型	描述
name	String	配置文件名称。
status	String	配置文件状态。
confContent	String	配置文件内容。
setting	**setting** object	具体的配置内容对象。
updateAt	String	更新时间。

表 **4-180** setting

参数	参数类型	描述
workers	Integer	并行执行管道的Filters+Outputs阶段的工作线程数，默认值为CPU核数。
batchSize	Integer	单个工作线程在尝试执行其Filters和Outputs之前将从inputs收集的最大事件数，该值较大通常更有效，但会增加内存开销，默认为125。
batchDelayMs	Integer	每个event被pipeline调度等待的最小时间。 单位毫秒。
queueType	String	用于事件缓冲的内部队列模型。memory为基于内存的传统队列，persisted为基于磁盘的ACKed持久化队列，默认值为memory。
queueCheckPointWrites	Integer	如果使用持久化队列，则表示强制执行检查点之前写入的最大事件数，默认值为1024。
queueMaxBytesMb	Integer	如果使用持久化队列，则表示持久化队列的总容量（以兆字节MB为单位），确保磁盘的容量大于该值，默认值为1024。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "name" : "conf1",
  "status" : "available",
  "confContent" : "xxxxxxxx",
  "setting" : {
    "workers" : 2,
    "batchSize" : 125,
    "batchDelayMs" : 50,
    "queueType" : "persisted",
    "queueCheckPointWrites" : 1024,
    "queueMaxBytesMb" : 1024
  },
  "updateAt" : "2021-09-29T02:44:44"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowGetConfDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowGetConfDetailRequest request = new ShowGetConfDetailRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowGetConfDetailResponse response = client.showGetConfDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security 
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment 
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this 
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowGetConfDetailRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_get_conf_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowGetConfDetailRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowGetConfDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.12 删除配置文件

功能介绍

删除配置文件。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/delete

表 4-181 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定删除配置文件的集群ID。

请求参数

表 **4-182** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteConfSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteConfRequest request = new DeleteConfRequest();
        request.withClusterId("{cluster_id}");
        DeleteConfReq body = new DeleteConfReq();
        request.withBody(body);
        try {
            DeleteConfResponse response = client.deleteConf(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteConfRequest()
        request.cluster_id = "{cluster_id}"
        request.body = DeleteConfReq(
        )
        response = client.delete_conf(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteConfRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.DeleteConfReq{
    }
    response, err := client.DeleteConf(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.4.13 删除自定义模板

功能介绍

该接口用于删除自定义模板。

调用方法

请参见如何调用API。

URI

DELETE /v1.0/{project_id}/lgsconf/deletetemplate

表 4-183 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。

请求参数

表 4-184 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	模板名称。

响应参数

无

请求示例

删除自定义模板。

```
DELETE /v1.0/6204a5bd270343b5885144cf9c8c158d/lgsconf/deletetemplate

{
  "name" : "template1"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

删除自定义模板。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteTemplateRequest request = new DeleteTemplateRequest();
        DeleteTemplateReq body = new DeleteTemplateReq();
        body.withName("template1");
        request.withBody(body);
        try {
            DeleteTemplateResponse response = client.deleteTemplate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

删除自定义模板。

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteTemplateRequest()
        request.body = DeleteTemplateReq(
            name="template1"
        )
        response = client.delete_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

删除自定义模板。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteTemplateRequest{}
    request.Body = &model.DeleteTemplateReq{
        Name: "template1",
    }
    response, err := client.DeleteTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.14 热启动 pipeline 迁移数据

功能介绍

该接口用于热启动pipeline迁移数据。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/hot-start

表 **4-185** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 **4-186** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。
keep_alive	否	Boolean	热启动操作时，需要与集群已存在管道的 “是否保持常驻” 值保持一致。 ● true: 开启保持常驻。 ● false: 关闭保持常驻。 开启“保持常驻”适用于需要长期运行的业务。开启“保持常驻”以后，将会在每个节点上面配置一个守护进程，当logstash服务出现故障的时候，会主动拉起并修复。“保持常驻”不适用于短期运行的业务，因为多次主动拉起logstash服务会导致数据迁移重复。

响应参数

无

请求示例

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
lgsconf/hot-start

{
  "keep_alive" : false,
  "name" : "confName1"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.15 热停止 pipeline 迁移数据

功能介绍

该接口用于热停止pipeline迁移数据。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/lgsconf/hot-stop

表 4-187 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 **4-188** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。

响应参数

无

请求示例

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
lgsconf/hot-stop

{
  "name" : "confName"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopHotPipelineSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);
        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopHotPipelineRequest request = new StopHotPipelineRequest();
        request.withClusterId("{cluster_id}");
        StopHotPipelineRequestBody body = new StopHotPipelineRequestBody();
        body.withName("confName");
        request.withBody(body);
        try {
            StopHotPipelineResponse response = client.stopHotPipeline(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopHotPipelineRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StopHotPipelineRequestBody(
            name="confName"
        )
        response = client.stop_hot_pipeline(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopHotPipelineRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.StopHotPipelineRequestBody{
        Name: "confName",
    }
    response, err := client.StopHotPipeline(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.16 更新集群路由

功能介绍

该接口用于更新集群路由。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/route

表 4-189 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 4-190 请求 Body 参数

参数	是否必选	参数类型	描述
configtype	是	String	操作类型。add_ip为增加集群路由，del_ip为删除集群路由。
configkey	是	String	路由ip地址，公网源数据所在的服务器ip。不能以0开头。
configvalue	是	String	路由子网掩码。如果上面ip取的是16位，子网掩码可以填255.255.0.0，24位的话子网掩码填255.255.255.0。

响应参数

无

请求示例

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/route

{
  "configtype" : "add_ip",
  "configkey" : "10.5.2.1",
  "configvalue" : "255.255.255.255"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.17 获取集群路由

功能介绍

该接口用于获取集群路由。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/route

表 **4-191** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

无

响应参数

状态码： 200

表 **4-192** 响应 Body 参数

参数	参数类型	描述
routeResps	Array of **RouteRespsResource** objects	路由IP。

表 **4-193** RouteRespsResource

参数	参数类型	描述
ipAddress	String	ip地址。
ipNetMask	String	子网掩码。
updateAt	String	更新时间。

请求示例

获取该集群路由。

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/route
```

响应示例

状态码： 200

请求已成功。

```
{
  "routeResps" : [ {
    "ipAddress" : "1.2.3.65",
    "ipNetMask" : "255.255.255.255",
    "updateAt" : "2023-07-17T08:09:20"
  }, {
    "ipAddress" : "1.1.1.1",
    "ipNetMask" : "255.255.255.255",
    "updateAt" : "2023-07-17T08:08:50"
  } ]
}
```

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.18 上传证书文件

功能介绍

该接口用于上传证书文件。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/certs/upload

表 4-194 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 **4-195** 请求 Body 参数

参数	是否必选	参数类型	描述
bucket_name	是	String	证书文件存放的OBS桶（桶类型必须为标准存储或者低频存储，不支持归档存储）。
certs_object	是	String	证书文件对象。证书文件大小不能超过1M。证书名称在4位到32位之间，必须以字母开头，以（.cer|.crt|.rsa|.jks|.pem|.p10|.pfx|.p12|.csr|.der|.keystore）结尾，可以包含字母、数字、中划线、下划线或者小数点，不能包含其他的特殊字符。

响应参数

无

请求示例

上传证书对象。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/certs/upload

{
  "bucket_name" : "test-bucket",
  "certs_object" : "test/cert.cer"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.4.19 查询证书列表

功能介绍

该接口用于查询证书列表。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/certs

表 4-196 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定待查询的集群ID。

表 4-197 Query 参数

参数	是否必选	参数类型	描述
start	否	String	指定查询起始值，默认值为1，即从第1个证书开始查询。
limit	否	String	指定查询个数，默认值为10，即一次查询10个证书信息。
certsType	否	String	证书类型。defaultCerts为默认证书类型，不指定查询证书类型默认查找自定义证书列表。

请求参数

无

响应参数

无

请求示例

获取集群上传记录。

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/certs
```

响应示例

状态码： 200

请求已成功。

```
{
  "certsRecords" : [ {
    "id" : "2630b1ab-47f8-445c-a4e0-5de74683d767",
    "status" : "available",
    "fileLocation" : "/rds/datastore/logstash/v7.10.0/package/logstash-7.10.0/extend/custom_certs/",
    "fileName" : "xxxx.cer",
    "updateAt" : "2023-06-14T06:54:36"
  } ],
  "totalSize" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListCertsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListCertsRequest request = new ListCertsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListCertsResponse response = client.listCerts(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListCertsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_certs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListCertsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListCerts(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.20 查询证书文件信息

功能介绍

该接口用于查询证书文件信息。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/certs/{cert_id}

表 4-198 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待查询的集群ID。
cert_id	是	String	指定查询的证书文件ID。

请求参数

无

响应参数

状态码： 200

表 4-199 响应 Body 参数

参数	参数类型	描述
id	String	证书id。
fileName	String	证书名称。
fileLocation	String	证书路径。
status	String	证书状态。
updateAt	String	证书上传时间。

请求示例

查询指定证书文件信息。

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/certs/
62967551-4087-402a-b9c5-68b56ac274cb
```

响应示例

状态码： 200

请求已成功。

```
{
  "id" : "636b6718-b8ae-4eb9-bf70-5fed8aa52181",
  "status" : "available",
  "fileLocation" : "/rds/datastore/logstash/vx.xx.xx/package/logstash-x.xx.xx/extend/custom_certs/",
  "fileName" : "ab.txt",
  "updateAt" : "2023-06-20T01:15:30"
}
```

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.21 删除证书文件

功能介绍

该接口用于删除证书文件。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/certs/{cert_id}/delete

表 4-200 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定删除证书文件的集群ID。
cert_id	是	String	指定删除的证书文件ID。

请求参数

无

响应参数

无

请求示例

删除指定集群证书文件。

```
DELTET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/certs/
62967551-4087-402a-b9c5-68b56ac274cb/delete
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.22 删除配置文件 V2

功能介绍

删除配置文件V2。

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/clusters/{cluster_id}/lgsconf/delete

表 4-201 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定删除配置文件的集群ID。

请求参数

表 4-202 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置文件名称。

响应参数

无

请求示例

无

响应示例

无

状态码

状态码	描述
200	请求已成功。

状态码	描述
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.4.23 删除自定义模板 V2

功能介绍

该接口用于删除自定义模板V2。

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/lgsconf/deletetemplate

表 4-203 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

请求参数

表 4-204 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	模板名称。

响应参数

无

请求示例

删除自定义模板。

```
POST /v2.0/6204a5bd270343b5885144cf9c8c158d/lgsconf/deletetemplate

{
  "name" : "template1"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5 日志管理接口

4.5.1 开启日志功能

功能介绍

该接口用于开启日志功能。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/open

表 4-205 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定开启日志的集群ID。

表 4-206 Query 参数

参数	是否必选	参数类型	描述
action	否	String	action支持base_log_collect和real_time_log_collect两种，base就是之前历史的能力，real_time为实时采集能力，默认不传就是base，兼容之前的逻辑

请求参数

表 4-207 请求 Body 参数

参数	是否必选	参数类型	描述
agency	否	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。action不等于real_time_log_collect时必选。
logBasePath	否	String	日志在OBS桶中的备份路径。action不等于real_time_log_collect时必选。
logBucket	否	String	用于存储日志的OBS桶的桶名。action不等于real_time_log_collect时必选。
index_prefix	否	String	保存日志的索引前缀。action等于real_time_log_collect时必选
keep_days	否	Integer	日志保存时间。action等于real_time_log_collect时必选
target_cluster_id	否	String	保存日志的目标集群。action等于real_time_log_collect时必选

响应参数

无

请求示例

- 开启日志功能。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
logs/open

{
  "agency" : "css_obs_agency",
  "logBasePath" : "css/log",
  "logBucket" : "000-words"
}
```

- 开启日志实时采集功能。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
logs/open?action=real_time_log_collect

{
  "index_prefix" : "css_log",
  "keep_days" : 30,
  "target_cluster_id" : "4f3deec3-efa8-4598-bf91-560aad1377a3"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

- 开启日志功能。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartLogsRequest request = new StartLogsRequest();
        request.withClusterId("{cluster_id}");
        StartLogsReq body = new StartLogsReq();
        body.withLogBucket("000-words");
        body.withLogBasePath("css/log");
        body.withAgency("css_obs_agency");
        request.withBody(body);
        try {
            StartLogsResponse response = client.startLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 开启日志实时采集功能。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartLogsRequest request = new StartLogsRequest();
        request.withClusterId("{cluster_id}");
        StartLogsReq body = new StartLogsReq();
        body.withTargetClusterId("4f3deec3-efa8-4598-bf91-560aad1377a3");
        body.withKeepDays(30);
        body.withIndexPrefix("css_log");
        request.withBody(body);
        try {
            StartLogsResponse response = client.startLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 开启日志功能。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartLogsRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartLogsReq(
            log_bucket="000-words",
            log_base_path="css/log",
            agency="css_obs_agency"
        )
        response = client.start_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 开启日志实时采集功能。

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartLogsRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartLogsReq(
            target_cluster_id="4f3deec3-efa8-4598-bf91-560aad1377a3",
            keep_days=30,
            index_prefix="css_log"
        )
        response = client.start_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 开启日志功能。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartLogsRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.StartLogsReq{
        LogBucket: "000-words",
        LogBasePath: "css/log",
        Agency: "css_obs_agency",
    }
    response, err := client.StartLogs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 开启日志实时采集功能。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartLogsRequest{}
    request.ClusterId = "{cluster_id}"
    targetClusterIdStartLogsReq:= "4f3deec3-efa8-4598-bf91-560aad1377a3"
    keepDaysStartLogsReq:= int32(30)
    indexPrefixStartLogsReq:= "css_log"
    request.Body = &model.StartLogsReq{
        TargetClusterId: &targetClusterIdStartLogsReq,
        KeepDays: &keepDaysStartLogsReq,
        IndexPrefix: &indexPrefixStartLogsReq,
    }
    response, err := client.StartLogs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.2 关闭日志功能

功能介绍

该接口用于关闭日志功能。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/logs/close

表 4-208 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定关闭日志的集群ID。

表 **4-209** Query 参数

参数	是否必选	参数类型	描述
action	否	String	action支持base_log_collect和real_time_log_collect两种，base就是之前历史的能力，real_time为实时采集能力，默认不传就是base，兼容之前的逻辑

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
```

```
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopLogsRequest request = new StopLogsRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopLogsResponse response = client.stopLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopLogsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopLogsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopLogs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.5.3 查询作业列表

功能介绍

该接口用于查询具体某个集群的日志任务记录列表。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/logs/records

表 4-210 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定查询集群ID。

表 4-211 Query 参数

参数	是否必选	参数类型	描述
start	否	Integer	指定查询起始值，默认值为1，即从第1个任务开始查询。
limit	否	Integer	指定查询个数，默认值为10，即一次查询10个任务信息。
status	否	String	指定查询的日志任务状态，未设置该参数时，查询集群的全部日志任务记录列表。 FAIL 日志任务失败状态 NOT_FAIL 日志任务运行或成功状态

请求参数

无

响应参数

状态码： 200

表 4-212 响应 Body 参数

参数	参数类型	描述
clusterLogRecord	Array of **clusterLogRecord** objects	集群日志记录实体对象。

表 4-213 clusterLogRecord

参数	参数类型	描述
id	String	日志任务ID，通过系统UUID生成。
clusterId	String	集群ID。
createAt	String	创建时间。格式：Unix时间戳格式。
logPath	String	日志在OBS桶中的备份路径。
status	String	任务状态。 ● RUNNING: 备份行中。 ● SUCCESS: 备份成功。 ● FAIL: 备份失败。
finishedAt	Long	结束时间，当创建未结束时结束时间为null。格式：Unix时间戳格式。
jobTypes	String	任务类型。 ● Manual: 手动备份。 ● Auto： 自动备份。
failedMsg	String	错误信息。当任务状态没有处于失败状态时该字段为null。
jobId	String	任务ID。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "clusterLogRecord" : [ {
    "id" : "d455a541-597e-4846-a6be-baad0ea361b1",
    "clusterId" : "4213d908-f5dc-4633-8401-cfd7175fca0c",
    "createAt" : 1656042837000,
    "logPath" : "css-backup-1610678043608/css/log",
    "status" : "RUNNING",
    "finishedAt" : null,
    "jobTypes" : "Manual",
    "failedMsg" : null,
    "jobId" : "2c9080df7c171342017c5e0884f8011c"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListLogsJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListLogsJobRequest request = new ListLogsJobRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListLogsJobResponse response = client.listLogsJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListLogsJobRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_logs_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListLogsJobRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListLogsJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.4 查询日志基础配置

功能介绍

该接口用于日志基础配置查询。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/logs/settings

表 **4-214** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

表 **4-215** Query 参数

参数	是否必选	参数类型	描述
action	否	String	action支持base_log_collect和real_time_log_collect两种，base就是之前历史的能力，real_time为实时采集能力，默认不传就是base，兼容之前的逻辑

请求参数

无

响应参数

状态码： 200

表 **4-216** 响应 Body 参数

参数	参数类型	描述
logConfiguration	**logConfiguration** object	日志配置实体对象。
realTimeLogCollectRecord	**realTimeLogCollect** object	日志实时采集配置。

表 **4-217** logConfiguration

参数	参数类型	描述
id	String	日志备份ID，通过系统UUID生成。
clusterId	String	集群ID。
obsBucket	String	用于存储日志的OBS桶的桶名。
agency	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。

参数	参数类型	描述
updateAt	Long	更新时间。格式为：Unix时间戳格式。
basePath	String	日志在OBS桶中的备份路径。
autoEnable	Boolean	自动备份开关。 ● true: 自动备份开启。 ● false: 自动备份关闭。
period	String	自动备份日志开始时间。当autoEnable为false时该字段为null。格式为：格林威治标准时间。
logSwitch	Boolean	日志开关。 ● true: 日志开启。 ● false: 日志关闭。

表 4-218 realTimeLogCollect

参数	参数类型	描述
id	String	日志采集ID，通过系统UUID生成。
clusterId	String	集群id
indexPrefix	String	日志保存索引的前缀。
keepDays	Integer	日志保存时间。
targetClusterId	String	保存日志的目标集群ID。
status	String	日志实时采集任务状态。
createAt	Long	日志实时采集任务开启时间。
updateAt	Long	日志实时采集任务更新时间。

请求示例

无

响应示例

状态码： 200

请求已成功。

- 日志实时采集响应示例。

```
{
  "logConfiguration" : null,
  "realTimeLogCollectRecord" : {
    "id" : "17939b7b-5a93-4ca5-8d3c-b9f2d0e715b4",
    "clusterId" : "e3201ceb-1a3e-49f3-bb2f-23a816440b20",
    "keepDays" : 30,
    "updateAt" : 1717666418870,
    "createAt" : 1717666405897,
    "targetClusterId" : "8c19644b-f2ef-44fc-abef-230a4c578ce7",
    "indexPrefix" : "aaa",
    "status" : "200"
  }
}
```

- 日志备份响应示例。

```
{
  "logConfiguration" : {
    "id" : "00032118-aff5-40e8-b19a-dd4bb576e572",
    "clusterId" : "e3201ceb-1a3e-49f3-bb2f-23a816440b20",
    "obsBucket" : "css-autobk-notdel-cn-north-7",
    "agency" : "css_obs_agency",
    "updateAt" : 1639624882000,
    "basePath" : "css/log",
    "autoEnable" : false,
    "period" : null,
    "logSwitch" : false
  },
  "realTimeLogCollectRecord" : null
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowGetLogSettingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowGetLogSettingRequest request = new ShowGetLogSettingRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowGetLogSettingResponse response = client.showGetLogSetting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowGetLogSettingRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_get_log_setting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowGetLogSettingRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowGetLogSetting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.5 修改日志基础配置

功能介绍

该接口用于修改日志基础配置。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/settings

表 4-219 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定更改日志基础配置的集群ID。

表 4-220 Query 参数

参数	是否必选	参数类型	描述
action	否	String	action支持base_log_collect和real_time_log_collect两种，base就是之前历史的能力，real_time为实时采集能力，默认不传就是base，兼容之前的逻辑

请求参数

表 4-221 请求 Body 参数

参数	是否必选	参数类型	描述
agency	是	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。 action不等于real_time_log_collect时必填。
logBasePath	是	String	日志在OBS桶中的备份路径。 action不等于real_time_log_collect时必填。

参数	是否必选	参数类型	描述
logBucket	是	String	用于存储日志的OBS桶的桶名。action不等于real_time_log_collect时必填。
index_prefix	否	String	日志保存的索引前缀。action等于real_time_log_collect时必填。
keep_days	否	Integer	日志保存时间。action等于real_time_log_collect时必填。
target_cluster_id	否	String	保存日志的目标集群。action等于real_time_log_collect时必填。

响应参数

无

请求示例

- 修改日志基础配置。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
logs/settings

{
  "agency" : "css_obs_agency",
  "logBasePath" : "css/log",
  "logBucket" : "000-words"
}
```

- 更新日志实时采集配置

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
logs/settings?action=real_time_log_collect

{
  "index_prefix" : "css_log",
  "keep_days" : 30,
  "target_cluster_id" : "4f3deec3-efa8-4598-bf91-560aad1377a3",
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

- 修改日志基础配置。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateLogSettingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateLogSettingRequest request = new UpdateLogSettingRequest();
        request.withClusterId("{cluster_id}");
        UpdateLogSettingReq body = new UpdateLogSettingReq();
        body.withLogBucket("000-words");
        body.withLogBasePath("css/log");
        body.withAgency("css_obs_agency");
        request.withBody(body);
        try {
            UpdateLogSettingResponse response = client.updateLogSetting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 更新日志实时采集配置

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateLogSettingSolution {
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateLogSettingRequest request = new UpdateLogSettingRequest();
        request.withClusterId("{cluster_id}");
        body.withUpdateLogSettingReq("{\"index_prefix\":\"css_log\",\"target_cluster_id\":\"4f3deec3-efa8-4598-bf91-560aad1377a3\",\"keep_days\":30}");
        request.withBody(listUpdateLogSettingReqUpdateLogSettingReq);
        try {
            UpdateLogSettingResponse response = client.updateLogSetting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 修改日志基础配置。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateLogSettingRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdateLogSettingReq(
            log_bucket="000-words",
            log_base_path="css/log",
            agency="css_obs_agency"
        )
        response = client.update_log_setting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 更新日志实时采集配置

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateLogSettingRequest()
        request.cluster_id = "{cluster_id}"
        request.body = listUpdateLogSettingReqUpdateLogSettingReq
        response = client.update_log_setting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 修改日志基础配置。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateLogSettingRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.UpdateLogSettingReq{
        LogBucket: "000-words",
        LogBasePath: "css/log",
        Agency: "css_obs_agency",
    }
    response, err := client.UpdateLogSetting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 更新日志实时采集配置

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
```

```
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateLogSettingRequest{}
    request.ClusterId = "{cluster_id}"
    var updateLogSettingReqUpdateLogSettingReq interface{} = "{\"index_prefix\":\"css_log
\",\"target_cluster_id\":\"4f3deec3-efa8-4598-bf91-560aad1377a3\",\"keep_days\":30}"
    request.Body = listUpdateLogSettingReqUpdateLogSettingReq
    response, err := client.UpdateLogSetting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.6 开启日志自动备份策略

功能介绍

该接口用于日志自动备份策略开启。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/policy/update

表 **4-222** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定开启日志备份的集群ID。

请求参数

表 **4-223** 请求 Body 参数

参数	是否必选	参数类型	描述
period	是	String	备份开始时间。格式：格林威治标准时间。

响应参数

无

请求示例

开启日志自动备份策略。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/logs/
policy/update

{
  "period" : "00:00 GMT+08:00"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启日志自动备份策略。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartLogAutoBackupPolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartLogAutoBackupPolicyRequest request = new StartLogAutoBackupPolicyRequest();
        request.withClusterId("{cluster_id}");
        StartLogAutoBackupPolicyReq body = new StartLogAutoBackupPolicyReq();
        body.withPeriod("00:00 GMT+08:00");
        request.withBody(body);
        try {
            StartLogAutoBackupPolicyResponse response = client.startLogAutoBackupPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启日志自动备份策略。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartLogAutoBackupPolicyRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartLogAutoBackupPolicyReq(
            period="00:00 GMT+08:00"
        )
        response = client.start_log_auto_backup_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启日志自动备份策略。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartLogAutoBackupPolicyRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.StartLogAutoBackupPolicyReq{
        Period: "00:00 GMT+08:00",
    }
    response, err := client.StartLogAutoBackupPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.7 关闭日志自动备份策略

功能介绍

该接口用于日志自动备份策略关闭。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/logs/policy/close

表 4-224 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定关闭日志自动备份的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopLogAutoBackupPolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopLogAutoBackupPolicyRequest request = new StopLogAutoBackupPolicyRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopLogAutoBackupPolicyResponse response = client.stopLogAutoBackupPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopLogAutoBackupPolicyRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_log_auto_backup_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopLogAutoBackupPolicyRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopLogAutoBackupPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.8 备份日志

功能介绍

该接口用于备份日志。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/collect

表 4-225 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定待备份日志的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateLogBackupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateLogBackupRequest request = new CreateLogBackupRequest();
        request.withClusterId("{cluster_id}");
        try {
            CreateLogBackupResponse response = client.createLogBackup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLogBackupRequest()
        request.cluster_id = "{cluster_id}"
        response = client.create_log_backup(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateLogBackupRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.CreateLogBackup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.9 查询日志

功能介绍

该接口用于查询日志信息。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/search

表 4-226 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

表 4-227 请求 Body 参数

参数	是否必选	参数类型	描述
instanceName	是	String	节点名称。通过**查询集群详情**获取instances中的name属性。
level	是	String	日志级别。可查询的日志级别为：INFO，ERROR，DEBUG，WARN。

参数	是否必选	参数类型	描述
logType	是	String	日志类型。可查询的日志类型为：deprecation，indexingSlow，searchSlow，instance。

响应参数

状态码： 200

表 4-228 响应 Body 参数

参数	参数类型	描述
logList	Array of **logList** objects	日志列表。

表 4-229 logList

参数	参数类型	描述
content	String	日志内容。
date	String	日期。
level	String	日志级别。

请求示例

查询日志信息。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/logs/
search

{
  "instanceName" : "css-4312-ess-esn-1-1",
  "level" : "INFO",
  "logType" : "instance"
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "logList" : [ {
    "content" : "xxxxxx",
    "date" : "2021-10-08T03:55:54,718",
    "level" : "INFO"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询日志信息。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowLogBackupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowLogBackupRequest request = new ShowLogBackupRequest();
        request.withClusterId("{cluster_id}");
        GetLogBackupReq body = new GetLogBackupReq();
        body.withLogType("instance");
        body.withLevel("INFO");
        body.withInstanceName("css-4312-ess-esn-1-1");
        request.withBody(body);
        try {
            ShowLogBackupResponse response = client.showLogBackup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询日志信息。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowLogBackupRequest()
        request.cluster_id = "{cluster_id}"
        request.body = GetLogBackupReq(
            log_type="instance",
            level="INFO",
            instance_name="css-4312-ess-esn-1-1"
        )
        response = client.show_log_backup(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询日志信息。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowLogBackupRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.GetLogBackupReq{
        LogType: "instance",
        Level: "INFO",
        InstanceName: "css-4312-ess-esn-1-1",
    }
    response, err := client.ShowLogBackup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.5.10 连通性测试

功能介绍

该接口用于连通性测试。

调用方法

请参见如何调用API。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/logs/connectivity

表 4-230 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定待测试的集群ID。

请求参数

表 4-231 请求 Body 参数

参数	是否必选	参数类型	描述
target_cluster_id	是	String	目标集群ID

响应参数

无

请求示例

目标集群连通性测试。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/logs/
connectivity

{
  "target_cluster_id" : "4f3deec3-efa8-4598-bf91-560aad1377a4"
}
```

响应示例

状态码： 200

请求已成功。

```
{ }
```

SDK 代码示例

SDK代码示例如下。

Java

目标集群连通性测试。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartTargetClusterConnectivityTestSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartTargetClusterConnectivityTestRequest request = new StartTargetClusterConnectivityTestRequest();
        request.withClusterId("{cluster_id}");
        StartTargetClusterConnectivityTestReq body = new StartTargetClusterConnectivityTestReq();
        body.withTargetClusterId("4f3deec3-efa8-4598-bf91-560aad1377a4");
        request.withBody(body);
        try {
            StartTargetClusterConnectivityTestResponse response =
client.startTargetClusterConnectivityTest(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

目标集群连通性测试。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartTargetClusterConnectivityTestRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartTargetClusterConnectivityTestReq(
            target_cluster_id="4f3deec3-efa8-4598-bf91-560aad1377a4"
        )
        response = client.start_target_cluster_connectivity_test(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

目标集群连通性测试。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartTargetClusterConnectivityTestRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.StartTargetClusterConnectivityTestReq{
        TargetClusterId: "4f3deec3-efa8-4598-bf91-560aad1377a4",
    }
    response, err := client.StartTargetClusterConnectivityTest(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.6 公网访问接口

4.6.1 开启公网访问

功能介绍

该接口用于开启公网访问。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/public/open

表 **4-232** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定开启公网访问的集群ID。

请求参数

表 **4-233** 请求 Body 参数

参数	是否必选	参数类型	描述
eip	是	**BindPublicReqEip** object	弹性IP信息。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1：是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0：否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 **4-234** BindPublicReqEip

参数	是否必选	参数类型	描述
bandWidth	是	**BindPublicReqEipBandWidth** object	公网带宽信息。

表 **4-235** BindPublicReqEipBandWidth

参数	是否必选	参数类型	描述
size	是	Integer	带宽大小。单位：Mbit/s

响应参数

状态码： 200

表 **4-236** 响应 Body 参数

参数	参数类型	描述
action	String	操作行为。固定为bindZone，表示绑定成功。

请求示例

开启公网访问。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/public/
open

{
  "eip" : {
    "bandWidth" : {
      "size" : 5
    }
  },
  "isAutoPay" : 1
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "action" : "bindZone"
}
```

SDK 代码示例

SDK代码示例如下。

Java

开启公网访问。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateBindPublicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateBindPublicRequest request = new CreateBindPublicRequest();
        request.withClusterId("{cluster_id}");
        BindPublicReq body = new BindPublicReq();
        BindPublicReqEipBandWidth bandWidthEip = new BindPublicReqEipBandWidth();
        bandWidthEip.withSize(5);
        BindPublicReqEip eipbody = new BindPublicReqEip();
        eipbody.withBandWidth(bandWidthEip);
        body.withIsAutoPay(1);
        body.withEip(eipbody);
        request.withBody(body);
        try {
            CreateBindPublicResponse response = client.createBindPublic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启公网访问。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateBindPublicRequest()
        request.cluster_id = "{cluster_id}"
        bandWidthEip = BindPublicReqEipBandWidth(
            size=5
        )
        eipbody = BindPublicReqEip(
            band_width=bandWidthEip
        )
        request.body = BindPublicReq(
            is_auto_pay=1,
            eip=eipbody
        )
        response = client.create_bind_public(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启公网访问。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateBindPublicRequest{}
    request.ClusterId = "{cluster_id}"
    bandWidthEip := &model.BindPublicReqEipBandWidth{
        Size: int32(5),
    }
    eipbody := &model.BindPublicReqEip{
        BandWidth: bandWidthEip,
    }
    isAutoPayBindPublicReq:= int32(1)
    request.Body = &model.BindPublicReq{
        IsAutoPay: &isAutoPayBindPublicReq,
        Eip: eipbody,
    }
    response, err := client.CreateBindPublic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.6.2 关闭公网访问

功能介绍

该接口用于关闭公网访问。包周期类型的集群不支持通过api进行关闭公网访问。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/public/close

表 4-237 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定关闭公网访问集群ID。

请求参数

表 4-238 请求 Body 参数

参数	是否必选	参数类型	描述
eip	否	**UnBindPublicReqEipReq** object	弹性IP信息。

表 4-239 UnBindPublicReqEipReq

参数	是否必选	参数类型	描述
bandWidth	否	**BindPublicReqEipBandWidth** object	公网带宽信息。

表 4-240 BindPublicReqEipBandWidth

参数	是否必选	参数类型	描述
size	是	Integer	带宽大小。单位：Mbit/s

响应参数

状态码： 200

表 **4-241** 响应 Body 参数

参数	参数类型	描述
action	String	操作行为。固定为：unbindZone，表示解绑成功。

请求示例

关闭公网访问。

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/public/
close

{
  "eip" : {
    "bandWidth" : {
      "size" : 5
    }
  }
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "action" : "unbindZone"
}
```

SDK 代码示例

SDK代码示例如下。

Java

关闭公网访问。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateUnbindPublicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateUnbindPublicRequest request = new UpdateUnbindPublicRequest();
        request.withClusterId("{cluster_id}");
        UnBindPublicReq body = new UnBindPublicReq();
        BindPublicReqEipBandWidth bandWidthEip = new BindPublicReqEipBandWidth();
        bandWidthEip.withSize(5);
        UnBindPublicReqEipReq eipbody = new UnBindPublicReqEipReq();
        eipbody.withBandWidth(bandWidthEip);
        body.withEip(eipbody);
        request.withBody(body);
        try {
            UpdateUnbindPublicResponse response = client.updateUnbindPublic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

关闭公网访问。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateUnbindPublicRequest()
        request.cluster_id = "{cluster_id}"
        bandWidthEip = BindPublicReqEipBandWidth(
            size=5
        )
        eipbody = UnBindPublicReqEipReq(
            band_width=bandWidthEip
        )
        request.body = UnBindPublicReq(
            eip=eipbody
        )
        response = client.update_unbind_public(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

关闭公网访问。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateUnbindPublicRequest{}
    request.ClusterId = "{cluster_id}"
    bandWidthEip := &model.BindPublicReqEipBandWidth{
        Size: int32(5),
    }
    eipbody := &model.UnBindPublicReqEipReq{
        BandWidth: bandWidthEip,
    }
    request.Body = &model.UnBindPublicReq{
        Eip: eipbody,
    }
    response, err := client.UpdateUnbindPublic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.6.3 修改公网访问带宽

功能介绍

该接口用于修改公网访问带宽。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/public/bandwidth

表 4-242 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定修改公网访问带宽集群ID。

请求参数

表 4-243 请求 Body 参数

参数	是否必选	参数类型	描述
bandWidth	是	**BindPublicReqEipBandWidth** object	公网带宽信息。
isAutoPay	否	Integer	是否自动支付。下单订购后，是否自动从客户的华为云账户中支付，而不需要客户手动去进行支付。该参数适用于包周期集群。 ● 1: 是（会自动选择折扣和优惠券进行优惠，然后自动从客户华为云账户中支付），自动支付失败后会生成订单成功(该订单应付金额是优惠后金额)、但订单状态为“待支付”，等待客户手动支付(手动支付时，客户还可以修改系统自动选择的折扣和优惠券)。 ● 0: 否（需要客户手动去支付，客户可以选择折扣和优惠券）。默认值为“0”。

表 4-244 BindPublicReqEipBandWidth

参数	是否必选	参数类型	描述
size	是	Integer	带宽大小。单位：Mbit/s

响应参数

无

请求示例

修改公网访问带宽。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/public/
bandwidth

{
  "bandWidth" : {
    "size" : 5
  },
  "isAutoPay" : 1
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改公网访问带宽。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdatePublicBandWidthSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdatePublicBandWidthRequest request = new UpdatePublicBandWidthRequest();
        request.withClusterId("{cluster_id}");
        BindPublicReqEipReq body = new BindPublicReqEipReq();
        BindPublicReqEipBandWidth bandWidthbody = new BindPublicReqEipBandWidth();
        bandWidthbody.withSize(5);
        body.withIsAutoPay(1);
        body.withBandWidth(bandWidthbody);
        request.withBody(body);
        try {
            UpdatePublicBandWidthResponse response = client.updatePublicBandWidth(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改公网访问带宽。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdatePublicBandWidthRequest()
        request.cluster_id = "{cluster_id}"
        bandWidthbody = BindPublicReqEipBandWidth(
            size=5
        )
        request.body = BindPublicReqEipReq(
            is_auto_pay=1,
            band_width=bandWidthbody
        )
        response = client.update_public_band_width(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改公网访问带宽。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePublicBandWidthRequest{}
    request.ClusterId = "{cluster_id}"
    bandWidthbody := &model.BindPublicReqEipBandWidth{
        Size: int32(5),
    }
    isAutoPayBindPublicReqEipReq:= int32(1)
    request.Body = &model.BindPublicReqEipReq{
        IsAutoPay: &isAutoPayBindPublicReqEipReq,
        BandWidth: bandWidthbody,
    }
    response, err := client.UpdatePublicBandWidth(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.6.4 开启公网访问控制白名单

功能介绍

该接口用于开启公网访问控制白名单。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/public/whitelist/update

表 4-245 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定开启公网访问控制白名单集群ID。

请求参数

表 4-246 请求 Body 参数

参数	是否必选	参数类型	描述
whiteList	是	String	开启白名单的用户IP。

响应参数

无

请求示例

开启公网访问控制白名单。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/public/
whitelist/update

{
  "whiteList" : "192.168.0.xx"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启公网访问控制白名单。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartPublicWhitelistSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartPublicWhitelistRequest request = new StartPublicWhitelistRequest();
        request.withClusterId("{cluster_id}");
        StartPublicWhitelistReq body = new StartPublicWhitelistReq();
        body.withWhiteList("192.168.0.xx");
        request.withBody(body);
        try {
            StartPublicWhitelistResponse response = client.startPublicWhitelist(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启公网访问控制白名单。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartPublicWhitelistRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartPublicWhitelistReq(
            white_list="192.168.0.xx"
        )
        response = client.start_public_whitelist(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启公网访问控制白名单。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartPublicWhitelistRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.StartPublicWhitelistReq{
        WhiteList: "192.168.0.xx",
    }
    response, err := client.StartPublicWhitelist(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.6.5 关闭公网访问控制白名单

功能介绍

该接口用于关闭公网访问控制白名单。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/public/whitelist/close

表 4-247 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定关闭公网访问控制白名单集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopPublicWhitelistSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopPublicWhitelistRequest request = new StopPublicWhitelistRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopPublicWhitelistResponse response = client.stopPublicWhitelist(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopPublicWhitelistRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_public_whitelist(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopPublicWhitelistRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopPublicWhitelist(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.7 快照管理接口

4.7.1 自动设置集群快照的基础配置（不推荐使用）

功能介绍

该接口用于自动设置集群快照的基础配置，包括配置OBS桶和IAM委托。

- “OBS桶”：快照存储的OBS桶位置。
- “备份路径”：快照在OBS桶中的存放路径。
- “IAM委托”：由于需要将快照保存在OBS中，所以需要在IAM中设置对应的委托获取对OBS服务的授权。

说明

自动设置集群快照接口将会自动创建快照OBS桶和委托。如果有多个集群，每个集群使用这个接口都会创建一个不一样的OBS桶，可能会导致OBS的配额不够，较多的OBS桶也难以维护。建议可以直接使用**修改集群快照的基础配置**。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/auto_setting

表 4-248 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定要备份的集群ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartAutoSettingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartAutoSettingRequest request = new StartAutoSettingRequest();
        request.withClusterId("{cluster_id}");
        try {
            StartAutoSettingResponse response = client.startAutoSetting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartAutoSettingRequest()
        request.cluster_id = "{cluster_id}"
        response = client.start_auto_setting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartAutoSettingRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StartAutoSetting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。

错误码

请参见**错误码**。

4.7.2 修改集群快照的基础配置

功能介绍

该接口用于修改集群快照的基础配置，可修改OBS桶和IAM委托。

可以使用该接口开启快照功能。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/setting

表 **4-249** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 **4-250** 请求 Body 参数

参数	是否必选	参数类型	描述
bucket	是	String	备份使用的OBS桶的桶名。
agency	是	String	委托名称，委托给CSS，允许CSS调用您的其他云服务。
basePath	否	String	快照在OBS桶中的存放路径。

响应参数

无

请求示例

开启快照功能。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
index_snapshot/setting

{
  "bucket" : "test-bucket",
  "agency" : "usearch",
  "basePath" : "css_repository/Es-name"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启快照功能。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateSnapshotSettingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateSnapshotSettingRequest request = new UpdateSnapshotSettingRequest();
        request.withClusterId("{cluster_id}");
        UpdateSnapshotSettingReq body = new UpdateSnapshotSettingReq();
        body.withBasePath("css_repository/Es-name");
        body.withAgency("usearch");
        body.withBucket("test-bucket");
        request.withBody(body);
        try {
            UpdateSnapshotSettingResponse response = client.updateSnapshotSetting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启快照功能。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateSnapshotSettingRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdateSnapshotSettingReq(
            base_path="css_repository/Es-name",
            agency="usearch",
            bucket="test-bucket"
        )
        response = client.update_snapshot_setting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启快照功能。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateSnapshotSettingRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.UpdateSnapshotSettingReq{
        BasePath: "css_repository/Es-name",
        Agency: "usearch",
        Bucket: "test-bucket",
    }
    response, err := client.UpdateSnapshotSetting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。
504	网关超时。

错误码

请参见**错误码**。

4.7.3 手动创建快照

功能介绍

该接口用于手动创建一个快照。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot

表 **4-251** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

参数	是否必选	参数类型	描述
cluster_id	是	String	指定要创建快照的集群ID。

请求参数

表 4-252 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	快照名称，快照名称在4位到64位之间，必须以字母开头，可以包含字母、数字、中划线或者下划线，注意字母不能大写且不能包含其他特殊字符。
description	否	String	快照描述，0～256个字符，不能包含!<>=&"'字符。
indices	否	String	指定要备份的索引名称，多个索引用逗号隔开，默认备份所有索引。支持使用“*”匹配多个索引，例如：2018-06*，表示备份名称前缀是2018-06的所有索引的数据。0～1024个字符，不能包含空格和大写字母，且不能包含"\<|>/?特殊字符。

响应参数

状态码： 201

表 4-253 响应 Body 参数

参数	参数类型	描述
backup	**backupRsp** object	快照信息。

表 4-254 backupRsp

参数	参数类型	描述
id	String	快照id。
name	String	快照名称。

请求示例

创建快照。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
index_snapshot

{
  "name" : "snapshot_001",
  "indices" : "myindex1myindex2"
}
```

响应示例

状态码： 201

创建类的请求已成功。

```
{
  "backup" : {
    "id" : "9dc4f5c9-33c0-45c7-9378-ae35ae350682",
    "name" : "snapshot_101"
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建快照。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateSnapshotSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateSnapshotRequest request = new CreateSnapshotRequest();
        request.withClusterId("{cluster_id}");
        CreateSnapshotReq body = new CreateSnapshotReq();
        body.withIndices("myindex1myindex2");
        body.withName("snapshot_001");
        request.withBody(body);
        try {
            CreateSnapshotResponse response = client.createSnapshot(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

创建快照。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateSnapshotRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CreateSnapshotReq(
            indices="myindex1myindex2",
            name="snapshot_001"
        )
        response = client.create_snapshot(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建快照。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateSnapshotRequest{}
    request.ClusterId = "{cluster_id}"
    indicesCreateSnapshotReq:= "myindex1myindex2"
    request.Body = &model.CreateSnapshotReq{
        Indices: &indicesCreateSnapshotReq,
        Name: "snapshot_001",
    }
    response, err := client.CreateSnapshot(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	创建类的请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
406	服务器无法根据客户端请求的内容特性完成请求。
500	表明服务端能被请求访问到，但是不能理解用户的请求。
501	服务器不支持请求的功能，无法完成请求。

错误码

请参见**错误码**。

4.7.4 恢复快照

功能介绍

该接口用于手动恢复一个快照。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/{snapshot_id}/restore

表 4-255 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	恢复快照所属的集群ID。
snapshot_id	是	String	快照ID。

请求参数

表 4-256 请求 Body 参数

参数	是否必选	参数类型	描述
targetCluster	是	String	快照要恢复到的集群的ID。

参数	是否必选	参数类型	描述
indices	否	String	指定要恢复的索引名称，多个索引用逗号隔开，默认恢复所有索引。支持使用“*”匹配多个索引，例如：2018-06*，表示恢复名称前缀是2018-06的所有索引的数据。0～1024个字符，不能包含空格和大写字母，且不能包含"\<|>/?特殊字符。
renamePattern	否	String	匹配要恢复的索引规则，最大支持1024个字符。根据此处定义的过滤条件去恢复符合条件的索引，过滤条件请使用正则表达式。 0～1024个字符，不能包含空格和大写字母，且不能包含"\<|>/?,特殊字符。 renamePattern参数与renameReplacement参数必须同时设置才能生效。
renameReplacement	否	String	索引重命名的规则。0～1024个字符，不能包含空格和大写字母，且不能包含"\<|>/?,特殊字符。例如，“restored_index_$1”表示在所有恢复的索引名称前面加上“restored_”。renamePattern参数与renameReplacement参数必须同时设置才能生效。
replace_exist_indices	否	Boolean	替换已存在的索引

响应参数

无

请求示例

恢复快照。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
index_snapshot/653c086f-ebe7-4383-b902-c031b354f1dd/restore

{
  "targetCluster" : "ea244205-d641-45d9-9dcb-ab2236bcd07e",
  "indices" : "myindex1,myindex2",
  "renamePattern" : "index",
  "renameReplacement" : "restored_index_$1"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

恢复快照。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class RestoreSnapshotSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        RestoreSnapshotRequest request = new RestoreSnapshotRequest();
        request.withClusterId("{cluster_id}");
        request.withSnapshotId("{snapshot_id}");
        RestoreSnapshotReq body = new RestoreSnapshotReq();
        body.withRenameReplacement("restored_index_$1");
        body.withRenamePattern("index");
        body.withIndices("myindex1,myindex2");
        body.withTargetCluster("ea244205-d641-45d9-9dcb-ab2236bcd07e");
        request.withBody(body);
        try {
            RestoreSnapshotResponse response = client.restoreSnapshot(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

恢复快照。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RestoreSnapshotRequest()
        request.cluster_id = "{cluster_id}"
        request.snapshot_id = "{snapshot_id}"
        request.body = RestoreSnapshotReq(
            rename_replacement="restored_index_$1",
            rename_pattern="index",
            indices="myindex1,myindex2",
            target_cluster="ea244205-d641-45d9-9dcb-ab2236bcd07e"
        )
        response = client.restore_snapshot(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

恢复快照。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RestoreSnapshotRequest{}
    request.ClusterId = "{cluster_id}"
    request.SnapshotId = "{snapshot_id}"
    renameReplacementRestoreSnapshotReq:= "restored_index_$1"
    renamePatternRestoreSnapshotReq:= "index"
    indicesRestoreSnapshotReq:= "myindex1,myindex2"
    request.Body = &model.RestoreSnapshotReq{
        RenameReplacement: &renameReplacementRestoreSnapshotReq,
        RenamePattern: &renamePatternRestoreSnapshotReq,
        Indices: &indicesRestoreSnapshotReq,
        TargetCluster: "ea244205-d641-45d9-9dcb-ab2236bcd07e",
    }
    response, err := client.RestoreSnapshot(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	创建类的请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.7.5 删除快照

功能介绍

该接口用于删除快照。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/{snapshot_id}

表 4-257 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	删除快照所属的集群的ID。
snapshot_id	是	String	要删除快照的ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteSnapshotSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteSnapshotRequest request = new DeleteSnapshotRequest();
        request.withClusterId("{cluster_id}");
        request.withSnapshotId("{snapshot_id}");
        try {
            DeleteSnapshotResponse response = client.deleteSnapshot(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteSnapshotRequest()
        request.cluster_id = "{cluster_id}"
        request.snapshot_id = "{snapshot_id}"
        response = client.delete_snapshot(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteSnapshotRequest{}
    request.ClusterId = "{cluster_id}"
    request.SnapshotId = "{snapshot_id}"
    response, err := client.DeleteSnapshot(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.7.6 设置自动创建快照策略

功能介绍

该接口用于设置自动创建快照，默认一天创建一个快照。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/policy

表 4-258 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定要自动创建快照的集群ID。

请求参数

表 4-259 请求 Body 参数

参数	是否必选	参数类型	描述
indices	否	String	需要备份的索引名。*代表所有索引。

参数	是否必选	参数类型	描述
prefix	否	String	自动创建快照的名称前缀，需要用户自己手动输入。 只能包含1~32位小写字母、数字、中划线或者下划线，并且以小写字母开头。 **说明** 当enable为true时该字段为必选字段。
period	否	String	每天创建快照的时刻，只支持整点，后面需加上时区，格式为“HH:mm z”，“HH:mm”表示整点时间，“z”表示时区。比如“00:00 GMT+08:00”、“01:00 GMT+08:00”等。 **说明** 默认值“00:00 GMT+08:00”，当frequency取值HOUR时，无需指定period值，默认从设置成功的下一整点创建。当enable为true时该字段为必选字段。
keepday	否	Integer	自定义设置快照保留的个数，范围是1～90。系统在半点时刻会自动删除超过保留个数的快照。过期删除策略只针对与当前自动创建快照策略相同执行频次的自动快照。 **说明** 当enable为true时该字段为必选字段。
enable	是	String	是否开启自动创建快照策略。 ● true：表示开启自动创建快照策略。 ● false：表示关闭自动创建快照策略。
deleteAuto	否	String	表示关闭自动创建快照策略时，是否需要清除所有自动创建的快照。 默认为“false”，表示不会删除之前已自动创建的快照。设置为true，表示在关闭自动创建快照策略的同时，删除所有已创建的快照。

参数	是否必选	参数类型	描述
frequency	否	String	自动创建快照的执行频次。 **说明** 默认值DAY，取值HOUR时，表示每小时整点执行一次；取值DAY时，表示每天执行一次；取值SUN、MON、TUE、WED、THU、FRI、SAT时，表示在每周对应时间执行一次。比如SUN表示每周日执行一次。保留快照个数与自动创建快照策略中设置的执行频次和索引有关。当执行频次间隔时间短或索引数据量大，保留自动快照可能会达不到设置的个数，请谨慎选择！

响应参数

无

请求示例

设置自动创建快照策略。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
index_snapshot/policy

{
  "prefix" : "snapshot",
  "period" : "16:00 GMT+08:00",
  "keepday" : 7,
  "frequency" : "DAY",
  "enable" : "true",
  "deleteAuto" : "false"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

设置自动创建快照策略。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateAutoCreatePolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateAutoCreatePolicyRequest request = new CreateAutoCreatePolicyRequest();
        request.withClusterId("{cluster_id}");
        SetRDSBackupCnfReq body = new SetRDSBackupCnfReq();
        body.withDeleteAuto("false");
        body.withEnable("true");
        body.withKeepday(7);
        body.withPeriod("16:00 GMT+08:00");
        body.withPrefix("snapshot");
        request.withBody(body);
        try {
            CreateAutoCreatePolicyResponse response = client.createAutoCreatePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

设置自动创建快照策略。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateAutoCreatePolicyRequest()
        request.cluster_id = "{cluster_id}"
        request.body = SetRDSBackupCnfReq(
            delete_auto="false",
            enable="true",
            keepday=7,
            period="16:00 GMT+08:00",
            prefix="snapshot"
        )
        response = client.create_auto_create_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

设置自动创建快照策略。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateAutoCreatePolicyRequest{}
    request.ClusterId = "{cluster_id}"
    deleteAutoSetRdsBackupCnfReq:= "false"
    keepdaySetRdsBackupCnfReq:= int32(7)
    periodSetRdsBackupCnfReq:= "16:00 GMT+08:00"
    prefixSetRdsBackupCnfReq:= "snapshot"
    request.Body = &model.SetRdsBackupCnfReq{
        DeleteAuto: &deleteAutoSetRdsBackupCnfReq,
        Enable: "true",
        Keepday: &keepdaySetRdsBackupCnfReq,
        Period: &periodSetRdsBackupCnfReq,
        Prefix: &prefixSetRdsBackupCnfReq,
    }
    response, err := client.CreateAutoCreatePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。

错误码

请参见**错误码**。

4.7.7 查询自动创建快照的策略

功能介绍

该接口用于查询自动创建快照策略。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/policy

表 4-260 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

参数	是否必选	参数类型	描述
cluster_id	是	String	指定需查询自动创建快照策略的集群ID。

请求参数

无

响应参数

状态码： 200

表 4-261 响应 Body 参数

参数	参数类型	描述
keepday	Integer	自定义设置快照保留的个数。
period	String	每天快照创建时刻。
prefix	String	快照命名前缀，需要用户自己手动输入。
bucket	String	快照存放的OBS桶的桶名。
basePath	String	快照在OBS桶中的存放路径。
agency	String	访问OBS桶用到的委托。
enable	String	是否开启自动创建快照策略。 ● true：表示开启自动创建快照策略。 ● false：表示关闭自动创建快照策略。
frequency	String	自动创建快照的执行频次。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "keepday" : 2,
  "frequency" : "DAY",
  "period" : "16:00 GMT+08:00",
  "prefix" : "snapshot",
  "bucket" : "es-backup",
  "basePath" : "css_repository/tests",
  "agency" : "usearch",
  "enable" : "true"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowAutoCreatePolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowAutoCreatePolicyRequest request = new ShowAutoCreatePolicyRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowAutoCreatePolicyResponse response = client.showAutoCreatePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAutoCreatePolicyRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_auto_create_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAutoCreatePolicyRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowAutoCreatePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。

错误码

请参见**错误码**。

4.7.8 查询快照列表

功能介绍

该接口用于查询集群的所有快照。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/index_snapshots

表 4-262 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定需查询的快照所属的集群的ID。

请求参数

无

响应参数

状态码： 200

表 4-263 响应 Body 参数

参数	参数类型	描述
backups	Array of **ListSnapshotBackupsResp** objects	快照列表。

表 4-264 ListSnapshotBackupsResp

参数	参数类型	描述
created	String	快照创建时间。
datastore	**ListSnapshotBackupsDatastoreResp** object	数据搜索引擎类型。
description	String	快照描述信息。
id	String	快照ID。
clusterId	String	集群ID。
clusterName	String	集群名字。
name	String	快照名称。
status	String	快照状态。
updated	String	快照更新时间，格式为ISO8601：CCYY-MM-DDThh:mm:ss。
backupType	String	快照创建类型: ● 0：表示自动创建。 ● 1：表示手动创建。
backupMethod	String	创建快照方式。
backupExpectedStartTime	String	快照开始执行时间。
backupKeepDay	Integer	快照保留时间。
backupPeriod	String	快照每天执行的时间点。
indices	String	要备份的索引。

参数	参数类型	描述
totalShards	Integer	要备份的索引的总shard数。
failedShards	Integer	备份失败的shard数。
version	String	快照的版本。
restoreStatus	String	快照恢复的状态。
startTime	Long	快照开始执行的时间戳。
endTime	Long	快照执行结束的时间戳。
bucketName	String	保存快照数据的桶名。

表 4-265 ListSnapshotBackupsDatastoreResp

参数	参数类型	描述
type	String	引擎类型，目前只支持elasticsearch。
version	String	Elasticsearch引擎版本号。详细请参考CSS**使用前必读**中支持的版本。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "backups" : [ {
    "created" : "2018-03-07T07:34:47",
    "datastore" : {
      "type" : "elasticsearch",
      "version" : "x.x.x"
    },
    "description" : "",
    "id" : "e29d99c1-3d19-4ea4-ae8d-f252df76cbe9",
    "clusterId" : "37cb1075-c38e-4cd8-81df-442d52df3786",
    "clusterName" : "Es-xfx",
    "name" : "snapshot-002",
    "status" : "COMPLETED",
    "updated" : "2018-03-07T07:40:12",
    "backupType" : "1",
    "backupMethod" : "manual",
    "backupExpectedStartTime" : null,
    "backupKeepDay" : null,
    "backupPeriod" : null,
    "indices" : ".kibanawebsite2",
    "totalShards" : 6,
    "failedShards" : 0,
    "version" : "x.x.x",
    "restoreStatus" : "success",
    "startTime" : 1520408087099,
    "endTime" : 1520408412219,
    "bucketName" : "obs-b8ed"
  }, {
    "created" : "2018-03-06T15:42:37",
    "datastore" : {
      "type" : "elasticsearch",
      "version" : "x.x.x"
    },
    "description" : "",
    "id" : "29a2254e-947f-4463-b65a-5f0b17515fae",
    "clusterId" : "37cb1075-c38e-4cd8-81df-442d52df3786",
    "clusterName" : "Es-xfx",
    "name" : "snapshot-001",
    "status" : "COMPLETED",
    "updated" : "2018-03-06T15:48:04",
    "backupType" : "1",
    "backupMethod" : "manual",
    "backupExpectedStartTime" : null,
    "backupKeepDay" : null,
    "backupPeriod" : null,
    "indices" : ".kibana",
    "totalShards" : 1,
    "failedShards" : 0,
    "version" : "x.x.x",
    "restoreStatus" : "none",
    "startTime" : 1520350957275,
    "endTime" : 1520351284357,
    "bucketName" : "obs-b8ed"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListSnapshotsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);
        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListSnapshotsRequest request = new ListSnapshotsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListSnapshotsResponse response = client.listSnapshots(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSnapshotsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_snapshots(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListSnapshotsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListSnapshots(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。

错误码

请参见错误码。

4.7.9 停用快照功能

功能介绍

该接口用于停用快照功能。

调用方法

请参见如何调用API。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/index_snapshots

表 4-266 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	停用快照所属的集群的ID。

请求参数

无

响应参数

无

请求示例

无

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopSnapshotSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopSnapshotRequest request = new StopSnapshotRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopSnapshotResponse response = client.stopSnapshot(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopSnapshotRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_snapshot(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopSnapshotRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopSnapshot(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
406	服务器无法根据客户端请求的内容特性完成请求。

错误码

请参见**错误码**。

4.7.10 开启自动创建快照功能

功能介绍

该接口用于打开自动备份功能

调用方法

请参见**如何调用API**。

URI

POST /v2.0/{project_id}/clusters/{cluster_id}/snapshots/policy/open

表 4-267 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	快照所属的集群的ID。

请求参数

表 4-268 请求 Body 参数

参数	是否必选	参数类型	描述
indices	否	String	指定要恢复的索引名称，多个索引用逗号隔开，默认恢复所有索引。支持使用“*”匹配多个索引，例如：2018-06*，表示恢复名称前缀是2018-06的所有索引的数据。0～1024个字符，不能包含空格和大写字母，且不能包含"\<|>/?特殊字符。默认值为*，表示恢复所有索引。
keepday	是	Integer	自定义设置快照保留的个数，范围是1～90。系统在半点时刻会自动删除超过保留个数的快照。过期删除策略只针对与当前自动创建快照策略相同执行频次的自动快照。

参数	是否必选	参数类型	描述
frequency	否	String	自动创建快照的执行频次。 **说明** 默认值DAY，取值HOUR时，表示每小时整点执行一次；取值DAY时，表示每天执行一次；取值SUN、MON、TUE、WED、THU、FRI、SAT时，表示在每周对应时间执行一次。比如SUN表示每周日执行一次。保留快照个数与自动创建快照策略中设置的执行频次和索引有关。当执行频次间隔时间短或索引数据量大，保留自动快照可能会达不到设置的个数，请谨慎选择！
period	是	String	每天创建快照的时刻，只支持整点，后面需加上时区，格式为“HH:mm z”，“HH:mm”表示整点时间，“z”表示时区。比如“00:00 GMT+08:00”、“01:00 GMT+08:00”等。 **说明** 默认值“00:00 GMT+08:00”，当frequency取值HOUR时，无需指定period值，默认从设置成功的下一整点创建。
prefix	是	String	自动创建的快照名称前缀，需要用户自己手动输入。只能包含1~32位小写字母、数字、中划线或者下划线，并且以小写字母开头。

响应参数

无

请求示例

打开自动备份功能。

```
POST /v2.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
snapshots/policy/open

{
  "indices" : "*",
  "keepday" : 7,
  "frequency" : "DAY",
  "period" : "16:00 GMT+08:00",
  "prefix" : "snapshot"
}
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.7.11 关闭自动创建快照功能

功能介绍

该接口用于关闭自动备份功能。

调用方法

请参见**如何调用API**。

URI

PUT /v2.0/{project_id}/clusters/{cluster_id}/snapshots/policy/close

表 4-269 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	快照所属的集群的ID。

请求参数

无

响应参数

无

请求示例

关闭自动备份功能。

```
PUT /v2.0/6204a5bd270343b5885144cf9c8c158d/clusters/ea244205-d641-45d9-9dcb-ab2236bcd07e/
snapshots/policy/close
```

响应示例

无

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
403	请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理T解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

错误码

请参见**错误码**。

4.8 终端节点接口

4.8.1 开启终端节点服务

功能介绍

该接口用于开启终端节点服务。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/vpcepservice/open

表 **4-270** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。

参数	是否必选	参数类型	描述
cluster_id	是	String	指定开启终端节点的集群ID。

请求参数

表 4-271 请求 Body 参数

参数	是否必选	参数类型	描述
endpointWithDnsName	否	Boolean	是否开启内网域名。 ● true：开启。 ● false：不开启。
professionVpcep	否	Boolean	创建专业型终端节点。 ● true：开启。 ● false：不开启。
dualstackEnable	否	Boolean	是否开启IPv4/IPv6双栈网络，仅支持在创建专业型终端节点时开启双栈网络，且集群的VPC子网支持IPv6。 ● true：开启。 ● false：不开启。

响应参数

状态码： 200

表 4-272 响应 Body 参数

参数	参数类型	描述
action	String	操作行为。固定为：createVpcepservice，表示已开启终端节点。

请求示例

开启终端节点服务。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
vpcepservice/open

{
  "endpointWithDnsName" : true
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "action" : "createVpcepservice"
}
```

SDK 代码示例

SDK代码示例如下。

Java

开启终端节点服务。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StartVpecpSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartVpecpRequest request = new StartVpecpRequest();
        request.withClusterId("{cluster_id}");
        StartVpecpReq body = new StartVpecpReq();
        body.withEndpointWithDnsName(true);
        request.withBody(body);
        try {
            StartVpecpResponse response = client.startVpecp(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启终端节点服务。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartVpecpRequest()
        request.cluster_id = "{cluster_id}"
        request.body = StartVpecpReq(
            endpoint_with_dns_name=True
        )
        response = client.start_vpecp(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启终端节点服务。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartVpecpRequest{}
    request.ClusterId = "{cluster_id}"
    endpointWithDnsNameStartVpecpReq:= true
    request.Body = &model.StartVpecpReq{
        EndpointWithDnsName: &endpointWithDnsNameStartVpecpReq,
    }
    response, err := client.StartVpecp(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.8.2 关闭终端节点服务

功能介绍

该接口用于关闭终端节点服务。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/vpcepservice/close

表 4-273 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待关闭终端节点的集群ID。

请求参数

无

响应参数

状态码： 200

表 4-274 响应 Body 参数

参数	参数类型	描述
action	String	操作行为。固定为：deleteVpcepservice，表示已关闭终端节点。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "action" : "deleteVpcepservice"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class StopVpecpSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopVpecpRequest request = new StopVpecpRequest();
        request.withClusterId("{cluster_id}");
        try {
            StopVpecpResponse response = client.stopVpecp(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopVpecpRequest()
        request.cluster_id = "{cluster_id}"
        response = client.stop_vpecp(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopVpecpRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.StopVpecp(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

```
  }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.8.3 获取终端节点连接

功能介绍

该接口用于获取终端节点连接。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/vpcepservice/connections

表 4-275 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待查询的集群ID。

表 **4-276** Query 参数

参数	是否必选	参数类型	描述
start	否	Integer	指定查询起始值，默认值为1，即从第1个任务开始查询。
limit	否	Integer	指定查询个数，默认值为10，即一次查询10个任务信息。

请求参数

无

响应参数

状态码： 200

表 **4-277** 响应 Body 参数

参数	参数类型	描述
connections	Array of **connections** objects	连接信息
vpcServiceName	String	终端节点服务名称
permissions	Array of **permissions** objects	权限列表
vpcepUpdateSwitch	Boolean	终端节点更新开关。 ● true： 开启终端节点。 ● false： 关闭终端节点。
total_count	Integer	终端节点数量。

表 **4-278** connections

参数	参数类型	描述
id	String	终端节点ID。
status	String	终端节点状态。 ● accepted：允许该终端节点连接。 ● rejected：拒绝该终端节点连接。
maxSession	String	最大连接数。

参数	参数类型	描述
specificationName	String	终端节点名称。
created_at	String	创建时间，格式为ISO8601：CCYY-MM-DDThh:mm:ss。
update_at	String	更新时间。默认为null。
domain_id	String	拥有者。
vpcepIp	String	终端节点IPv4。
vpcepIpv6Address	String	终端节点IPv6。
vpcepDnsName	String	终端节点内网域名。

表 4-279 permissions

参数	参数类型	描述
id	String	id
permission	String	权限
permission_type	String	权限类型
created_at	String	创建时间

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "connections" : [ {
    "id" : "54b7f739-31a1-40d0-862b-ac85b83ab2da",
    "status" : "accepted",
    "maxSession" : "3000",
    "specificationName" : "default",
    "vpcepIp" : "192.168.0.122",
    "vpcepIpv6Address" : null,
    "vpcepDnsName" : null,
    "created_at" : "2024-06-11T09:36:24Z",
    "update_at" : null,
    "domain_id" : "db9b76a0d9ae431f8e85e89da2ca867c"
  }, {
    "id" : "e88ad0bc-c2c7-419c-bd9b-a961111f0a42",
    "status" : "accepted",
    "maxSession" : "3000",
    "specificationName" : "default",
    "vpcepIp" : "192.168.0.133",
    "vpcepIpv6Address" : null,
    "vpcepDnsName" : null,
    "created_at" : "2023-10-12T07:33:16Z",
    "update_at" : null,
    "domain_id" : "db9b76a0d9ae431f8e85e89da2ca867c"
  } ],
  "vpcepUpdateSwitch" : false,
  "total_count" : 2,
  "vpcServiceName" : "cn-north-4.css-op-no-delete.cf072729-b076-49db-83d3-020dc9f589bd",
  "permissions" : [ {
    "id" : "432a6429-f377-4168-8b24-feb5885af08c",
    "permission" : "iam:domain::db9b76a0d9ae431f8e85e89da2ca867c",
    "permission_type" : "domainId",
    "created_at" : "2023-10-12T07:33:11Z"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowVpcepConnectionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowVpcepConnectionRequest request = new ShowVpcepConnectionRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowVpcepConnectionResponse response = client.showVpcepConnection(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowVpcepConnectionRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_vpcep_connection(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowVpcepConnectionRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowVpcepConnection(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.8.4 更新终端节点连接

功能介绍

该接口用于更新终端节点连接。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/vpcepservice/connections

表 4-280 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定更新终端节点的集群ID。

请求参数

表 4-281 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	期望的操作行为。 ● receive：允许连接。 ● reject：拒绝连接。
endpointIdList	是	Array of strings	终端节点ID列表。

响应参数

无

请求示例

更新终端节点连接。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
vpcepservice/connections

{
  "action" : "receive",
  "endpointIdList" : [ "f132bb14-e1d5-4f25-9f7c-a29e4c8effd4" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

更新终端节点连接。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateVpcepConnectionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateVpcepConnectionRequest request = new UpdateVpcepConnectionRequest();
        request.withClusterId("{cluster_id}");
        UpdateVpcepConnectionReq body = new UpdateVpcepConnectionReq();
        List<String> listbodyEndpointIdList = new ArrayList<>();
        listbodyEndpointIdList.add("f132bb14-e1d5-4f25-9f7c-a29e4c8effd4");
        body.withEndpointIdList(listbodyEndpointIdList);
        body.withAction("receive");
        request.withBody(body);
        try {
            UpdateVpcepConnectionResponse response = client.updateVpcepConnection(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新终端节点连接。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateVpcepConnectionRequest()
        request.cluster_id = "{cluster_id}"
        listEndpointIdListbody = [
            "f132bb14-e1d5-4f25-9f7c-a29e4c8effd4"
        ]
        request.body = UpdateVpcepConnectionReq(
            endpoint_id_list=listEndpointIdListbody,
            action="receive"
        )
        response = client.update_vpcep_connection(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新终端节点连接。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateVpcepConnectionRequest{}
    request.ClusterId = "{cluster_id}"
    var listEndpointIdListbody = []string{
        "f132bb14-e1d5-4f25-9f7c-a29e4c8effd4",
    }
    request.Body = &model.UpdateVpcepConnectionReq{
        EndpointIdList: listEndpointIdListbody,
        Action: "receive",
    }
    response, err := client.UpdateVpcepConnection(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.8.5 修改终端节点服务白名单

功能介绍

该接口用于修改终端节点服务白名单。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/vpcepservice/permissions

表 4-282 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改的集群ID。

请求参数

表 4-283 请求 Body 参数

参数	是否必选	参数类型	描述
vpcPermissions	是	Array of strings	白名单(用户的账号ID)。

响应参数

无

请求示例

修改终端节点服务白名单。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/
vpcepservice/permissions

{
  "vpcPermissions" : [ "dffef70c1db243cbb373e45791165281" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改终端节点服务白名单。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateVpcepWhitelistSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateVpcepWhitelistRequest request = new UpdateVpcepWhitelistRequest();
        request.withClusterId("{cluster_id}");
        UpdateVpcepWhitelistReq body = new UpdateVpcepWhitelistReq();
        List<String> listbodyVpcPermissions = new ArrayList<>();
        listbodyVpcPermissions.add("dffef70c1db243cbb373e45791165281");
        body.withVpcPermissions(listbodyVpcPermissions);
        request.withBody(body);
        try {
            UpdateVpcepWhitelistResponse response = client.updateVpcepWhitelist(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改终端节点服务白名单。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateVpcepWhitelistRequest()
        request.cluster_id = "{cluster_id}"
        listVpcPermissionsbody = [
            "dffef70c1db243cbb373e45791165281"
        ]
        request.body = UpdateVpcepWhitelistReq(
            vpc_permissions=listVpcPermissionsbody
        )
        response = client.update_vpcep_whitelist(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改终端节点服务白名单。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateVpcepWhitelistRequest{}
    request.ClusterId = "{cluster_id}"
    var listVpcPermissionsbody = []string{
        "dffef70c1db243cbb373e45791165281",
    }
    request.Body = &model.UpdateVpcepWhitelistReq{
        VpcPermissions: listVpcPermissionsbody,
    }
    response, err := client.UpdateVpcepWhitelist(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.9 参数配置接口

4.9.1 修改参数配置

功能介绍

该接口用于修改参数配置。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/ymls/update

表 4-284 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定修改参数配置的集群ID。

请求参数

表 4-285 请求 Body 参数

参数	是否必选	参数类型	描述
edit	是	**UpdateYmlsReqEdit** object	配置文件信息。

表 4-286 UpdateYmlsReqEdit

参数	是否必选	参数类型	描述
modify	是	**UpdateYmlsReqEditModify** object	配置文件操作。 ● modify: 修改参数配置。 ● delete: 删除参数配置。 ● reset: 重置参数配置。

表 **4-287** UpdateYmlsReqEditModify

参数	是否必选	参数类型	描述
elasticsearch.yml	是	Object	参数配置列表。值为需要修改的json数据。

响应参数

状态码： 200

表 **4-288** 响应 Body 参数

参数	参数类型	描述
acknowledged	Boolean	修改是否成功。 ● true: 修改成功。 ● false: 修改失败。
externalMessage	String	错误信息描述。当acknowledged为true时，该字段返回null。
httpErrorResponse	String	HTTP错误信息。默认为null。

请求示例

修改参数配置。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ymls/
update

{
  "edit" : {
    "modify" : {
      "elasticsearch.yml" : {
        "thread_pool.force_merge.size" : 1
      }
    }
  }
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "acknowledged" : true,
  "externalMessage" : null,
  "httpErrorResponse" : null
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改参数配置。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateYmlsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateYmlsRequest request = new UpdateYmlsRequest();
        request.withClusterId("{cluster_id}");
        UpdateYmlsReq body = new UpdateYmlsReq();
        UpdateYmlsReqEditModify modifyEdit = new UpdateYmlsReqEditModify();
        modifyEdit.withElasticsearchYml("{\"thread_pool.force_merge.size\":1}");
        UpdateYmlsReqEdit editbody = new UpdateYmlsReqEdit();
        editbody.withModify(modifyEdit);
        body.withEdit(editbody);
        request.withBody(body);
        try {
            UpdateYmlsResponse response = client.updateYmls(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改参数配置。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateYmlsRequest()
        request.cluster_id = "{cluster_id}"
        modifyEdit = UpdateYmlsReqEditModify(
            elasticsearch_yml="{\"thread_pool.force_merge.size\":1}"
        )
        editbody = UpdateYmlsReqEdit(
            modify=modifyEdit
        )
        request.body = UpdateYmlsReq(
            edit=editbody
        )
        response = client.update_ymls(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改参数配置。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateYmlsRequest{}
    request.ClusterId = "{cluster_id}"
    var elasticsearchYmlModify interface{} = "{\"thread_pool.force_merge.size\":1}"
    modifyEdit := &model.UpdateYmlsReqEditModify{
        ElasticsearchYml: &elasticsearchYmlModify,
    }
    editbody := &model.UpdateYmlsReqEdit{
        Modify: modifyEdit,
    }
    request.Body = &model.UpdateYmlsReq{
        Edit: editbody,
    }
    response, err := client.UpdateYmls(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.9.2 获取参数配置任务列表

功能介绍

该接口可获取参数配置的任务操作列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/ymls/joblists

表 4-289 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

表 4-290 Query 参数

参数	是否必选	参数类型	描述
start	否	Integer	指定查询起始值，默认值为1，即从第1个任务开始查询。
limit	否	Integer	指定查询个数，默认值为10，即一次查询10个任务信息。

请求参数

无

响应参数

状态码： 200

表 4-291 响应 Body 参数

参数	参数类型	描述
configList	Array of **configListRsp** objects	历史修改配置列表。

表 **4-292** configListRsp

参数	参数类型	描述
id	String	操作ID
clusterId	String	集群ID。
createAt	String	创建时间。格式：Unix时间戳格式。
status	String	任务执行状态。 ● true: 执行成功。 ● false: 执行失败。
finishedAt	String	结束时间，当创建未结束时结束时间为null。格式：Unix时间戳格式。
modifyDeleteReset	String	修改参数配置记录。
failedMsg	String	返回错误信息。当状态为success时该参数为null。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "configList" : [ {
    "id" : "7ed7258a-60a8-46fe-8814-52819d491b80",
    "clusterId" : "4213d908-f5dc-4633-8401-cfd7175fca0c",
    "createAt" : 1633658735000,
    "status" : "success",
    "finishedAt" : null,
    "modifyDeleteReset" : "{\"modify\":{\"elasticsearch.yml\":{\"thread_pool.force_merge.size\":\"1\"}}}",
    "failedMsg" : ""
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListYmlsJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListYmlsJobRequest request = new ListYmlsJobRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListYmlsJobResponse response = client.listYmlsJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListYmlsJobRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_ymls_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListYmlsJobRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListYmlsJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.9.3 获取参数配置列表

功能介绍

该接口用于获取当前集群现有的参数配置列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/ymls/template

表 4-293 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定查询集群ID。

请求参数

无

响应参数

状态码： 200

表 4-294 响应 Body 参数

参数	参数类型	描述
configurations	Object	集群参数配置列表。该对象中key值以具体获取为准，value拥有以下属性。 ● id: 参数配置ID。 ● key: 参数名称。 ● vlaue: 参数值。 ● defaultValue: 参数默认值。 ● regex: 参数约束限制。 ● desc: 参数中文描述。 ● type: 参数类型描述。 ● moduleDesc: 参数功能中文描述。 ● modifyEnable: 参数是否可修改 true： 可以修改。 false：不可修改。 ● enableValue: 参数支持修改的值。 ● fileName: 参数存在的文件名称。默认为elasticsearch.yml。 ● version: 版本信息。 ● descENG: 参数英文描述。 ● moduleDescENG: 参数功能英文描述。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "configurations" : {
    "http.cors.allow-credentials" : {
      "id" : "b462d13c-294b-4e0f-91d3-58be2ad02b99",
      "key" : "http.cors.allow-credentials",
      "value" : "false",
      "defaultValue" : "false",
      "regex" : "^(true|false)$",
      "desc" : "跨域访问是否返回头部的Access-Control-Allow-Credentials，布尔类型，包括true和false",
      "type" : "Boolean",
      "moduleDesc" : "跨域访问",
      "modifyEnable" : "true",
      "enableValue" : "true,false",
      "fileName" : "elasticsearch.yml",
      "version" : null,
      "descENG" : "Whether to return the Access-Control-Allow-Credentials of the header during cross-domain access. The value is a Boolean value and the options are true and false.",
      "moduleDescENG" : "Cross-domain Access"
    }
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListYmlsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListYmlsRequest request = new ListYmlsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListYmlsResponse response = client.listYmls(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListYmlsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_ymls(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListYmlsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListYmls(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.10 负载均衡

4.10.1 查询集群支持的 elbv3 负载均衡器

功能介绍

展示查询集群支持的elbv3负载均衡器

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/loadbalancers

表 **4-295** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待查询的集群ID。

请求参数

无

响应参数

状态码： 200

表 **4-296** 响应 Body 参数

参数	参数类型	描述
loadbalancers	Array of **LoadbalancersResource** objects	负载均衡器列表。

表 **4-297** LoadbalancersResource

参数	参数类型	描述
id	String	负载均衡器ID。
name	String	负载均衡器名称。
l7_flavor_id	String	7层协议Id。
ip_target_enable	Boolean	是否开启跨VPC后端。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "loadbalancers" : [ {
    "id" : "5d45faad-6cb3-479b-96b8-3e2de0cc6268",
    "name" : "elb-css",
    "l7_flavor_id" : "9c8c2425-e061-4bf8-ac65-cd1db92b18e1",
    "ip_target_enable" : true
  }, {
    "id" : "5d45faad-6cb3-479b-96b8-3e2de0cc6269",
    "name" : "elb-b832",
    "l7_flavor_id" : "9c8c2425-e061-4bf8-ac65-cd1db92b18e1",
    "ip_target_enable" : true
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListElbsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListElbsRequest request = new ListElbsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListElbsResponse response = client.listElbs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListElbsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_elbs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListElbsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListElbs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.10.2 打开或关闭 ES 负载均衡器

功能介绍

该接口打开或关闭ES负载均衡器。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/loadbalancers/es-switch

表 4-298 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改的集群ID。

请求参数

表 4-299 请求 Body 参数

参数	是否必选	参数类型	描述
enable	是	Boolean	打开或关闭es负载均衡器。 ● true：开启。 ● false：关闭。
agency	否	String	委托名称。
elb_id	否	String	负载均衡器id。

响应参数

状态码： 200

表 4-300 响应 Body 参数

参数	参数类型	描述
elb_id	String	负载均衡器id。

请求示例

开启负载均衡器。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/58ee0f27-70b3-47e0-ac72-9e3df6cd15cd/
loadbalancers/es-switch

{
  "enable" : true,
  "elb_id" : "5d45faad-6cb3-479b-96b8-3e2de0cc6268",
  "agency" : "css_elb_agency"
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "elb_id" : "5d45faad-6cb3-479b-96b8-3e2de0cc6268"
}
```

SDK 代码示例

SDK代码示例如下。

Java

开启负载均衡器。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class EnableOrDisableElbSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        EnableOrDisableElbRequest request = new EnableOrDisableElbRequest();
        request.withClusterId("{cluster_id}");
        UpdateEsElbRequestBody body = new UpdateEsElbRequestBody();
        body.withElbId("5d45faad-6cb3-479b-96b8-3e2de0cc6268");
        body.withAgency("css_elb_agency");
        body.withEnable(true);
        request.withBody(body);
        try {
            EnableOrDisableElbResponse response = client.enableOrDisableElb(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启负载均衡器。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = EnableOrDisableElbRequest()
        request.cluster_id = "{cluster_id}"
        request.body = UpdateEsElbRequestBody(
            elb_id="5d45faad-6cb3-479b-96b8-3e2de0cc6268",
            agency="css_elb_agency",
            enable=True
        )
        response = client.enable_or_disable_elb(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启负载均衡器。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.EnableOrDisableElbRequest{}
    request.ClusterId = "{cluster_id}"
    elbIdUpdateEsElbRequestBody:= "5d45faad-6cb3-479b-96b8-3e2de0cc6268"
    agencyUpdateEsElbRequestBody:= "css_elb_agency"
    request.Body = &model.UpdateEsElbRequestBody{
        ElbId: &elbIdUpdateEsElbRequestBody,
        Agency: &agencyUpdateEsElbRequestBody,
        Enable: true,
    }
    response, err := client.EnableOrDisableElb(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.10.3 ES 监听器配置

功能介绍

该接口用于es监听器配置。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/es-listeners

表 4-301 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待更改集群名称的集群ID。

请求参数

表 4-302 请求 Body 参数

参数	是否必选	参数类型	描述
protocol	是	String	协议类型，支持HTTP、HTTPS
protocol_port	是	Integer	端口。
server_cert_id	否	String	server证书Id。如protocol为HTTPS则该字段必选。
ca_cert_id	否	String	CA证书Id。如protocol为HTTPS且为双向认证时则该字段必选。

响应参数

状态码： 200

表 4-303 响应 Body 参数

参数	参数类型	描述
elb_id	String	负载均衡器id。

请求示例

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/es-listeners

{
  "protocol" : "HTTPS",
  "protocol_port" : "9200",
  "server_cert_id" : "1b74ae4afad741e6bc9497429d3d0838",
  "ca_cert_id" : "2a7b050f598e4ea8a61cffbe2bdb312f"
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "elb_id" : "56a83547-a55a-407a-8245-7f9b3a0606c0"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateElbListenerSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateElbListenerRequest request = new CreateElbListenerRequest();
        request.withClusterId("{cluster_id}");
        CreateEsListenerRequestBody body = new CreateEsListenerRequestBody();
        body.withCaCertId("2a7b050f598e4ea8a61cffbe2bdb312f");
        body.withServerCertId("1b74ae4afad741e6bc9497429d3d0838");
        body.withProtocolPort(9200);
        body.withProtocol("HTTPS");
        request.withBody(body);
        try {
            CreateElbListenerResponse response = client.createElbListener(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateElbListenerRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CreateEsListenerRequestBody(
            ca_cert_id="2a7b050f598e4ea8a61cffbe2bdb312f",
            server_cert_id="1b74ae4afad741e6bc9497429d3d0838",
            protocol_port=9200,
            protocol="HTTPS"
        )
        response = client.create_elb_listener(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateElbListenerRequest{}
    request.ClusterId = "{cluster_id}"
    caCertIdCreateEsListenerRequestBody:= "2a7b050f598e4ea8a61cffbe2bdb312f"
    serverCertIdCreateEsListenerRequestBody:= "1b74ae4afad741e6bc9497429d3d0838"
    request.Body = &model.CreateEsListenerRequestBody{
        CaCertId: &caCertIdCreateEsListenerRequestBody,
        ServerCertId: &serverCertIdCreateEsListenerRequestBody,
        ProtocolPort: int32(9200),
        Protocol: "HTTPS",
    }
    response, err := client.CreateElbListener(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见错误码。

4.10.4 获取该 esELB 的信息，以及页面需要展示健康检查状态

功能介绍

该接口用于获取该ES集群的ELB信息，以及页面需要展示健康检查状态。

调用方法

请参见如何调用API。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/es-listeners

表 4-304 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见获取项目ID和名称。
cluster_id	是	String	指定待查询的集群ID。

请求参数

无

响应参数

状态码： 200

表 4-305 响应 Body 参数

参数	参数类型	描述
serverCertName	String	服务器证书名称。
serverCertId	String	服务器证书ID。

参数	参数类型	描述
cacertName	String	ca证书名称。
cacertId	String	ca证书ID。
elb_enable	Boolean	elb开关信息。
authentication_type	String	认证方式。
loadBalancer	**LoadBalancer** object	负载均衡器对象信息。
listener	**Elbv3Listener** object	监听器对象信息
healthmonitors	Array of **Member** objects	健康检查结果集合。

表 **4-306** LoadBalancer

参数	参数类型	描述
id	String	负载均衡器ID。
name	String	负载均衡器名称。
guaranteed	String	是否独享型LB，false：共享型，true：独享型。
billing_info	String	资源账单信息。空：按需计费。非空：包周期计费。
description	String	描述信息。
vpc_id	String	负载均衡器所属VPC ID。
provisioning_status	String	负载均衡器的配置状态。
listeners	Array of **IdListWrapper** objects	关联的listener列表。
vip_address	String	负载均衡器的IPv4虚拟IP地址。
vip_port_id	String	负载均衡器的IPv4对应的port ID。
ipv6_vip_address	String	负载均衡器的IPv6地址。
publicips	Array of **PublicIpInfo** objects	负载均衡器绑定的公网IP。

表 **4-307** IdListWrapper

参数	参数类型	描述
id	String	listener id。

表 **4-308** PublicIpInfo

参数	参数类型	描述
publicip_id	String	弹性公网ip配置id。
publicip_address	String	弹性公网IP地址。
ip_version	Integer	IP版本信息。取值范围：4和6 4：IPv4。6：IPv6。

表 **4-309** Elbv3Listener

参数	参数类型	描述
id	String	监听器ID。
name	String	监听器的名称。
protocol	String	监听器的监听协议。
protocol_port	Integer	监听器的前端监听端口。
ipgroup	**ListenerIpGroup** object	listener对象中的ipgroup信息。

表 **4-310** ListenerIpGroup

参数	参数类型	描述
ipgroup_id	String	监听器关联的访问控制组的id。 创建时必选，更新时非必选。
enable_ipgroup	Boolean	访问控制组的状态。 True:开启访问控制；False：关闭访问控制。

表 **4-311** Member

参数	参数类型	描述
name	String	后端服务器名称。

参数	参数类型	描述
address	String	后端服务器对应的IP地址。
protocol_port	Integer	后端服务器业务端口号。
operating_status	String	后端云服务器的健康状态。ONLINE：后端云服务器正常。NO_MONITOR：后端云服务器所在的服务器组没有健康检查器。OFFLINE：后端云服务器关联的ECS服务器不存在或已关机。
instance_id	String	member关联的实例ID。空表示member关联的实例为非真实设备。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "loadBalancer" : {
    "id" : "5d45faad-6cb3-479b-96b8-3e2de0cc6268",
    "name" : "elb-css",
    "guaranteed" : "true",
    "description" : "",
    "listeners" : [ {
      "id" : "011d14fa-908b-4cd9-b0d6-0768ddc6cb71"
    }, {
      "id" : "04b86029-c281-4490-a3bd-5ea1266658ba"
    } ],
    "publicips" : [ {
      "publicip_id" : "f678e23c-96a5-49e8-8ea2-bd8b47d41e78",
      "publicip_address" : "100.1.1.1",
      "ip_version" : 4
    } ],
    "billing_info" : null,
    "vpc_id" : "4f3deec3-efa8-4598-bf91-560aad1377a3",
    "provisioning_status" : "ACTIVE",
    "vip_address" : "10.0.0.1",
    "ipv6_vip_address" : "2409:27ff:2003:3e:1fd::f3"
  },
  "listener" : {
    "id" : "41ff041d-c7b9-4142-9167-fa93d54f97da",
    "name" : "css-searchServer",
    "protocol" : "HTTPS",
    "ipgroup" : null,
    "protocol_port" : 9265
  },
  "healthmonitors" : [ {
    "instance_id" : "bac86342-2222-43e6-817f-57f040a174a6",
    "name" : "",
    "address" : "10.0.0.87",
    "protocol_port" : 9200,
    "operating_status" : "ONLINE"
  }, {
    "instance_id" : "d935b82c-f94b-4ae0-9997-ddc90885d8c6",
    "name" : "",
    "address" : "10.0.0.61",
    "protocol_port" : 9200,
    "operating_status" : "ONLINE"
  } ],
  "serverCertName" : "server1",
  "serverCertId" : "82375af01c0d40f6a44c15962c570625",
  "cacertName" : null,
  "cacertId" : null,
  "elb_enable" : true,
  "authentication_type" : "single"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ShowElbDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowElbDetailRequest request = new ShowElbDetailRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowElbDetailResponse response = client.showElbDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowElbDetailRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_elb_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowElbDetailRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowElbDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.10.5 更新 ES 监听器

功能介绍

该接口用于更新ES监听器。

调用方法

请参见**如何调用API**。

URI

PUT /v1.0/{project_id}/clusters/{cluster_id}/es-listeners/{listener_id}

表 **4-312** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。
listener_id	是	String	指定待操作的监听器ID。

请求参数

表 **4-313** 请求 Body 参数

参数	是否必选	参数类型	描述
listener	是	**EsListenerRequest** object	监听器对象。

表 **4-314** EsListenerRequest

参数	是否必选	参数类型	描述
default_tls_container_ref	是	String	监听器使用的服务器证书ID。
client_ca_tls_container_ref	否	String	监听器使用的CA证书ID。如果更新双向认证，则该参数为必选。

响应参数

状态码： 200

表 **4-315** 响应 Body 参数

参数	参数类型	描述
listener	**EsListenerResponse** object	监听器对象。

表 **4-316** EsListenerResponse

参数	参数类型	描述
protocol	String	监听器的监听协议。

参数	参数类型	描述
id	String	监听器ID。
name	String	监听器的名称。
protocol_port	String	监听器的前端监听端口。
ipgroup	**EsIpgroupResource** object	listener对象中的访问控制信息。

表 4-317 EsIpgroupResource

参数	参数类型	描述
ipgroup_id	String	监听器关联的访问控制组的id。
enable_ipgroup	Boolean	访问控制组的状态。
type	String	访问控制组的类型。

请求示例

```
PUT /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/58ee0f27-70b3-47e0-ac72-9e3df6cd15cd/es-listeners/4425eb63-78ce-4c63-b60e-492befdac0e7

{
  "listener" : {
    "default_tls_container_ref" : "9c82320a8dd049eba49b28e70a2c0c3e"
  }
}
```

响应示例

状态码： 200

请求已成功。

```
{
  "listener" : {
    "id" : "9c82320a8dd049eba49b28e70a2c0c3e",
    "name" : "name",
    "protocol" : "tcp",
    "protocol_port" : 8080,
    "ipgroup" : {
      "ipgroup_id" : "4f3deec3-efa8-4598-bf91-560aad1377a3",
      "enable_ipgroup" : true,
      "type" : "white"
    }
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class UpdateEsListenerSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateEsListenerRequest request = new UpdateEsListenerRequest();
        request.withClusterId("{cluster_id}");
        request.withListenerId("{listener_id}");
        UpdateEsListenerRequestBody body = new UpdateEsListenerRequestBody();
        EsListenerRequest listenerbody = new EsListenerRequest();
        listenerbody.withDefaultTlsContainerRef("9c82320a8dd049eba49b28e70a2c0c3e");
        body.withListener(listenerbody);
        request.withBody(body);
        try {
            UpdateEsListenerResponse response = client.updateEsListener(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateEsListenerRequest()
        request.cluster_id = "{cluster_id}"
        request.listener_id = "{listener_id}"
        listenerbody = EsListenerRequest(
            default_tls_container_ref="9c82320a8dd049eba49b28e70a2c0c3e"
        )
        request.body = UpdateEsListenerRequestBody(
            listener=listenerbody
        )
        response = client.update_es_listener(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
    request := &model.UpdateEsListenerRequest{}
    request.ClusterId = "{cluster_id}"
    request.ListenerId = "{listener_id}"
    listenerbody := &model.EsListenerRequest{
        DefaultTlsContainerRef: "9c82320a8dd049eba49b28e70a2c0c3e",
    }
    request.Body = &model.UpdateEsListenerRequestBody{
        Listener: listenerbody,
    }
    response, err := client.UpdateEsListener(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.10.6 查询证书列表

功能介绍

该接口用于查询证书列表。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/elb/certificates

表 **4-318** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待查询的集群ID。

请求参数

无

响应参数

状态码： 200

表 **4-319** 响应 Body 参数

参数	参数类型	描述
certificates	**CertificatesResource** object	证书列表信息。

表 **4-320** CertificatesResource

参数	参数类型	描述
id	String	证书ID。
name	String	证书名称。
type	String	SL证书的类型。分为服务器证书(server)、CA证书(client)。

请求示例

无

响应示例

状态码： 200

请求已成功。

```
{
  "certificates" : [ {
    "id" : "8c415d2e2f4c4fdbbfc9c21c98d7832b",
    "name" : "server1",
    "type" : "server"
  }, {
    "id" : "8c415d2e2f4c4fdbbfc9c21c98d7832b",
    "name" : "ca1",
    "type" : "client"
  }, {
    "id" : "8c415d2e2f4c4fdbbfc9c21c98d7832b",
    "name" : "server-css",
    "type" : "server"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListElbCertsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListElbCertsRequest request = new ListElbCertsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListElbCertsResponse response = client.listElbCerts(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListElbCertsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_elb_certs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListElbCertsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListElbCerts(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.11 智能运维

4.11.1 获取智能运维任务列表及详情

功能介绍

该接口用于获取智能运维任务列表及详情。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/clusters/{cluster_id}/ai-ops

表 4-321 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待查询的集群ID。

表 4-322 Query 参数

参数	是否必选	参数类型	描述
limit	否	Integer	分页参数，列表当前分页的数量限制。
start	否	Integer	偏移量。 偏移量为一个大于0小于终端节点服务总个数的整数，表示从偏移量后面的终端节点服务开始查询。

请求参数

无

响应参数

状态码： 200

表 4-323 响应 Body 参数

参数	参数类型	描述
total_size	Integer	检测任务个数。
aiops_list	Array of **aiops_list** objects	检测任务详情列表

表 4-324 aiops_list

参数	参数类型	描述
id	String	检测任务id。
name	String	检测任务名称。
desc	String	检测任务描述。

参数	参数类型	描述
status	Integer	任务执行状态。 ● 150：未开启。 ● 200：已开启。 ● 300：已发送。
summary	**summary** object	风险概要。
create_time	String	检测任务创建时间戳。
smn_status	String	检测任务SMN告警任务发送状态。 ● not_open：未开启。 ● not_trigger：未触发。 ● sent：已发送。 ● send_fail： 发送失败。
smn_fail_reason	String	发送失败原因。
task_risks	Array of **AIOpsRiskInfo** objects	风险项详情。

表 **4-325** summary

参数	参数类型	描述
high	Integer	检测项判定为高风险的数量。
medium	Integer	检测项判定为中风险的数量。
suggestion	Integer	检测项判定为建议的数量。

表 **4-326** AIOpsRiskInfo

参数	参数类型	描述
riskType	String	检测项介绍。
level	String	风险等级。 ● high ● medium ● suggestion
desc	String	风险描述。

参数	参数类型	描述
suggestion	String	风险建议。

请求示例

获取智能运维任务列表及详情。

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ai-ops
```

响应示例

状态码： 200

请求成功。

```
{
  "aiops_list" : [ {
    "id" : "7381a80b-68cb-4b9e-8226-37d686b18b1d",
    "name" : "aiops-test",
    "desc" : "",
    "status" : 200,
    "summary" : {
      "high" : 1,
      "medium" : 0,
      "suggestion" : 1
    },
    "create_time" : 1687944156750,
    "smn_status" : "not_open",
    "smn_fail_reason" : null,
    "task_risks" : [ {
      "riskType" : "检测集群7天内是否有快照备份失败，或7天内无快照备份记录。",
      "level" : "suggestion",
      "desc" : "集群7天内无快照备份记录。",
      "suggestion" : "建议您开启集群快照功能并保持7天内快照备份成功。如快照备份失败，建议您进入集群快照
页面详细查看失败原因、进入日志管理页面查看日志详情。"
    }, {
      "riskType" : "检测集群节点数与AZ(可用区)数，判定Elasticsearch分布式集群的高可用性。",
      "level" : "high",
      "desc" : "当前集群为单个或两个节点，当出现一个节点故障时可能导致整个集群不可用，服务可用性风险较
高。",
      "suggestion" : "建议您可通过CSS集群控制台-Elasticsearch-更多-形态变更-切换可用区更改为多AZ集群，通
过更改集群规格调整集群节点数量。"
    } ]
  } ],
  "total_size" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListAiOpsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAiOpsRequest request = new ListAiOpsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListAiOpsResponse response = client.listAiOps(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAiOpsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_ai_ops(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAiOpsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListAiOps(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.11.2 创建一次集群检测任务

功能介绍

该接口用于创建一个集群检测任务。

调用方法

请参见**如何调用API**。

URI

POST /v1.0/{project_id}/clusters/{cluster_id}/ai-ops

表 4-327 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待操作的集群ID。

请求参数

表 **4-328** 请求 Header 参数

参数	是否必选	参数类型	描述
X-Language	否	String	语言

表 **4-329** 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	检测任务名称。
description	否	String	检测任务描述。
alarm	否	**alarm** object	检测任务完成后发送SMN告警消息。

表 **4-330** alarm

参数	是否必选	参数类型	描述
level	是	String	SMN告警消息敏感度。 ● high：高风险。 ● medium：中风险。 ● suggestion：建议。 ● norisk：无风险。
smn_topic	是	String	SMN主题名称。

响应参数

无

请求示例

创建一个集群检测任务。

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ai-ops

{
   "name": " aiops-test ",
"description": "创建集群检测任务。
 ",
   "alarm":{
       "level":"high",
       "smn_topic":"aiops-test"
   }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

创建一个集群检测任务。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class CreateAiOpsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateAiOpsRequest request = new CreateAiOpsRequest();
        request.withClusterId("{cluster_id}");
        CreateAiOpsRequestBody body = new CreateAiOpsRequestBody();
        CreateAiOpsRequestBodyAlarm alarmbody = new CreateAiOpsRequestBodyAlarm();
        alarmbody.withLevel("high")
            .withSmnTopic("aiops-test");
        body.withAlarm(alarmbody);
        body.withDescription("创建集群检测任务。
 ");
        body.withName(" aiops-test ");
        request.withBody(body);
        try {
            CreateAiOpsResponse response = client.createAiOps(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

创建一个集群检测任务。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateAiOpsRequest()
        request.cluster_id = "{cluster_id}"
        alarmbody = CreateAiOpsRequestBodyAlarm(
            level="high",
            smn_topic="aiops-test"
        )
        request.body = CreateAiOpsRequestBody(
            alarm=alarmbody,
            description="创建集群检测任务。
 ",
            name=" aiops-test "
        )
        response = client.create_ai_ops(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建一个集群检测任务。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateAiOpsRequest{}
    request.ClusterId = "{cluster_id}"
    alarmbody := &model.CreateAiOpsRequestBodyAlarm{
        Level: "high",
        SmnTopic: "aiops-test",
    }
    descriptionCreateAiOpsRequestBody:= "创建集群检测任务。
 "
    request.Body = &model.CreateAiOpsRequestBody{
        Alarm: alarmbody,
        Description: &descriptionCreateAiOpsRequestBody,
        Name: " aiops-test ",
    }
    response, err := client.CreateAiOps(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。

状态码	描述
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.11.3 删除一个检测任务记录

功能介绍

该接口用于删除一个检测任务记录。

调用方法

请参见**如何调用API**。

URI

DELETE /v1.0/{project_id}/clusters/{cluster_id}/ai-ops/{aiops_id}

表 4-331 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
cluster_id	是	String	指定待删除的集群ID。
aiops_id	是	String	指定检测任务ID。

请求参数

无

响应参数

无

请求示例

删除一个检测任务记录。

```
DELETE /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ai-ops/e19c9298-465e-42ad-a0ae-b6b552222925
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class DeleteAiOpsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteAiOpsRequest request = new DeleteAiOpsRequest();
        request.withClusterId("{cluster_id}");
        request.withAiopsId("{aiops_id}");
        try {
            DeleteAiOpsResponse response = client.deleteAiOps(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8
```

```python
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteAiOpsRequest()
        request.cluster_id = "{cluster_id}"
        request.aiops_id = "{aiops_id}"
        response = client.delete_ai_ops(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```go
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
    request := &model.DeleteAiOpsRequest{}
    request.ClusterId = "{cluster_id}"
    request.AiopsId = "{aiops_id}"
    response, err := client.DeleteAiOps(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求已成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

请参见**错误码**。

4.11.4 获取智能运维告警可用的 SMN 主题

功能介绍

该接口用于获取智能运维告警可用的SMN主题。

调用方法

请参见**如何调用API**。

URI

GET /v1.0/{project_id}/domains/{domain_id}/ai-ops/smn-topics

表 **4-332** 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法请参见**获取项目ID和名称**。
domain_id	是	String	域账号ID。

请求参数

无

响应参数

状态码： 200

表 **4-333** 响应 Body 参数

参数	参数类型	描述
topicsName	Array of strings	主题名称

请求示例

获取智能运维任务列表及详情。

```
GET /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters/4f3deec3-efa8-4598-bf91-560aad1377a3/ai-ops
```

响应示例

状态码： 200

请求成功。

```
{
  "topicsName" : [ "aiops-test" ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.css.v1.region.CssRegion;
import com.huaweicloud.sdk.css.v1.*;
import com.huaweicloud.sdk.css.v1.model.*;


public class ListSmnTopicsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CssClient client = CssClient.newBuilder()
                .withCredential(auth)
                .withRegion(CssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListSmnTopicsRequest request = new ListSmnTopicsRequest();
        request.withDomainId("{domain_id}");
        try {
            ListSmnTopicsResponse response = client.listSmnTopics(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcss.v1.region.css_region import CssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcss.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSmnTopicsRequest()
        request.domain_id = "{domain_id}"
        response = client.list_smn_topics(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    css "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/css/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := css.NewCssClient(
        css.CssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListSmnTopicsRequest{}
    request.DomainId = "{domain_id}"
    response, err := client.ListSmnTopics(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功。
400	非法请求。 建议直接修改该请求，不要重试该请求。

状态码	描述
409	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
412	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

错误码

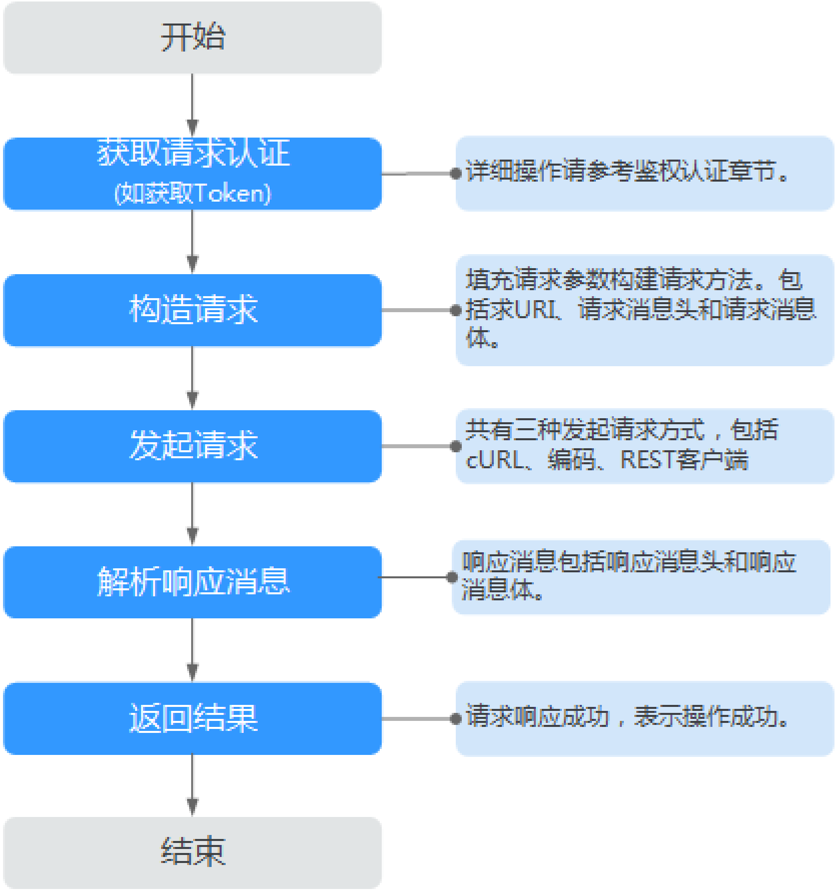
请参见**错误码**。

5 应用示例

5.1 创建一个按需收费的 Elasticsearch 集群

本节通过调用云搜索服务的API创建CSS服务集群，CSS的接口调用流程如图5-1所示。

图 5-1 接口调用流程


开始
获取请求认证
(如获取Token)
详细操作请参考鉴权认证章节。
构造请求
填充请求参数构建请求方法。包括求URL、请求消息头和请求消息体。
发起请求
共有三种发起请求方式，包括cURL、编码、REST客户端
解析响应消息
响应消息包括响应消息头和响应消息体。
返回结果
请求响应成功，表示操作成功。
结束

说明

通过IAM服务获取到的Token有效期为24小时，需要使用同一个Token鉴权时，可以先将Token缓存，避免频繁调用。

涉及 API

当您使用Token认证方式完成认证鉴权时，需要获取用户Token并在调用接口时增加“X-Auth-Token”到业务接口请求消息头中。

- IAM获取Token的API
- CSS创建集群的API

具体步骤

1. 获取Token，发送“POST https://**IAM*的Endpoint***/v3/auth/tokens”。

   Token认证，具体操作请参考**认证鉴权**。

   在响应Header中获取“X-Subject-Token”的值即为用户Token。

2. 设置请求消息头，在Request Header中增加“Content-type ”和“X-Auth-Token”。
   - **Content-Type**：消息体的类型（格式），默认取值为“application/json”。
   - **X-Auth-Token**：填写**1**获取的用户Token。

3. 发送创建集群的请求，在Request Body中传入参数如下：

```
POST /v1.0/6204a5bd270343b5885144cf9c8c158d/clusters

{
  "cluster" : {
    "instance" : {
      "flavorRef" : "ess.spec-4u16g",
      "volume" : {
        "volume_type" : "COMMON",
        "size" : 120
      },
      "nics" : {
        "vpcId" : "{VPC ID}",
        "netId" : "{NET ID}",
        "securityGroupId" : "{Security group ID}"
      },
      "availability_zone" : "{AZ CODE}"
    },
    "datastore" : {
      "version" : "{cluster-version}",
      "type" : "elasticsearch"
    },
    "name" : "cluster-name",
    "instanceNum" : 3,
    "backupStrategy" : {
      "period" : "16:00 GMT+08:00",
      "prefix" : "snapshot",
      "keepday" : 7,
      "frequency" : "DAY",
      "bucket" : "css-obs-backup",
      "basePath" : "css_repository/obs-path",
      "agency" : "css_obs_agency"
    },
    "httpsEnable" : true,
    "authorityEnable" : true,
    "adminPwd" : "{password}",
    "enterprise_project_id" : "0",
    "tags" : [ {
      "key" : "k1",
      "value" : "v1"
    }, {
      "key" : "k2",
      "value" : "v2"
    } ]
  }
}
```

响应消息样例。

```
{
  "cluster": {
    "id": "ef683016-871e-48bc-bf93-74a29d60d214",
    "name": "ES-Test"
  }
}
```

请求响应成功后，返回200 OK，表示集群创建成功。

若请求失败，则会返回错误码及对应的错误信息说明，详细错误码信息请参考状态码。

6 权限和授权项

如果您需要对您所拥有的CSS（Cloud Search Service）进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称IAM），如果华为云账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用CSS服务的其它功能。

默认情况下，新建的IAM用户没有任何权限，您需要将其加入用户组，并给用户组授予策略或角色，才能使用户组中的用户获得相应的权限，这一过程称为授权。授权后，用户就可以基于已有权限对云服务进行操作。

权限根据授权的精细程度，分为角色和策略。角色以服务为粒度，是IAM最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。策略以API接口为粒度进行权限拆分，授权更加精细，可以精确到某个操作、资源和条件，能够满足企业对权限最小化的安全管控要求。

说明

如果您要允许或是禁止某个接口的操作权限，请使用策略。

账号具备所有接口的调用权限，如果使用账号下的IAM用户发起API请求时，该IAM用户必须具备调用该接口所需的权限，否则，API请求将调用失败。每个接口所需要的权限，与各个接口所对应的授权项相对应，只有发起请求的用户被授予授权项所对应的策略，该用户才能成功调用该接口。

支持的授权项

策略包含系统策略和自定义策略，如果系统策略不满足授权要求，CSS服务管理员可以创建自定义策略，并通过给用户组授予自定义策略来进行精细的访问控制。策略支持的操作与API相对应，授权项列表说明如下：

- 权限：允许或拒绝对指定资源在特定条件下进行某项操作。
- 对应API接口：自定义策略实际调用的API接口。
- 授权项：自定义策略中支持的Action，在自定义策略中的Action中写入授权项，可以实现授权项对应的权限功能。
- 依赖的授权项：部分Action存在对其他Action的依赖，需要将依赖的Action同时写入授权项，才能实现对应的权限功能。
- IAM项目(Project)/企业项目(Enterprise Project)：自定义策略的授权范围，包括IAM项目与企业项目。授权范围如果同时支持IAM项目和企业项目，表示此授权项对应的自定义策略，可以在IAM和企业管理两个服务中给用户组授权并生效。如

果仅支持IAM项目，不支持企业项目，表示仅能在IAM中给用户组授权并生效，如果在企业管理中授权，则该自定义策略不生效。关于IAM项目与企业项目的区别，详情请参见：**IAM与企业管理的区别**。

说明

“√”表示支持，“x”表示暂不支持。

表 6-1 API 授权项列表

权限	对应API接口	授权项（Action）	IAM项目 (Project)	企业项目 (Enterprise Project)
创建集群	POST /v1.0/{project_id}/clusters	css:cluster:create	√	√
查询集群列表	GET /v1.0/{project_id}/clusters	css:cluster:list	√	√
查询集群详情	GET /v1.0/{project_id}/clusters/{cluster_id}	css:cluster:get	√	√
删除集群	DELETE /v1.0/{project_id}/clusters/{cluster_id}	css:cluster:delete	√	√
重启集群	POST /v1.0/{project_id}/clusters/{cluster_id}/restart	css:cluster:restart	√	√
扩容集群	POST /v1.0/{project_id}/clusters/{cluster_id}/extend	css:cluster:scaleOut	√	√
扩容实例的数量和存储容量	POST /v1.0/{project_id}/clusters/{cluster_id}/role_extend	css:cluster:expand	√	√
查询指定集群的标签	GET /v1.0/{project_id}/css-cluster/{cluster_id}/tags	css:tag:get	√	√
加载自定义词库	POST /v1.0/{project_id}/clusters/{cluster_id}/thesaurus	css:IKThesaurus:load	√	√
查询自定义词库状态	GET /v1.0/{project_id}/clusters/{cluster_id}/thesaurus	css:IKThesaurus:get	√	√
删除自定义词库	DELETE /v1.0/{project_id}/clusters/{cluster_id}/thesaurus	css:IKThesaurus:delete	√	√

权限	对应API接口	授权项（Action）	IAM项目 (Project)	企业项目 (Enterprise Project)
自动设置集群快照的基础配置	POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/auto_setting	css:snapshot:enableAtomaticSnapshot	√	√
修改集群快照的基础配置	POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/setting	css:snapshot:setSnapshotContiguration	√	√
设置自动创建快照策略	POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/policy	css:snapshot:setSnapshotPolicy	√	√
查询集群的自动创建快照策略	GET /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/policy	css:snapshot:getSnapshotPolicy	√	√
手动创建快照	POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot	css:snapshot:create	√	√
查询快照列表	GET /v1.0/{project_id}/clusters/{cluster_id}/index_snapshots	css:snapshot:list	√	√
恢复快照	POST /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/{snapshot_id}/restore	css:snapshot:restore	√	√
删除快照	DELETE /v1.0/{project_id}/clusters/{cluster_id}/index_snapshot/{snapshot_id}	css:snapshot:delete	√	√
停用快照功能	DELETE /v1.0/{project_id}/clusters/{cluster_id}/index_snapshots	css:snapshot:disableSnapshotFuction	√	√

7 附录

7.1 状态码

状态码如表7-1所示。

表 7-1 状态码说明

状态码	编码	状态码说明
100	Continue	继续请求。 这个临时响应用来通知客户端，它的部分请求已经被服务器接收，且仍未被拒绝。
101	Switching Protocols	切换协议。只能切换到更高版本的协议。 例如，切换到HTTPS的高版本协议。
200	OK	请求已成功。
201	Created	创建类的请求已成功。
202	Accepted	已经接受请求，但未处理完成。
203	Non-Authoritative Information	非授权信息，请求成功。
204	NoContent	请求已成功，同时HTTPS响应不包含响应体。 在响应OPTIONS方法的HTTPS请求时返回此状态码。
205	Reset Content	重置内容，服务器处理成功。
206	Partial Content	服务器成功处理了部分GET请求。
300	Multiple Choices	多种选择。请求的资源可包括多个位置，相应可返回一个资源特征与地址的列表用于用户终端（例如：浏览器）选择。

状态码	编码	状态码说明
301	Moved Permanently	永久移动，请求的资源已被永久的移动到新的URI，返回信息会包括新的URI。
302	Found	资源被临时移动。
303	See Other	查看其它地址。 使用GET和POST请求查看。
304	Not Modified	所请求的资源未修改，服务器返回此状态码时，不会返回任何资源。
305	Use Proxy	所请求的资源必须通过代理访问。
306	Unused	已经被废弃的HTTPS状态码。
400	BadRequest	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
402	Payment Required	保留请求。
403	Forbidden	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
404	NotFound	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
405	MethodNotAllowed	请求中带有该资源不支持的方法。 建议直接修改该请求，不要重试该请求。
406	Not Acceptable	服务器无法根据客户端请求的内容特性完成请求。
407	Proxy Authentication Required	请求要求代理的身份认证，与401类似，但请求者应当使用代理进行授权。
408	Request Time-out	服务器等待请求时发生超时。 客户端可以随时再次提交该请求而无需进行任何更改。
409	Conflict	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
410	Gone	客户端请求的资源已经不存在。 返回该状态码，表明请求的资源已被删除。

状态码	编码	状态码说明
411	Length Required	服务器无法处理客户端发送的不带Content-Length的请求信息。
412	Precondition Failed	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。
413	Request Entity Too Large	由于请求的实体过大，服务器无法处理，因此拒绝请求。为防止客户端的连续请求，服务器可能会关闭连接。如果只是服务器暂时无法处理，则会包含一个Retry-After的响应信息。
414	Request-URI Too Large	请求的URI过长（URI通常为网址），服务器无法处理。
415	Unsupported Media Type	服务器无法处理请求附带的媒体格式。
416	Requested range not satisfiable	客户端请求的范围无效。
417	Expectation Failed	服务器无法满足Expect的请求头信息。
422	UnprocessableEntity	请求格式正确，但是由于含有语义错误，无法响应。
429	TooManyRequests	表明请求超出了客户端访问频率的限制或者服务端接收到多于它能处理的请求。建议客户端读取相应的Retry-After首部，然后等待该首部指出的时间后再重试。
500	InternalServerError	表明服务端能被请求访问到，但是不能理解用户的请求。
501	Not Implemented	服务器不支持请求的功能，无法完成请求。
502	Bad Gateway	充当网关或代理的服务器，从远端服务器接收到了一个无效的请求。
503	ServiceUnavailable	被请求的服务无效。 建议直接修改该请求，不要重试该请求。
504	ServerTimeout	请求在给定的时间内无法完成。客户端仅在为请求指定超时（Timeout）参数时会得到该响应。
505	HTTP Version not supported	服务器不支持请求的HTTPS协议的版本，无法完成处理。

7.2 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见**API网关错误码**进行处理。

状态码	错误码	错误信息	描述	处理措施
202	CSS.0011	This operation cannot be performed because another operation is being performed on the instance or the instance is faulty. Please try again later.	该实例正在进行其它操作或该实例故障，无法执行该操作，请稍后重试。	稍后重试。
400	CSS.0001	Incorrect parameters.	参数错误。	根据具体提示信息修改参数。
400	CSS.0006	The request is null. Enter the request parameters.	请求为空，请输入请求参数。	根据错误信息，输入请求参数。
400	CSS.0017	Invalid parameter letter case.	参数大小写错误。	请根据命名规则重新填写。
400	CSS.0021	The account is frozen.	账户冻结。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.0026	The log size is too large.	日志内容过大。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.0036	Internal error.	内部错误。	请稍后重试或联系技术支持。
400	CSS.0047	The cluster does not exist or has been deleted.	集群不存在或已删除。	集群已经不存在，无法进行操作。
400	CSS.0048	The OBS bucket does not exist.	OBS桶不存在。	请检查OBS桶是否还存在或有访问该桶权限。
400	CSS.0050	Invalid cluster administrator password.	集群管理员密码不合法。	请根据接口参考描述，重新设置密码。
400	CSS.0051	The current policy does not allow the #actionName # operation.	当前策略不允许#actionName#执行。	请为账户配置相关权限。
400	CSS.0052	Insufficient CPU quota.	CPU资源不足。	建议释放掉不用的集群、减少创建CPU个数、扩充配额。
400	CSS.0053	Insufficient resource quota on the node.	节点资源不足。	建议释放不用的集群、减小新节点个数、扩充配额。
400	CSS.0054	Insufficient memory quota.	内存资源不足。	建议释放掉不用的集群、减少需要修改的内存大小、扩充配额。
400	CSS.0055	Insufficient disk quota.	磁盘数量资源不足。	建议释放掉不用的集群、减少创建磁盘大小、扩充配额。

状态码	错误码	错误信息	描述	处理措施
400	CSS.0060	Insufficient disk capacity quota.	磁盘容量资源不足。	建议释放不用的资源、减小所需磁盘容量大小、扩充配额。
400	CSS.0061	Insufficient EIP quota.	弹性IP资源不足。	建议释放不用的资源、扩充配额。
400	CSS.0062	Insufficient ELB quota.	弹性负载均衡资源不足。	建议释放不用的资源、扩充配额。
400	CSS.0063	Insufficient VPC endpoint quota.	终端节点资源不足。	建议释放不用的资源、扩充配额。
400	CSS.0064	Insufficient subnet IP address quota.	子网ip地址不足。	建议释放不用的资源、扩充配额。
400	CSS.0065	The disk has been sold out.	磁盘已售罄。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.0066	Invalid password.	密码不合法。	请根据接口参考描述，重新设置密码。
400	CSS.0068	The flavor has been sold out.	规格已售罄。	请释放不用的资源。
400	CSS.0069	The disk has been sold out.	磁盘已售罄。	请释放资源，或扩充配额。
400	CSS.0070	Interconnection with the enterprise project is not supported.	不支持对接企业项目。	该局点不支持企业项目。
400	CSS.0071	Invalid bandwidth size.	带宽大小不合法。	请根据错误信息修改带宽大小。

状态码	错误码	错误信息	描述	处理措施
400	CSS.0072	Failed to create the SMN agency. Check whether the account has the required permission.	创建SMN委托失败,请查看账号是否有创建SMN委托的权限。	为账户授权。
400	CSS.0074	The number of data nodes (including cold nodes) must be greater than or equal to three.	数据节点(包括冷节点)个数应大于等于3个。	为集群扩容，或取消副本校验。
400	CSS.0075	The target flavor is not supported.	不能变更为目标规格。	目标规格需要小于原规格。
400	CSS.0076	Invalid node role type.	节点角色type非法。	请根据接口参数描述，确认type类型。
400	CSS.0077	The cluster is being scaled out or scaled in.	集群处于扩容或缩容状态。	等待集群状态正常后再进行其他操作。
400	CSS.0078	The cluster is in a status that does not support upgrade.	目标集群状态不满足升级要求。	保证当前集群状态正常。
400	CSS.0080	The default agency already exists. Check whether the agency has the required permissions before using it.	默认委托已经存在，确保权限无误后，可以正常使用。	请确保账户拥有相关权限。

状态码	错误码	错误信息	描述	处理措施
400	CSS.0081	Failed to create the agency. Check whether the account has the permission to create an agency.	创建委托失败，请查看账号是否有创建委托的权限。	为账户授相关权限。
400	CSS.0082	The record does not exist.	记录不存在。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.0083	Some indexes have not been backed up. Perform full index backup in the snapshot module of the cluster.	存在未备份的索引，请到集群快照模块进行全量索引备份。	请到集群快照模块进行全量索引备份。
400	CSS.1111	Invalid scale-out parameters.	扩容参数不合法。	检查扩容参数。
400	CSS.1112	The number of instances has reached the quota.	实例数量达到配额。	删除部分集群或申请扩大配额。
400	CSS.1119	The Kibana public access already exists.	kibana公网访问已经存在。	请勿重复绑定kibana公网访问。
400	CSS.1120	The Kibana public access does not exist.	kibana公网访问不存在。	请绑定Kibana公网访问。

状态码	错误码	错误信息	描述	处理措施
400	CSS.1122	To change the security mode, disable the public access and try again.	安全模式转换，需要先关闭公网访问再重试。	根据错误信息关闭公网访问。
400	CSS.1123	To change the security mode, disable the Kibana public access and try again.	安全模式转换，请先关闭kibana公网访问再重试。	根据错误信息关闭Kibana公网访问。
400	CSS.2026	This is a weak password. Enter a strong password.	这是一个弱密码，请输入一个强度大一些的密码。	请根据错误信息，增强密码强度。
400	CSS.3011	Invalid retention period.	保留天数不合法。	保留天数在1到90天的正整数。
400	CSS.3016	The backup file is being deleted. Try again later.	备份文件正在删除中，请您稍后再进行操作。	等待备份文件删除结束后再进行操作。
400	CSS.3026	The backup file does not exist.	备份文件不存在。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.3030	The new instance volume cannot be smaller than the old instance volume.	新实例卷大小不得小于旧实例卷大小。	根据提示信息修改参数值。
400	CSS.5002	Invalid database type.	数据库类型不合法。	暂时只支持ElasticSearch类型。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5003	Invalid database version.	数据库版本不合法。	Elasticsearch搜索引擎和Kibana目前支持7.6.2和7.10.2版本。
400	CSS.5004	The datastore parameter is not specified.	datastore字段为空。	请补充datastore字段信息。
400	CSS.5005	The database type or version is not supported.	不支持的数据库类型或版本。	Elasticsearch搜索引擎和Kibana目前支持7.6.2和7.10.2版本。
400	CSS.5006	Flavor is illegal.	所选规格不合法。	请根据接口参数描述，查看所需要的规格。
400	CSS.5007	The selected flavors do not exist.	所选规格不存在。	请根据接口参数描述，查看所需要的规格。
400	CSS.5008	Invalid volume type.	卷类型不合法。	请输入正确的卷类型。
400	CSS.5009	Invalid volume size.	卷大小超出允许范围。	请输入正确的范围。
400	CSS.5011	Invalid AZ.	可用区不合法。	请输入正确的可用区。
400	CSS.5017	Invalid retention period of automated backups.	自动备份保留天数无效。	自动备份保留天数为1到90天。
400	CSS.5018	Invalid automated backup period.	自动备份周期无效。	只支持整点，后面需加上时区，格式为“HH:mm z”
400	CSS.5020	Invalid automated backup start time.	自动备份开始时间不合法。	请根据接口参数描述，设置合法备份时间。
400	CSS.5021	The VPC does not exist or does not belong to the user.	VPC不存在，或不属于该用户。	检查VPC是否正确。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5023	The security group does not exist or does not belong to the VPC.	安全组不存在，或不属于该VPC。	检查安全组是否正确。
400	CSS.5033	The volume size is not an integer multiple of 10.	卷大小不是10的整数倍。	扩容时volume必须是4和10的倍数。
400	CSS.5036	Datastore does not exist.	Datastore不存在。	暂只支持Elasticsearch类型。
400	CSS.5037	The parameter does not exist.	参数不存在。	根据对应的api参考确认参数范围。
400	CSS.5038	The value of the parameter is out of range.	参数值超出范围。	根据提示信息修改参数值。
400	CSS.5039	Duplicate parameters.	参数重复。	请删除重复的参数。
400	CSS.5044	Invalid backup prefix.	备份前缀名称设置不合法。	请根据接口参数描述，重新设置备份前缀。
400	CSS.5046	NICS is left blank.	NICS为空。	补充NICS（VPC、子网相关信息）。
400	CSS.5047	Invalid instance quantity.	实例个数不合法。	请根据接口参数描述，确认合法参数范围。
400	CSS.5048	Invalid extended parameters.	扩展参数不合法。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5049	Invalid cluster name.	集群名不合法。	请重新修改集群名称。
400	CSS.5050	The cluster name already exists.	集群名已经存在。	请重新修改集群名称。
400	CSS.5051	Invalid IP address.	IP取值不合法。	检查并输入正确的IP。
400	CSS.5052	Available zone is illegal.	可用区取值不合法。	检查并输入正确的可用区。
400	CSS.5055	Invalid flavor ID.	所选规格ID取值不合法。	请根据接口参数描述，重新修改规格ID。
400	CSS.5056	Invalid string length.	字符串长度不合法。	请根据具体提示信息判断字符长度范围。
400	CSS.5057	The cluster name contains invalid characters or the name length is not allowed.	集群名称含不合法字符或长度不合法。	请重新修改集群名称。
400	CSS.5059	Invalid database username.	数据库用户名不合法。	当前仅支持elasticsearech。
400	CSS.5061	Invalid volume type.	Volume 类型违规。	请根据API参考确认volume类型是否合规。
400	CSS.5063	Invalid value of the Specific parameter.	Specific 字段无效。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5068	Invalid hard disk parameters.	硬盘参数不合法。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5069	Invalid node quantity.	节点个数不合法。	请根据接口参数描述，重新配置节点个数。
400	CSS.5073	Instances of this type cannot be restarted separately.	该类型的实例不允许单独重启。	请勿选择资源中不包含的类型。
400	CSS.5074	Subnet does not belong to VPC	子网不属于vpc	请确认并输入正确的子网参数。
400	CSS.5077	Invalid cluster name.	集群名不合法。	请重新修改集群名称。
400	CSS.5078	Invalid disk size.	磁盘大小违规。	请根据接口参数描述，确认磁盘大小范围。
400	CSS.5087	Inconsistent disk specifications are used for the same type of instances.	同一实例类型不满足磁盘一致性。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5088	The SSL is not globally consistent.	SSL非全局一致。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5090	In this status, the current operation is not allowed.	此状态下不允许当前操作。	请根据错误信息，处理之后再进行操作。
400	CSS.5091	The number of nodes has reached the upper limit. Contact the administrator.	节点个数达到最大值，请联系管理员处理。	请重新调整节点个数。
400	CSS.5092	Invalid storage information.	硬盘信息不合法。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5093	Invalid node specifications.	节点规格信息不合法。	请根据获取实例规格列表接口获取对应的节点规格。
400	CSS.5094	Invalid node scale-out type.	节点扩容类型不合法。	请检查并输入合规的节点类型。
400	CSS.5095	The current instance type does not support this operation.	当前实例类型不支持此操作。	本地磁盘不支持该操作。
400	CSS.5098	Invalid EIP binding type.	EIP绑定类型不合法。	请根据接口参数描述，确认EIP绑定类型。
400	CSS.5106	Invalid instance type.	实例类型不合法。	请输入正确实例类型。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5107	Invalid cluster mode.	集群模式不合法。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5110	Inconsistent parameter groups are used for the same type of instances.	同类型实例参数组不一致。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5111	The backup file ID does not exist.	备份文件ID不存在。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5130	Invalid agency name.	委托名称不合法。	请根据错误信息，判断具体操作。
400	CSS.5134	The total number of master nodes must be an odd number.	master总节点个数需要为奇数。	请根据错误信息修改参数。
400	CSS.5135	The disk size must be a multiple of 20.	磁盘大小需要为20的倍数。	请根据错误信息修改参数。
400	CSS.5137	Invalid tag key.	标签key不合法。	请根据错误信息，修改该参数。
400	CSS.5138	Invalid tag value.	标签value不合法。	请根据错误信息，修改该参数。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5140	The cluster does not support this operation.	该集群不支持此操作。	当前集群不支持该操作，待集群状态正常后再进行操作。
400	CSS.5141	Tag not found.	标签不存在。	请确保请求体中的标签属于该集群。
400	CSS.5158	Only bound public IP addresses can be unbound from clusters.	公网访问只有在绑定状态下才能进行解绑操作。	请确认公网访问绑定状态。
400	CSS.5159	Only unbound public IP addresses can be bound to clusters.	公网访问只有在解绑状态下才能进行绑定操作。	请确认公网访问绑定状态。
400	CSS.5160	Failed to bind the public IP address. Please try again.	公网IP绑定出错,请重试。	请稍后重试或联系技术支持。
400	CSS.5164	Failed to update the public network access whitelist. Please try again.	更新公网访问白名单出错,请重试。	请稍后重试或联系技术支持。
400	CSS.5167	No data nodes are selected.	没有选择数据节点。	请至少选择一个数据节点。
400	CSS.5169	The selected version does not support the security mode.	所选版本不支持安全模式。	详情参考安全模式的用户指南。
400	CSS.5170	The volume parameter is not required for local disk specifications.	本地盘规格不需要配置volume参数。	根据错误信息删除volume参数。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5171	If httpsEnable is set to true, authorityEnable cannot be set to false.	httpsEnable设置为true的情况下authorityEnable不能设置为false。	httpsEnable设置为true的情况下authorityEnable不能设置为false。
400	CSS.5172	The number of instances must be greater than or equal to the number of AZs.	实例个数需要大于等于可用区个数。	请增加实例个数。
400	CSS.5174	Failed to obtain VPC information. Try again later.	Vpc信息获取失败，请稍后重试。	请确保VPC存在，且网络正常。
400	CSS.5177	The https enable parameter is not specified.	缺少https enable。	httpsEnable为必选参数。
400	CSS.5178	Invalid time zone.	时区不合法。	请根据接口参数描述，重新修改时区格式。
400	CSS.5180	Kibana public access is not enabled.	没有开启Kibana公网访问。	打开Kibana公网访问。
400	CSS.5181	The whitelist does not exist.	白名单不存在。	当前白名单为空，请添加后再进行相关操作。
400	CSS.5182	The VPC endpoint service is not enabled.	终端节点服务没有开启。	请开启终端节点服务后再进行相关操作。
400	CSS.5300	This operation is not allowed for yearly/monthly clusters.	包年/包月集群不允许此操作。	禁止该操作。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5301	This operation is not allowed because the cluster has been frozen.	集群已冻结，禁止此操作。	如果因欠费冻结，续费即可。因其他原因冻结，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.5302	Invalid subscription period type.	订购周期类型不合法。	请根据接口参数描述，修改订购周期类型参数。
400	CSS.5303	Invalid subscription period.	订购周期数不合法。	请根据接口参数描述，修改订购周期参数。
400	CSS.5304	Invalid auto-renewal parameter.	自动续费参数不合法。	请根据接口参数描述，修改续费参数。
400	CSS.5305	Invalid automated payment parameters.	自动支付参数不合法。	请根据接口参数描述，修改自动支付参数。
400	CSS.5306	An unpaid order exists.	存在未支付的订单。	支付该订单，或取消该订单。
400	CSS.5307	An order is being processed.	存在处理中的订单。	等待订单处理完成。
400	CSS.5308	The bandwidth exceeds the limit.	带宽超过限额。	缩小带宽大小。

状态码	错误码	错误信息	描述	处理措施
400	CSS.5310	If httpsEnable is set to false, public network access cannot be enabled. If authorityEnable is set to false, Kibana public network access cannot be enabled.	httpsEnable为false，无法开启公网访问，authorityEnable为false无法开启kibana公网访问。	根据错误信息开启相关功能。
400	CSS.5312	Nodes and disks cannot be scaled out at the same time in a cluster billed in yearly/ monthly mode.	包周期集群不支持节点和磁盘同时扩容。	不支持该操作。
400	CSS.5313	Insufficient permissions on the VPC endpoint. Check whether the user has the VPC endpoint administrator role.	终端节点权限不足，请检查用户是否有VPCEndpoint Administrator角色。	根据错误信息，检查相关权限。
400	CSS.6000	Failed to create the cluster. Contact technical support or try again later.	创建集群失败，请联系技术支持或者稍后重试。	请稍后重试，如果仍然失败，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.6001	Failed to scale out the cluster. Contact technical support or try again later.	扩容集群失败，请联系技术支持或者稍后重试。	请稍后重试，如果仍然失败，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.6003	Failed to restore the cluster. Contact technical support or try again later.	恢复集群失败，请联系技术支持或者稍后重试。	请稍后重试，如果仍然失败，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.6004	Failed to create the node due to an ECS exception. Contact technical support or try again later. Error code: #failedReason#.	创建节点失败，原因：ECS异常，错误码#failedReason#! 请联系技术支持或稍后重试。	请稍后重试，如果仍然失败，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
400	CSS.6009	Failed to create the snapshot. Contact technical support or try again later.	快照创建失败，请联系技术支持或者稍后重试。	请稍后重试，如果仍然失败，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。

状态码	错误码	错误信息	描述	处理措施
400	CSS.6012	Failed to restore the snapshot because the OBS object does not exist or has been deleted.	恢复快照失败，因为OBS对象不存在或已被删除。	请检查OBS桶是否存在。
400	CSS.6013	Parameters cannot be modified because another operation is being performed on the cluster.	集群正在进行其它操作，无法修改参数。	请确保集群处于可用状态时，再进行修改参数操作。
400	CSS.6014	Unchangeable parameters are contained.	设置的参数中包含不可修改的参数。	请检查提交的修改参数请求中，是否含有不可修改的参数值，将对应参数删除后重试操作。
400	CSS.6015	Parameters are not in the YAML format.	设置的参数不满足YAML格式。	请修改参数格式后重试操作。
403	CSS.0010	The instance ID or user ID may be null, or the operation is not authorized.	实例ID和用户ID可能为空，或者操作越权。	请检查账号权限。
403	CSS.0015	No resources are found or the access is denied.	访问资源不存在，或者无访问权限。	权限不足，添加权限后再操作。
403	CSS.5045	The current tenant is not allowed to perform this operation.	当前租户不允许操作。	确认租户是否冻结或是否有相关权限。
404	CSS.0002	The instance does not exist.	CSS不存在。	检查权限以及资源是否存在。

状态码	错误码	错误信息	描述	处理措施
404	CSS.0007	The VM information of the DB instance cannot be found.	请检查数据库，确认实例的虚拟机信息，找不到该实例的虚拟机信息。	请稍后重试或联系技术支持。
404	CSS.0016	Resource not found.	未找到资源。	根据提示信息判断资源是否存在。
404	CSS.0022	The instance does not exist or has been deleted.	实例不存在或已删除。	请确保资源是否存在。
404	CSS.0028	The object does not exist.	对象不存在。	根据具体提示信息判断，当前资源是否存在。
404	CSS.1103	The instance does not exist!	该实例不存在！	请确认集群是否存在。
404	CSS.3027	The cluster does not exist!	集群不存在！	集群不存在，请勿对该集群进行任何操作。
404	CSS.5012	The region or AZ does not exist.	Region或可用区不存在。	请输入正确的Region或可用区。
404	CSS.5022	The subnet does not exist or does not belong to the VPC.	子网不存在，或不属于该vpc。	检查子网是否存在，并且属于该VPC。
404	CSS.5042	The source instance does not exist.	源实例不存在。	请求体中包含集群中不存在的实例。
409	CSS.0012	Operation is not allowed because the cluster has a standby instance.	该实例有备机，不能进行该操作。	等待集群备份结束后进行操作
409	CSS.0029	The object is being created. Please wait.	对象正在创建中，请稍等。	集群或快照正在创建，请勿进行其他操作。
409	CSS.0031	The object already exists.	对象已存在。	请重新修改对象名称。

状态码	错误码	错误信息	描述	处理措施
409	CSS.0043	The snapshot name already exists.	快照名已存在。	请重新修改快照名称。
409	CSS.1102	The instance name already exists.	实例名称已存在。	请重新修改实例名称。
409	CSS.3004	Backup is in progress.	正在备份中，请稍等。	根据错误信息，等待集群状态成功后，再进行操作。
413	CSS.0030	The number of snapshots reaches the quota.	快照数量达到配额。	建议释放掉无需备份的资源。
415	CSS.0004	Invalid operation.	操作不合法。	根据错误信息，修改该参数。
415	CSS.5108	The operation is not supported.	不支持该操作。	不支持该操作。
419	CSS.0023	Insufficient resource.	资源不足。	请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
422	CSS.0040	The parameter cannot be processed.	参数无法处理。	请稍后重试或联系技术支持。
500	CSS.0005	Server error.	服务器错误。	内部错误，请联系技术支持。您可以登录华为云官网，在华为云首页，单击右上角的“联系我们”，反馈您所遇到的问题，技术支持会快速为您定位问题，解决问题。
500	CSS.1004	Database service error.	数据库服务异常。	请稍后重试或联系技术支持。

状态码	错误码	错误信息	描述	处理措施
500	CSS.9999	Failed to process the request.	请求处理失败。	根据提示信息检查请求路径以及请求体的正确性。

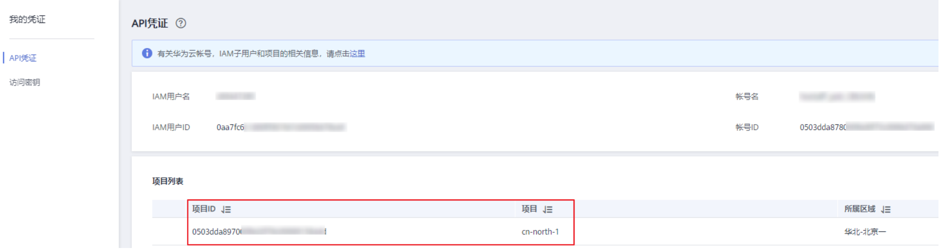
7.3 获取项目 ID 和名称

在调用接口的时候，部分请求中需要填入项目ID（project_id）或项目名称（project name），所以需要获取到项目ID和名称。

从控制台获取项目 ID 和名称

1. 登录管理控制台。
2. 在页面右上角单击用户名，在下拉列表中单击“我的凭证”，进入“API凭证”页面。
3. 在项目列表中查看“项目ID”和“项目”（“项目”即项目名称）。

图 7-1 查看项目 ID 和名称



API凭证

有关华为云帐号，IAM子用户和项目的相关信息，请点击这里

IAM用户名 [redacted] 帐号名 [redacted]

IAM用户ID 0aa7fc6[redacted] 帐号ID 0503dda8780[redacted]

项目列表

项目ID	项目	所属区域
0503dda8970[redacted]	cn-north-1	华北-北京一

调用 API 获取项目 ID

项目ID还用通过调用查询指定条件下的项目信息API获取。

获取项目ID的接口为“GET https://{iam-endpoint}/v3/projects”，其中{iam-endpoint}为IAM的终端节点，可以参考终端节点获取，接口的认证鉴权请参见认证鉴权。

响应示例如下，例如CSS部署的区域为“xxx”，在响应消息体中搜索“name”为“xxx”，其对应的“projects”下的“id”即为项目ID。

```
{
   "projects": [
      {
         "domain_id": "65382450e8f64ac0870cd180d14exxxx",
         "is_domain": false,
         "parent_id": "65382450e8f64ac0870cd180d14exxxx",
         "name": "xxx",    //项目名称，即部署区域名称
         "description": "",
         "links": {
            "next": null,
            "previous": null,
            "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897dxxxx"
         },
         "id": "a4a5d4098fb4474fa22cd05f897dxxxx",    //项目ID
         "enabled": true
      }
```

```
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```