容器镜像服务（企业版）

API 参考

文档版本 1

发布日期 2026-09-09

华为云计算技术有限公司

版权所有 © 华为云计算技术有限公司 2026。 保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明

和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

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 1
1.1 概述 1
1.2 调用说明 1
1.3 终端节点（Endpoint） 1
1.4 约束与限制 1
1.5 基本概念 2
2 如何调用 API 4
2.1 构造请求 4
2.2 认证鉴权 7
2.3 返回结果 9
3 API 概览 11
4 API 17
4.1 实例管理 17
4.1.1 创建实例 - CreateInstance 17
4.1.2 查询实例列表 - ListInstance 36
4.1.3 获取实例详情 - ShowInstance 43
4.1.4 删除实例 - DeleteInstance 50
4.1.5 查看实例配置 - ShowInstanceConfiguration 56
4.1.6 更新实例配置 - UpdateInstanceConfiguration 62
4.1.7 获取上传下载的相关审计日志列表 - ListAuditLogs 68
4.1.8 获取实例统计数据 - ListInstanceStatistics 75
4.2 命名空间管理 82
4.2.1 创建命名空间 - CreateInstanceNamespace 82
4.2.2 获取命名空间列表 - ListInstanceNamespaces 89
4.2.3 获取命名空间详情 - ShowInstanceNamespace 98
4.2.4 修改命名空间 - UpdateInstanceNamespace 106
4.2.5 删除命名空间 - DeleteInstanceNamespace 114
4.3 制品仓库管理 120
4.3.1 获取制品仓库列表 - ListInstanceRepositories 120
4.3.2 获取命名空间下所有制品仓库列表 - ListNamespaceRepositories 128
4.3.3 获取制品仓库详情 - ShowInstanceRepository 135
4.3.4 修改制品仓库 - UpdateInstanceRepository 142

 版权所有 © 华为云计算技术有限公司

4.3.5 删除制品仓库 - DeleteInstanceRepository 148
4.3.6 获取当前项目下所有企业仓库实例的仓库列表 - ListAllInstanceRepositories 154
4.4 制品版本管理 158
4.4.1 获取制品版本列表 - ListInstanceArtifacts 158
4.4.2 获取制品版本详情 - ShowInstanceArtifact 166
4.4.3 删除制品版本 - DeleteInstanceArtifact 176
4.4.4 获取制品附件列表 - ListInstanceAccessories 182
4.4.5 获取制品附加信息 - ShowInstanceArtifactAddition 189
4.4.6 获取仓库实例的所有制品版本列表 - ListInstanceAllArtifacts 196
4.4.7 获取制品仓库的 Tag 列表 - ListInstanceTags 204
4.4.8 删除制品的 Tag - DeleteInstanceTag 209
4.4.9 获取制品 Manifest 信息 - ShowInstanceArtifactManifest 212
4.5 访问凭证管理 216
4.5.1 获取临时访问凭证 - CreateInstanceTempCredential 216
4.5.2 创建长期访问凭证 - CreateInstanceLtCredential 222
4.5.3 启用/停用长期访问凭证 - UpdateInstanceLtCredential 228
4.5.4 删除长期访问凭证 - DeleteInstanceLtCredential 234
4.5.5 查询长期访问凭证列表 - ListInstanceLtCredentials 240
4.6 标签管理 247
4.6.1 按照标签检索资源列表 - ListInstanceResourceInstances 247
4.6.2 按照标签检索资源数量 - ShowInstanceResourceInstancesCount 255
4.6.3 批量添加资源标签 - CreateInstanceResourceTags 263
4.6.4 批量删除资源标签 - DeleteInstanceResourceTags 270
4.6.5 查询项目下所有实例标签 - ListInstanceProjectTags 277
4.6.6 查询某种资源类型的标签列表 - ListResourcesTagsByProjectId 283
4.6.7 查询资源标签 - ListInstanceResourceTags 287
4.6.8 按照标签检索子资源列表 - ListSubResourceInstances 293
4.6.9 按照标签检索子资源数量 - ShowSubResourceInstancesCount 302
4.6.10 批量添加子资源标签 - CreateSubResourceTags 309
4.6.11 批量删除子资源标签 - DeleteSubResourceTags 316
4.6.12 查询实例下所有命名空间标签 - ListNamespaceTags 323
4.6.13 查询子资源标签 - ListSubResourceTags 329
4.7 域名管理 335
4.7.1 增加域名 - AddDomainName 335
4.7.2 查询所有域名列表 - ListDomainNames 342
4.7.3 更新域名 - UpdateDomainName 349
4.7.4 删除域名 - DeleteDomainName 356
4.8 特性开关 362
4.8.1 查询特性开关信息 - ListFeatureGates 362
4.8.2 查询全局特性开关信息 - ListGlobalFeatureGates 368
4.9 访问控制管理 374
4.9.1 新增内网访问 - CreateInstanceInternalEndpoint 374

 版权所有 © 华为云计算技术有限公司

4.9.2 获取内网访问列表 - ListInstanceInternalEndpoints....380
4.9.3 查询内网访问详情 - ShowInstanceInternalEndpoint....388
4.9.4 删除内网访问 - DeleteInstanceInternalEndpoint....394
4.9.5 开启或关闭公网访问 - CreateInstanceEndpointPolicy....400
4.9.6 更新公网访问白名单 - UpdateInstanceEndpointPolicy....406
4.9.7 获取公网访问信息 - ShowInstanceEndpointPolicy....412
4.10 触发器管理....419
4.10.1 创建触发器 - CreateInstanceWebhook....419
4.10.2 获取触发器列表 - ListInstanceWebhooks....429
4.10.3 获取触发器详情 - ShowInstanceWebhook....438
4.10.4 修改触发器 - UpdateInstanceWebhook....446
4.10.5 删除触发器 - DeleteInstanceWebhook....455
4.10.6 获取触发器执行任务列表 - ListInstanceWebhookJobs....461
4.11 任务管理....469
4.11.1 获取任务列表 - ListInstanceJobs....469
4.11.2 获取任务详情 - ShowInstanceJob....476
4.11.3 删除任务 - DeleteInstanceJob....482
4.12 镜像同步管理....488
4.12.1 创建镜像同步的目标仓库 - CreateInstanceRegistry....488
4.12.2 获取镜像同步的目标仓库列表 - ListInstanceRegistries....502
4.12.3 获取镜像同步的目标仓库详情 - ShowInstanceRegistry....510
4.12.4 修改镜像同步的目标仓库 - UpdateInstanceRegistry....517
4.12.5 删除镜像同步的目标仓库 - DeleteInstanceRegistry....525
4.12.6 创建镜像同步策略 - CreateInstanceReplicationPolicy....530
4.12.7 获取镜像同步策略列表 - ListInstanceReplicationPolicies....544
4.12.8 获取镜像同步策略详情 - ShowInstanceReplicationPolicy....553
4.12.9 修改镜像同步策略 - UpdateInstanceReplicationPolicy....561
4.12.10 删除镜像同步策略 - DeleteInstanceReplicationPolicy....571
4.12.11 手动执行镜像同步策略 - ExecuteInstanceReplicationPolicy....576
4.12.12 获取镜像同步策略执行记录列表 - ListInstanceReplicationPolicyExecutions....582
4.12.13 停止镜像同步任务 - StopInstanceReplicationPolicyExecution....590
4.12.14 获取镜像同步策略执行任务的镜像列表 - ListInstanceReplicationPolicyExecTasks....596
4.12.15 获取镜像同步策略执行任务的镜像版本列表 - ListInstanceReplicationPolicyExecSubTasks....603
4.13 镜像签名管理....611
4.13.1 获取签名策略列表 - ListInstanceSignPolicies....611
4.13.2 创建签名策略 - CreateInstanceSignPolicy....623
4.13.3 获取签名策略详情 - ShowInstanceSignPolicy....633
4.13.4 修改签名策略 - UpdateInstanceSignPolicy....641
4.13.5 删除签名策略 - DeleteInstanceSignPolicy....652
4.13.6 手动执行签名策略 - ExecuteInstanceSignPolicy....658
4.13.7 获取签名执行记录列表 - ListInstanceSignPolicyExecutions....664
4.13.8 获取签名执行记录任务的镜像列表 - ListInstanceSignPolicyExecTasks....671

 版权所有 © 华为云计算技术有限公司

4.13.9 获取签名策略执行任务的镜像版本列表 - ListInstanceSignatureExecutionSubtasks....678
4.14 老化策略管理....686
4.14.1 创建老化策略 - CreateInstanceRetentionPolicy....686
4.14.2 获取老化策略列表 - ListInstanceRetentionPolicies....696
4.14.3 获取老化策略详情 - ShowInstanceRetentionPolicy....705
4.14.4 修改老化策略 - UpdateInstanceRetentionPolicy....714
4.14.5 删除老化策略 - DeleteInstanceRetentionPolicy....724
4.14.6 执行老化策略 - ExecuteInstanceRetentionPolicy....730
4.14.7 获取老化策略执行记录列表 - ListInstanceRetentionPolicyExecutions....736
4.14.8 获取老化策略执行任务的镜像列表 - ListInstanceRetentionPolicyExecTasks....743
4.14.9 获取老化策略执行任务的镜像版本列表 - ListInstanceRetentionPolicyExecSubTasks....751
4.15 镜像版本不可变....759
4.15.1 创建不可变 Tag 策略 - CreateImmutableRule....759
4.15.2 获取不可变 Tag 策略列表 - ListImmutableRules....767
4.15.3 修改不可变 Tag 策略 - UpdateImmutableRule....775
4.15.4 删除不可变 Tag 策略 - DeleteImmutableRule....783
4.16 制品扫描....789
4.16.1 手动启动制品扫描 - StartManualScanning....789
4.16.2 获取制品扫描的漏洞信息 - ListInstanceArtifactVulnerabilities....795
4.17 制品清理....803
4.17.1 查询制品清理的任务列表 - ListGcTasks....803
4.17.2 查询制品清理的任务详情 - ShowGcTask....811
4.17.3 停止制品清理任务 - StopGcTask....817
4.17.4 获取制品清理的计划信息 - ShowGcSchedule....823
4.17.5 执行制品清理计划 - ExecuteGcSchedule....830
4.17.6 配置制品清理计划 - UpdateGcSchedule....837
5 权限和授权项....845
5.1 权限和授权项说明....845
5.2 角色与策略授权参考（旧版 IAM）....846
5.3 身份策略授权参考（新版 IAM）....847
6 附录....885
6.1 状态码....885
6.2 错误码....886
6.3 获取项目 ID....898
6.4 获取用户名和用户 ID....899

版权所有 © 华为云计算技术有限公司

1 使用前必读

1.1 概述

欢迎使用容器镜像服务企业版（Software Repository for Container，SWR）。容器镜像服务企业版是一种支持镜像全生命周期管理的服务，提供简单易用、安全可靠的镜像管理功能，帮助您快速部署容器化服务。

您可以使用本文档提供API对容器镜像服务企业版进行相关操作，如创建、删除、查询等。

在调用容器镜像服务企业版API之前，请确保已经充分了解容器镜像服务企业版相关概念，详细信息请参见“**什么是容器镜像服务**”。

1.2 调用说明

容器镜像服务企业版提供了REST（Representational State Transfer）风格API，支持您通过HTTPS请求调用。

调用方法请参见**如何调用API**。

同时容器镜像服务企业版还提供多种编程语言的SDK供您使用，SDK的使用方法请参见**https://sdkcenter.developer.huaweicloud.com/zh-cn/?product=SWR**。

1.3 终端节点（Endpoint）

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**中查询容器镜像服务企业版的终端节点。

请您根据业务需要选择对应区域的终端节点。

1.4 约束与限制

- 您能创建的容器镜像服务企业版组织数量与配额有关系，如**表1-1**所示。

版权所有 © 华为云计算技术有限公司

表 1-1 配额

资源类型	配额
组织数量	5

如果您需要扩大配额，请参见“**如何申请扩大配额**”。

- 更详细的限制请参见具体API的说明。

1.5 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他们进行日常管理工作。

- 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**我的凭证**下，您可以查看账号ID和IAM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（Region）

  从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。

  详情请参见**区域和可用区**。

- 可用区（AZ，Availability Zone）

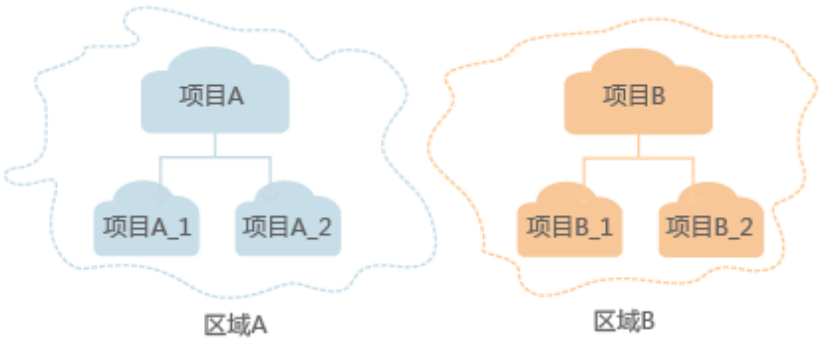
  一个可用区是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

- 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，并在子项目中创建资源，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。

版权所有 © 华为云计算技术有限公司

图 1-1 项目隔离模型



同样在**我的凭证**下，您可以查看项目ID。

- 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《**企业管理用户指南**》。

版权所有 © 华为云计算技术有限公司

2 如何调用 API

2.1 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}

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

表 2-1 URI 中的参数说明

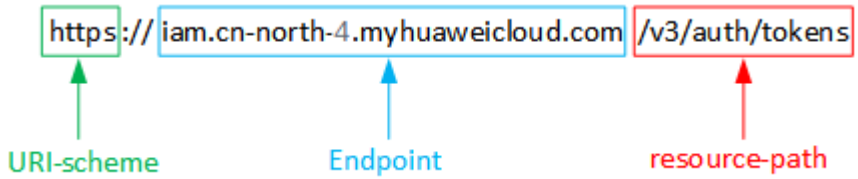
参数	描述
URI-scheme	表示用于传输请求的协议，当前所有API均采用**HTTPS**协议。
Endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从**地区和终端节点**获取。 例如IAM服务在“华北-北京四”区域的Endpoint为“iam.cn-north-4.myhuaweicloud.com”。
resource-path	资源路径，也即API访问路径。从具体API的URI模块获取，例如“获取用户Token”API的resource-path为“/v3/auth/tokens”。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。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。

版权所有 © 华为云计算技术有限公司

例如您需要获取IAM在“华北-北京四”区域的Token，则需使用“华北-北京四”区域的Endpoint（iam.cn-north-4.myhuaweicloud.com），并在**获取用户Token**的URI部分找到resource-path（/v3/auth/tokens），拼接起来如下所示。

```
https://iam.cn-north-4.myhuaweicloud.com/v3/auth/tokens
```

图 2-1 URI 示意图



说明

为查看方便，在每个具体API的URI部分，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。

表 2-2 HTTP 方法

方法	说明
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**获取用户Token**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-1.myhuaweicloud.com/v3/auth/tokens
```

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2-3**。

版权所有 © 华为云计算技术有限公司

表 2-3 公共请求消息头

名称	描述	是否必选	示例
Host	请求的服务器信息，从服务API的URL中获取。值为hostname[:port]。端口缺省时使用默认的端口，https的默认端口为443。	否 使用AK/SK认证时该字段必选。	code.test.com or code.test.com:443
Content-Type	消息体的类型（格式）。推荐用户使用默认值application/json，有其他取值时会在具体接口中专门说明。	是	application/json
Content-Length	请求body长度，单位为Byte。	否	3495
X-Project-Id	project id，项目编号。请参考**获取项目ID**章节获取项目编号。	否 如果是专属云场景采用AK/SK认证方式的接口请求或者多project场景采用AK/SK认证的接口请求，则该字段必选。	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	用户Token。 用户Token也就是调用**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（Headers）中包含的“X-Subject-Token”的值即为Token值。	否 使用Token认证时该字段必选。	注：以下仅为Token示例片段 MIIPAgYJKoZIhvcNAQcCo...ggg1BBIINPXsidG9rZ

说明

API同时支持使用AK/SK认证，AK/SK认证是使用SDK对请求进行签名，签名过程会自动往请求中添加Authorization（签名认证信息）和X-Sdk-Date（请求发送的时间）请求头。

AK/SK认证的详细说明请参见**认证鉴权**的“AK/SK认证”。

版权所有 © 华为云计算技术有限公司

对于**获取用户Token**接口，由于不需要认证，所以只添加“Content-Type”即可，添加消息头后的请求如下所示。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

请求消息体（可选）

该部分可选。请求消息体通常以结构化格式（如JSON或XML）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。若请求消息体中的参数支持中文，则中文字符必须为UTF-8编码。

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**获取用户Token**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，加粗的斜体字段需要根据实际值填写，其中***username***为用户名，***domainname***为用户所属的账号名称，*********为用户登录密码，***xxxxxxxxxxxxxxxxxx***为project的名称，如“cn-north-1”，您可以从**地区和终端节点**获取。

说明

scope参数定义了Token的作用域，下面示例中获取的Token仅能访问project下的资源。您还可以设置Token的作用域为某个账号下所有资源或账号的某个project下的资源，详细定义请参见**获取用户Token**。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxxxxxxxxxxxx"
            }
        }
    }
```

到这里为止这个请求需要的内容就具备齐全了，您可以使用**curl**、**Postman**或直接编写代码等方式发送请求调用API。对于获取用户Token接口，返回的响应消息头中“x-subject-token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

2.2 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。

版权所有 © 华为云计算技术有限公司

- Token认证：通过Token认证调用请求。
- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key）加密调用请求。推荐使用AK/SK认证，其安全性比Token认证要高。

Token 认证

说明

Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。

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。

云服务存在两种部署方式：项目级服务和全局级服务。其中：

- 项目级服务需要获取项目级别的Token，此时请求body中**auth.scope**的取值为**project**。
- 全局级服务需要获取全局级别的Token，此时请求body中**auth.scope**的取值为**domain**。

调用本服务API需要项目级别的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",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxx"
            }
        }
    }
}
```

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-1.myhuaweicloud.com/v3/auth/projects
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**。

版权所有 © 华为云计算技术有限公司

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**。

> **说明**
>
> 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。

2.3 返回结果

状态码

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**获取用户Token**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**获取用户Token**接口，返回如**图2-2**所示的消息头，其中“x-subject-token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

版权所有 © 华为云计算技术有限公司

图 2-2 获取用户 Token 响应消息头

响应消息体（可选）

该部分可选。响应消息体通常以结构化格式（如JSON或XML）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**获取用户Token**接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "az-01",
......
```

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": "The format of message is error",
    "error_code": "AS.0001"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

版权所有 © 华为云计算技术有限公司

3 API 概览

通过使用容器镜像服务企业版提供的接口，您可以实现对组织、镜像仓库、镜像信息的查询，也可以实现对组织、镜像仓库、镜像的管理操作。

实例管理

API	说明
创建实例	创建企业仓库实例
查询实例列表	创建企业仓库实例
获取实例详情	获取实例详情
删除实例	删除实例
查看实例配置	查看实例配置
修改实例配置	修改实例配置
获取上传下载的相关审计日志列表	获取上传下载的相关审计日志列表
获取实例统计数据	获取实例统计数据

命名空间管理

API	说明
创建命名空间	创建命名空间
获取命名空间列表	获取命名空间列表
获取命名空间详情	获取命名空间详情
修改命名空间	修改命名空间
删除命名空间	删除命名空间

版权所有 © 华为云计算技术有限公司

制品仓库管理

API	说明
获取制品仓库列表	获取制品仓库列表
获取命名空间下所有制品仓库列表	获取命名空间下所有制品仓库列表
获取制品仓库详情	获取制品仓库详情
修改制品仓库	修改制品仓库
删除制品仓库	删除制品仓库

制品版本管理

API	说明
获取制品版本列表	获取制品版本列表
获取制品详情	获取制品详情
删除制品	删除制品
获取制品附件列表	获取制品附件列表
获取制品附加信息	获取制品附加信息
获取仓库治理的所有制品版本列表	获取仓库治理的所有制品版本列表
获取制品仓库的Tag列表	获取制品仓库的Tag列表

访问凭证管理

API	说明
获取临时访问凭证	获取临时访问凭证
创建长期访问凭证	创建长期访问凭证
启用/停用长期访问凭证	启用/停用长期访问凭证
删除长期访问凭证	删除长期访问凭证
查询长期访问凭证列表	查询长期访问凭证列表

标签管理

API	说明
按照标签检索资源列表	按照标签检索资源列表

版权所有 © 华为云计算技术有限公司

按照标签检索资源数量	按照标签检索资源数量
批量添加资源标签	批量添加资源标签
批量删除资源标签	批量删除资源标签
查询项目下所有实例标签	查询项目下所有实例标签
查询资源标签	查询资源标签
按照标签检索子资源列表	按照标签检索子资源列表
按照标签检索子资源数量	按照标签检索子资源数量
批量添加子资源标签	批量添加子资源标签
批量删除子资源标签	批量删除子资源标签
查询实例下所有命名空间标签	查询实例下所有命名空间标签
查询子资源标签	查询子资源标签

域名管理

API	说明
增加域名	增加域名
查询所有域名列表	查询所有域名列表
更新域名	更新域名
删除域名	删除域名

特性开关

API	说明
查询特性开关信息	查询特性开关信息
查询全局特性开关信息	查询全局特性开关信息

访问控制管理

API	说明
新增内网访问	新增内网访问
获取内网访问列表	获取内网访问列表
查询内网访问详情	查询内网访问详情

版权所有 © 华为云计算技术有限公司

删除内网访问	删除内网访问
开启或关闭公网访问	开启或关闭公网访问
更新公网访问白名单	更新公网访问白名单
获取公网访问信息	获取公网访问信息

触发器管理

API	说明
创建触发器策略	创建触发器策略
获取触发器策略列表	获取触发器策略列表
获取触发器策略详情	获取触发器策略详情
修改触发器策略	修改触发器策略
删除触发器策略	删除触发器策略
获取触发器执行任务列表	获取触发器执行任务列表

任务管理

API	说明
获取任务列表	获取任务列表
获取任务详情	获取任务详情
删除任务	删除任务

镜像同步管理

API	说明
创建镜像同步的目标仓库	创建镜像同步的目标仓库
获取镜像同步的目标仓库列表	获取镜像同步的目标仓库列表
获取镜像同步的目标仓库详情	获取镜像同步的目标仓库详情
修改镜像同步的目标仓库	修改镜像同步的目标仓库
删除镜像同步的目标仓库	删除镜像同步的目标仓库
创建镜像同步策略	创建镜像同步策略
获取镜像同步策略列表	获取镜像同步策略列表

版权所有 © 华为云计算技术有限公司

获取镜像同步策略详情	获取镜像同步策略详情
修改镜像同步策略	修改镜像同步策略
删除镜像同步策略	删除镜像同步策略
手动执行镜像同步策略	手动执行镜像同步策略
获取镜像同步策略执行记录列表	获取镜像同步策略执行记录列表
停止镜像同步任务	停止镜像同步任务
获取镜像同步策略执行任务的镜像列表	获取镜像同步策略执行任务的镜像列表
获取镜像同步策略执行任务的镜像版本列表	获取镜像同步策略执行任务的镜像版本列表

镜像签名管理

API	说明
获取签名策略列表	获取签名策略列表
创建签名策略	创建签名策略
获取签名策略详情	获取签名策略详情
修改签名策略	修改签名策略
删除签名策略	删除签名策略
手动执行签名策略	手动执行签名策略
获取签名执行记录列表	获取签名执行记录列表
获取签名执行记录任务的镜像列表	获取签名执行记录任务的镜像列表
获取签名策略执行任务的镜像版本列表	获取签名策略执行任务的镜像版本列表

老化策略管理

API	说明
创建老化策略	创建老化策略
获取老化策略列表	获取老化策略列表
获取老化策略详情	获取老化策略详情
修改老化策略	修改老化策略

版权所有 © 华为云计算技术有限公司

删除老化策略	删除老化策略
执行老化策略	执行老化策略
获取老化策略执行记录列表	获取老化策略执行记录列表
获取老化策略执行任务的镜像列表	获取老化策略执行任务的镜像列表
获取老化策略执行任务的镜像版本列表	获取老化策略执行任务的镜像版本列表

镜像版本不可变

API	说明
创建不可变Tag策略	创建不可变Tag策略
获取不可变Tag策略列表	获取不可变Tag策略列表
修改不可变Tag策略	修改不可变Tag策略
删除不可变Tag策略	删除不可变Tag策略

版权所有 © 华为云计算技术有限公司

4 API

4.1 实例管理

4.1.1 创建实例 - CreateInstance

功能介绍

创建企业仓库实例

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:create	Write	instance *	-	-	-

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
		-	• g:EnterpriseProjectId • g:RequestTag/<tag-key> • g:TagKeys • swr:VpcId • swr:SubnetId • swr:EnableObsEncrypt		

URI

POST /v2/{project_id}/instances

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

请求参数

表 4-2 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-3 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	企业仓库实例名称，长度范围为3-48个字符，以小写字母开头，支持小写字母、数字和中划线(-)，不能以中划线(-)结尾，中划线不能连续
description	否	String	企业仓库实例描述
spec	是	String	企业仓库实例规格，目前支持企业版(swr.ee.professional)
vpc_id	是	String	用户虚拟私有云ID
subnet_id	是	String	用户子网的网络ID
project_id	是	String	vpc和子网所在项目编号
charge_mode	是	String	实例计费模式，目前只支持按需(postPaid)
enterprise_project_id	是	String	企业项目编号
resource_tags	否	Array of **resource_tags** objects	指定资源tag.
obs_encrypt	否	Boolean	obs桶是否开启加密, 如果开启了加密，则可以根据encrypt_type指定加密算法
encrypt_type	否	String	obs桶加密类型，空值使用AES-256加密算法, gm为SM4国密加密算法
obs_bucket_name	否	String	指定obs桶的名称，当指定自定义obs桶之后，则无需对obs_encrypt、encrypt_type进行传值。
enable_intranet_access	否	Boolean	是否创建内网访问，不传此值，默认会创建内网访问。
obs_enc_kms_key_id	否	String	KMS的密钥ID，用于对自动创建的OBS桶进行加密。密钥的加密算法支持AES-256和SM4。指定该字段时必须开启OBS桶加密开关，且不能指定自定义OBS桶。

版权所有 © 华为云计算技术有限公司

表 4-4 resource_tags

参数	是否必选	参数类型	描述
key	否	String	tag键值.
value	否	String	tag值.

响应参数

状态码：202

表 4-5 响应 Body 参数

参数	参数类型	描述
job_id	String	任务ID
instance_id	String	实例ID

状态码：400

表 4-6 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-7 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-8 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-9 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-10 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

- 创建自定义obs桶的企业实例

```
POST https://{endpoint}/v2/{project_id}/instances

{
  "name" : "my-instance",
  "spec" : "swr.ee.professional",
  "vpc_id" : "vpc-12345678",
  "subnet_id" : "subnet-87654321",
  "project_id" : "project-123456",
  "charge_mode" : "postPaid",
```


版权所有 © 华为云计算技术有限公司

```
  "enterprise_project_id" : "0",
  "resource_tags" : [ {
    "key" : "test",
    "value" : "test"
  } ],
  "obs_bucket_name" : "obs_bucket1"
}
```

- 创建obs桶默认算法加密的企业实例

```
POST https://{endpoint}/v2/{project_id}/instances

{
  "name" : "my-instance",
  "description" : "My test instance",
  "spec" : "swr.ee.professional",
  "vpc_id" : "vpc-12345678",
  "subnet_id" : "subnet-87654321",
  "project_id" : "myproject-12345",
  "charge_mode" : "postPaid",
  "enterprise_project_id" : "0",
  "resource_tags" : [ {
    "key" : "test",
    "value" : "test"
  } ],
  "obs_encrypt" : true,
  "encrypt_type" : ""
}
```

- 创建obs桶国密加密的企业实例

```
POST https://{endpoint}/v2/{project_id}/instances

{
  "name" : "my-instance",
  "description" : "My test instance",
  "spec" : "swr.ee.professional",
  "vpc_id" : "vpc-12345678",
  "subnet_id" : "subnet-87654321",
  "project_id" : "myproject-12345",
  "charge_mode" : "postPaid",
  "enterprise_project_id" : "0",
  "resource_tags" : [ {
    "key" : "test",
    "value" : "test"
  } ],
  "obs_encrypt" : true,
  "encrypt_type" : "gm"
}
```

- 创建OBS桶时使用已有的KMS密钥进行加密

```
POST https://{endpoint}/v2/{project_id}/instances

{
  "name" : "my-instance",
  "description" : "My test instance",
  "spec" : "swr.ee.professional",
  "vpc_id" : "vpc-12345678",
  "subnet_id" : "subnet-87654321",
  "project_id" : "myproject-12345",
  "charge_mode" : "postPaid",
  "enterprise_project_id" : "0",
  "resource_tags" : [ {
    "key" : "test",
    "value" : "test"
  } ],
  "obs_encrypt" : true,
  "obs_enc_kms_key_id" : "123456101"
}
```

 版权所有 © 华为云计算技术有限公司

响应示例

状态码：202

创建实例成功

```
{
  "job_id" : "d8d691c4-27c4-4199-8c47-19dea40e5631",
  "instance_id" : "c6bc8d38-e14d-4767-a5c1-d55db63a2fb4"
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 创建自定义obs桶的企业实例

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRequest request = new CreateInstanceRequest();
        CreateInstanceRequestBody body = new CreateInstanceRequestBody();
        List<CreateInstanceRequestBodyResourceTags> listbodyResourceTags = new ArrayList<>();
        listbodyResourceTags.add(
            new CreateInstanceRequestBodyResourceTags()
                .withKey("test")
                .withValue("test")
        );
        body.withObsBucketName("obs_bucket1");
        body.withResourceTags(listbodyResourceTags);
        body.withEnterpriseProjectId("0");

body.withChargeMode(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("postPaid"))
```

 版权所有 © 华为云计算技术有限公司

```
;
        body.withProjectId("project-123456");
        body.withSubnetId("subnet-87654321");
        body.withVpcId("vpc-12345678");

body.withSpec(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("swr.ee.professional"
));
        body.withName("my-instance");
        request.withBody(body);
        try {
            CreateInstanceResponse response = client.createInstance(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obs桶默认算法加密的企业实例

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRequest request = new CreateInstanceRequest();
        CreateInstanceRequestBody body = new CreateInstanceRequestBody();
        List<CreateInstanceRequestBodyResourceTags> listbodyResourceTags = new ArrayList<>();
        listbodyResourceTags.add(
            new CreateInstanceRequestBodyResourceTags()
                .withKey("test")
                .withValue("test")
```

 版权所有 © 华为云计算技术有限公司

```
        );
        body.withEncryptType(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue(""));
        body.withObsEncrypt(true);
        body.withResourceTags(listbodyResourceTags);
        body.withEnterpriseProjectId("0");

body.withChargeMode(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("postPaid"))
;
        body.withProjectId("myproject-12345");
        body.withSubnetId("subnet-87654321");
        body.withVpcId("vpc-12345678");

body.withSpec(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("swr.ee.professional"
));
        body.withDescription("My test instance");
        body.withName("my-instance");
        request.withBody(body);
        try {
            CreateInstanceResponse response = client.createInstance(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obs桶国密加密的企业实例

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
```

 版权所有 © 华为云计算技术有限公司

```
                .build();
        CreateInstanceRequest request = new CreateInstanceRequest();
        CreateInstanceRequestBody body = new CreateInstanceRequestBody();
        List<CreateInstanceRequestBodyResourceTags> listbodyResourceTags = new ArrayList<>();
        listbodyResourceTags.add(
            new CreateInstanceRequestBodyResourceTags()
                .withKey("test")
                .withValue("test")
        );
        body.withEncryptType(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("gm"));
        body.withObsEncrypt(true);
        body.withResourceTags(listbodyResourceTags);
        body.withEnterpriseProjectId("0");

body.withChargeMode(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("postPaid"))
;
        body.withProjectId("myproject-12345");
        body.withSubnetId("subnet-87654321");
        body.withVpcId("vpc-12345678");

body.withSpec(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("swr.ee.professional"
));
        body.withDescription("My test instance");
        body.withName("my-instance");
        request.withBody(body);
        try {
            CreateInstanceResponse response = client.createInstance(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OBS桶时使用已有的KMS密钥进行加密

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceSolution {

    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}";
```

 版权所有 © 华为云计算技术有限公司

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRequest request = new CreateInstanceRequest();
        CreateInstanceRequestBody body = new CreateInstanceRequestBody();
        List<CreateInstanceRequestBodyResourceTags> listbodyResourceTags = new ArrayList<>();
        listbodyResourceTags.add(
            new CreateInstanceRequestBodyResourceTags()
                .withKey("test")
                .withValue("test")
        );
        body.withObsEncKmsKeyId("123456101");
        body.withObsEncrypt(true);
        body.withResourceTags(listbodyResourceTags);
        body.withEnterpriseProjectId("0");

body.withChargeMode(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("postPaid"))
;
        body.withProjectId("myproject-12345");
        body.withSubnetId("subnet-87654321");
        body.withVpcId("vpc-12345678");

body.withSpec(CreateInstanceRequestBody.CreateInstanceRequestBody.fromValue("swr.ee.professional"
));
        body.withDescription("My test instance");
        body.withName("my-instance");
        request.withBody(body);
        try {
            CreateInstanceResponse response = client.createInstance(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

- 创建自定义obs桶的企业实例

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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"]
```

 版权所有 © 华为云计算技术有限公司

```
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateInstanceRequest()
        listResourceTagsbody = [
            CreateInstanceRequestBodyResourceTags(
                key="test",
                value="test"
            )
        ]
        request.body = CreateInstanceRequestBody(
            obs_bucket_name="obs_bucket1",
            resource_tags=listResourceTagsbody,
            enterprise_project_id="0",
            charge_mode="postPaid",
            project_id="project-123456",
            subnet_id="subnet-87654321",
            vpc_id="vpc-12345678",
            spec="swr.ee.professional",
            name="my-instance"
        )
        response = client.create_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)
```

- 创建obs桶默认算法加密的企业实例

```
    # coding: utf-8

    import os
    from huaweicloudsdkcore.auth.credentials import BasicCredentials
    from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
    from huaweicloudsdkcore.exceptions import exceptions
    from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
            .with_credentials(credentials) \
            .with_region(SwrRegion.value_of("<YOUR REGION>")) \
            .build()

        try:
            request = CreateInstanceRequest()
            listResourceTagsbody = [
                CreateInstanceRequestBodyResourceTags(
                    key="test",
                    value="test"
```

```
            )
        ]
        request.body = CreateInstanceRequestBody(
            encrypt_type="",
            obs_encrypt=True,
            resource_tags=listResourceTagsbody,
            enterprise_project_id="0",
            charge_mode="postPaid",
            project_id="myproject-12345",
            subnet_id="subnet-87654321",
            vpc_id="vpc-12345678",
            spec="swr.ee.professional",
            description="My test instance",
            name="my-instance"
        )
        response = client.create_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)
```

- 创建obs桶国密加密的企业实例

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceRequest()
        listResourceTagsbody = [
            CreateInstanceRequestBodyResourceTags(
                key="test",
                value="test"
            )
        ]
        request.body = CreateInstanceRequestBody(
            encrypt_type="gm",
            obs_encrypt=True,
            resource_tags=listResourceTagsbody,
            enterprise_project_id="0",
            charge_mode="postPaid",
            project_id="myproject-12345",
            subnet_id="subnet-87654321",
            vpc_id="vpc-12345678",
            spec="swr.ee.professional",
            description="My test instance",
            name="my-instance"
```

 版权所有 © 华为云计算技术有限公司

```
        )
        response = client.create_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)
```

- 创建OBS桶时使用已有的KMS密钥进行加密

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceRequest()
        listResourceTagsbody = [
            CreateInstanceRequestBodyResourceTags(
                key="test",
                value="test"
            )
        ]
        request.body = CreateInstanceRequestBody(
            obs_enc_kms_key_id="123456101",
            obs_encrypt=True,
            resource_tags=listResourceTagsbody,
            enterprise_project_id="0",
            charge_mode="postPaid",
            project_id="myproject-12345",
            subnet_id="subnet-87654321",
            vpc_id="vpc-12345678",
            spec="swr.ee.professional",
            description="My test instance",
            name="my-instance"
        )
        response = client.create_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

- 创建自定义obs桶的企业实例

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRequest{}
    keyResourceTags:= "test"
    valueResourceTags:= "test"
    var listResourceTagsbody = []model.CreateInstanceRequestBodyResourceTags{
        {
            Key: &keyResourceTags,
            Value: &valueResourceTags,
        },
    }
    obsBucketNameCreateInstanceRequestBody:= "obs_bucket1"
    request.Body = &model.CreateInstanceRequestBody{
        ObsBucketName: &obsBucketNameCreateInstanceRequestBody,
        ResourceTags: &listResourceTagsbody,
        EnterpriseProjectId: "0",
        ChargeMode: model.GetCreateInstanceRequestBodyChargeModeEnum().POST_PAID,
        ProjectId: "project-123456",
        SubnetId: "subnet-87654321",
        VpcId: "vpc-12345678",
        Spec: model.GetCreateInstanceRequestBodySpecEnum().SWR_EE_PROFESSIONAL,
        Name: "my-instance",
    }
    response, err := client.CreateInstance(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```

 版权所有 © 华为云计算技术有限公司

```
    } else {
        fmt.Println(err)
    }
}
```

- 创建obs桶默认算法加密的企业实例

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRequest{}
    keyResourceTags:= "test"
    valueResourceTags:= "test"
    var listResourceTagsbody = []model.CreateInstanceRequestBodyResourceTags{
        {
            Key: &keyResourceTags,
            Value: &valueResourceTags,
        },
    }
    encryptTypeCreateInstanceRequestBody:= model.GetCreateInstanceRequestBodyEncryptTypeEnum().EMPTY
    obsEncryptCreateInstanceRequestBody:= true
    descriptionCreateInstanceRequestBody:= "My test instance"
    request.Body = &model.CreateInstanceRequestBody{
        EncryptType: &encryptTypeCreateInstanceRequestBody,
        ObsEncrypt: &obsEncryptCreateInstanceRequestBody,
        ResourceTags: &listResourceTagsbody,
        EnterpriseProjectId: "0",
        ChargeMode: model.GetCreateInstanceRequestBodyChargeModeEnum().POST_PAID,
```

 版权所有 © 华为云计算技术有限公司

```
        ProjectId: "myproject-12345",
        SubnetId: "subnet-87654321",
        VpcId: "vpc-12345678",
        Spec: model.GetCreateInstanceRequestBodySpecEnum().SWR_EE_PROFESSIONAL,
        Description: &descriptionCreateInstanceRequestBody,
        Name: "my-instance",
    }
    response, err := client.CreateInstance(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建obs桶国密加密的企业实例

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRequest{}
    keyResourceTags:= "test"
    valueResourceTags:= "test"
    var listResourceTagsbody = []model.CreateInstanceRequestBodyResourceTags{
        {
            Key: &keyResourceTags,
            Value: &valueResourceTags,
        },
    }
```

 版权所有 © 华为云计算技术有限公司

```
    encryptTypeCreateInstanceRequestBody:=
model.GetCreateInstanceRequestBodyEncryptTypeEnum().GM
    obsEncryptCreateInstanceRequestBody:= true
    descriptionCreateInstanceRequestBody:= "My test instance"
    request.Body = &model.CreateInstanceRequestBody{
        EncryptType: &encryptTypeCreateInstanceRequestBody,
        ObsEncrypt: &obsEncryptCreateInstanceRequestBody,
        ResourceTags: &listResourceTagsbody,
        EnterpriseProjectId: "0",
        ChargeMode: model.GetCreateInstanceRequestBodyChargeModeEnum().POST_PAID,
        ProjectId: "myproject-12345",
        SubnetId: "subnet-87654321",
        VpcId: "vpc-12345678",
        Spec: model.GetCreateInstanceRequestBodySpecEnum().SWR_EE_PROFESSIONAL,
        Description: &descriptionCreateInstanceRequestBody,
        Name: "my-instance",
    }
    response, err := client.CreateInstance(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建OBS桶时使用已有的KMS密钥进行加密

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.CreateInstanceRequest{}
    keyResourceTags:= "test"
    valueResourceTags:= "test"
    var listResourceTagsbody = []model.CreateInstanceRequestBodyResourceTags{
        {
            Key: &keyResourceTags,
            Value: &valueResourceTags,
        },
    }
    obsEncKmsKeyIdCreateInstanceRequestBody:= "123456101"
    obsEncryptCreateInstanceRequestBody:= true
    descriptionCreateInstanceRequestBody:= "My test instance"
    request.Body = &model.CreateInstanceRequestBody{
        ObsEncKmsKeyId: &obsEncKmsKeyIdCreateInstanceRequestBody,
        ObsEncrypt: &obsEncryptCreateInstanceRequestBody,
        ResourceTags: &listResourceTagsbody,
        EnterpriseProjectId: "0",
        ChargeMode: model.GetCreateInstanceRequestBodyChargeModeEnum().POST_PAID,
        ProjectId: "myproject-12345",
        SubnetId: "subnet-87654321",
        VpcId: "vpc-12345678",
        Spec: model.GetCreateInstanceRequestBodySpecEnum().SWR_EE_PROFESSIONAL,
        Description: &descriptionCreateInstanceRequestBody,
        Name: "my-instance",
    }
    response, err := client.CreateInstance(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
202	创建实例成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.1.2 查询实例列表 - ListInstance

功能介绍

查询实例列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:list	List	instance *	-	-	-

URI

GET /v2/{project_id}/instances

表 4-11 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

表 4-12 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为100，最大值为100。**注意：offset和limit参数需要配套使用。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
status	否	String	实例状态 取值范围： • Initial：初始化 • Creating：创建中 • Running：运行中 • Unavailable：不可用
enterprise_project_id	否	String	企业项目ID
with_tag	否	Boolean	是否返回注册表实例的标签
tags	否	String	根据标签过滤，多个标签以英文逗号隔开，最多支持20个标签同时过滤查询。格式为key1=value1,key2=value2。如果标签的key/value值中有特殊字符=，需要把=转义成%253D。

请求参数

表 4-13 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-14 响应 Body 参数

参数	参数类型	描述
instances	Array of **Instance** objects	实例列表
total	Integer	实例总数

版权所有 © 华为云计算技术有限公司

表 4-15 Instance

参数	参数类型	描述
id	String	实例ID
name	String	实例名称
description	String	实例描述
vpc_id	String	用户虚拟私有云ID
subnet_id	String	用户子网的网络ID
spec	String	实例规格
version	String	实例版本
charge_mode	String	实例计费模式
access_address	String	访问地址
created_at	String	实例创建时间
updated_at	String	实例更新时间
expires_at	String	实例过期时间
status	String	实例状态 取值范围： ● Initial：初始化 ● Creating：创建中 ● Running：运行中 ● Unavailable：不可用
obs_bucket_name	String	obs桶名称
project_id	String	项目ID
enterprise_project_id	String	企业项目ID
user_def_obs	Boolean	是否为用户指定obs桶
product_id	String	产品ID
order_id	String	订单ID，包周期预留字段
vpc_name	String	VPC的名称
vpc_cidr	String	VPC的可用子网的范围
subnet_name	String	VPC的子网名称
subnet_cidr	String	子网的网段
vpcep_service_id	String	实例对应的VPC终端节点服务ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
tags	Array of ResourceTag objects	标签列表

表 4-16 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-17 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-18 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-19 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-20 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-21 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances
```

响应示例

状态码：200

查询实例列表成功

```
{
  "total" : 1,
```


版权所有 © 华为云计算技术有限公司

```
 "instances" : [ {
   "id" : "c6bc8d38-e14d-4767-a5c1-d55db63a2fb4",
   "name" : "first-blood",
   "description" : "",
   "vpc_id" : "0d09f1bc-95fb-4e04-90c6-1ac60daa2c07",
   "subnet_id" : "6b638754-2b95-47d7-9f90-632c8706ece7",
   "spec" : "basic",
   "version" : "v1",
   "charge_mode" : "postPaid",
   "access_address" : "762165ca-132e-4915-b6be-9a2b9d03d19d.cer.cn-north-7.myhuaweicloud.com",
   "created_at" : "2020-11-21T08:49:30.316Z",
   "updated_at" : "2020-11-21T08:49:30.316Z",
   "expires_at" : "0001-01-01T00:00:00Z",
   "status" : "Running"
 } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRequest request = new ListInstanceRequest();
        try {
            ListInstanceResponse response = client.listInstance(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());
```

 版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRequest()
        response = client.list_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
```

 版权所有 © 华为云计算技术有限公司

```
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRequest{}
    response, err := client.ListInstance(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	查询实例列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.3 获取实例详情 - ShowInstance

功能介绍

获取实例详情

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:get	Read	instance *	• g:EnterpriseProjectId • g:ResourceTag/<tag-key>	-	-

URI

GET /v2/{project_id}/instances/{instance_id}

表 4-22 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-23 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-24 响应 Body 参数

参数	参数类型	描述
id	String	实例ID
name	String	实例名称
description	String	实例描述
vpc_id	String	用户虚拟私有云ID
subnet_id	String	用户子网的网络ID
spec	String	实例规格
version	String	实例版本
charge_mode	String	实例计费模式
access_address	String	访问地址
created_at	String	实例创建时间
updated_at	String	实例更新时间
expires_at	String	实例过期时间
status	String	实例状态 取值范围： ● Initial：初始化 ● Creating：创建中 ● Running：运行中 ● Unavailable：不可用
obs_bucket_name	String	obs桶名称
project_id	String	项目ID
enterprise_project_id	String	企业项目ID
user_def_obs	Boolean	是否为用户指定obs桶
product_id	String	产品ID
order_id	String	订单ID，包周期预留字段
vpc_name	String	VPC的名称
vpc_cidr	String	VPC的可用子网的范围
subnet_name	String	VPC的子网名称

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
subnet_cidr	String	子网的网段
vpcep_service_id	String	实例对应的VPC终端节点服务ID
tags	Array of ResourceTag objects	标签列表

表 4-25 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-26 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-27 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-28 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-29 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-30 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : "c6bc8d38-e14d-4767-a5c1-d55db63a2fb4",
```


版权所有 © 华为云计算技术有限公司

```
  "name" : "first-blood",
  "description" : "",
  "vpc_id" : "0d09f1bc-95fb-4e04-90c6-1ac60daa2c07",
  "subnet_id" : "6b638754-2b95-47d7-9f90-632c8706ece7",
  "spec" : "basic",
  "version" : "v1",
  "charge_mode" : "postPaid",
  "access_address" : "762165ca-132e-4915-b6be-9a2b9d03d19d.cer.cn-north-7.myhuaweicloud.com",
  "created_at" : "2020-11-21T08:49:30.316Z",
  "updated_at" : "2020-11-21T08:49:30.316Z",
  "expires_at" : "0001-01-01T00:00:00Z",
  "status" : "Running"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceRequest request = new ShowInstanceRequest();
        request.withInstanceId("{instance_id}");
        try {
            ShowInstanceResponse response = client.showInstance(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());
```

 版权所有 © 华为云计算技术有限公司

```
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceRequest()
        request.instance_id = "{instance_id}"
        response = client.show_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ShowInstance(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.4 删除实例 - DeleteInstance

功能介绍

删除实例

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:delete	Write	instance *	• g:EnterpriseProjectId • g:ResourceTag/<tag-key>	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}

表 4-31 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-32 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-33 请求 Body 参数

参数	是否必选	参数类型	描述
delete_obs	否	Boolean	是否删除obs桶
delete_dns	否	Boolean	是否删除DNS域名信息

响应参数

状态码：202

删除请求已接受

状态码：400

表 4-34 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-35 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-36 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-37 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-38 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}

{
  "delete_obs" : true,
  "delete_dns" : true
}
```

响应示例

无

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceRequest request = new DeleteInstanceRequest();
        request.withInstanceId("{instance_id}");
        DeleteInstanceRequestBody body = new DeleteInstanceRequestBody();
        body.withDeleteDns(true);
        body.withDeleteObs(true);
        request.withBody(body);
        try {
            DeleteInstanceResponse response = client.deleteInstance(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
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceRequest()
        request.instance_id = "{instance_id}"
        request.body = DeleteInstanceRequestBody(
            delete_dns=True,
            delete_obs=True
        )
        response = client.delete_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceRequest{}
    request.InstanceId = "{instance_id}"
    deleteDnsDeleteInstanceRequestBody:= true
    deleteObsDeleteInstanceRequestBody:= true
    request.Body = &model.DeleteInstanceRequestBody{
        DeleteDns: &deleteDnsDeleteInstanceRequestBody,
        DeleteObs: &deleteObsDeleteInstanceRequestBody,
    }
    response, err := client.DeleteInstance(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
202	删除请求已接受
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.5 查看实例配置 - ShowInstanceConfiguration

功能介绍

查看实例配置

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getConfigurations	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/configurations

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
instance_id	是	String	企业仓库实例ID

请求参数

表 4-40 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-41 响应 Body 参数

参数	参数类型	描述
anonymous_access	Boolean	是否开启匿名登录

状态码：400

表 4-42 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-43 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-44 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-45 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-46 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/configurations
```

响应示例

状态码：200

查看实例配置成功

```
{
  "anonymous_access" : true
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceConfigurationSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceConfigurationRequest request = new ShowInstanceConfigurationRequest();
        request.withInstanceId("{instance_id}");
        try {
            ShowInstanceConfigurationResponse response = client.showInstanceConfiguration(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceConfigurationRequest()
        request.instance_id = "{instance_id}"
        response = client.show_instance_configuration(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.ShowInstanceConfigurationRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ShowInstanceConfiguration(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	查看实例配置成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.6 更新实例配置 - UpdateInstanceConfiguration

功能介绍

更新实例配置

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateConfigurations	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/configurations

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
instance_id	是	String	企业仓库实例ID

请求参数

表 4-48 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-49 请求 Body 参数

参数	是否必选	参数类型	描述
anonymous_access	是	Boolean	是否开启匿名登录

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

更新实例配置成功

状态码：400

表 4-50 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-51 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-52 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-53 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-54 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/configurations

{
  "anonymous_access" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;
```


版权所有 © 华为云计算技术有限公司

```
public class UpdateInstanceConfigurationSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceConfigurationRequest request = new UpdateInstanceConfigurationRequest();
        request.withInstanceId("{instance_id}");
        UpdateInstanceConfigurationRequestBody body = new UpdateInstanceConfigurationRequestBody();
        body.withAnonymousAccess(true);
        request.withBody(body);
        try {
            UpdateInstanceConfigurationResponse response = client.updateInstanceConfiguration(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()
```

 版权所有 © 华为云计算技术有限公司

```
    try:
        request = UpdateInstanceConfigurationRequest()
        request.instance_id = "{instance_id}"
        request.body = UpdateInstanceConfigurationRequestBody(
            anonymous_access=True
        )
        response = client.update_instance_configuration(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceConfigurationRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.UpdateInstanceConfigurationRequestBody{
        AnonymousAccess: true,
    }
    response, err := client.UpdateInstanceConfiguration(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```

 版权所有 © 华为云计算技术有限公司

```
    } 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	更新实例配置成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.7 获取上传下载的相关审计日志列表 - ListAuditLogs

功能介绍

获取上传下载的相关审计日志列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getAuditLogs	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/audit-logs

表 4-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-56 Query 参数

参数	是否必选	参数类型	描述
operation	是	String	类型，目前支持pull,delete,create
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-57 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-58 响应 Body 参数

参数	参数类型	描述
audit_logs	Array of **AuditLog** objects	审计日志列表
total	Integer	审计日志总数

表 4-59 AuditLog

参数	参数类型	描述
id	Long	Audit log ID
operation	String	操作(e.g., create, update, delete)
resource_type	String	资源类型 取值范围： • artifact：镜像制品 • repository：镜像仓库 • tag：镜像版本 • project：命名空间
resource	String	资源名称
username	String	用户ID
op_time	String	操作时间

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-60 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-61 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-62 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-63 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-64 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/audit-logs
```

响应示例

状态码：200

查询成功

```
{
  "audit_logs" : [ {
    "id" : 138,
    "op_time" : "2020-12-24T11:19:54.907Z",
    "operation" : "pull",
    "resource" : "library/
ubuntu:sha256:22eb792e7202aee4895dc0ad09126c310161e4ca40d3cec70c32505cffee7f06",
    "resource_type" : "artifact",
    "username" : "robot$Clair-f279f0f6-45d9-11eb-b153-02550a00004a"
  }, {
    "id" : 137,
    "op_time" : "2020-12-24T11:17:32.815Z",
    "operation" : "pull",
    "resource" : "library/
ubuntu:sha256:22eb792e7202aee4895dc0ad09126c310161e4ca40d3cec70c32505cffee7f06",
    "resource_type" : "artifact",
    "username" : "robot$Clair-9dabde3d-45d9-11eb-83a4-02550a000048"
  }, {
    "id" : 136,
    "op_time" : "2020-12-24T10:50:26.372Z",
    "operation" : "pull",
    "resource" : "library/
ubuntu:sha256:22eb792e7202aee4895dc0ad09126c310161e4ca40d3cec70c32505cffee7f06",
    "resource_type" : "artifact",
    "username" : "robot$Clair-d4613172-45d5-11eb-a8bc-02550a000045"
  } ],
  "total" : 3
}
```


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListAuditLogsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAuditLogsRequest request = new ListAuditLogsRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListAuditLogsResponse response = client.listAuditLogs(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 import *
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAuditLogsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_audit_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListAuditLogsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListAuditLogs(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.1.8 获取实例统计数据 - ListInstanceStatistics

功能介绍

获取实例统计数据

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getStatistics	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/statistics

表 4-65 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-66 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-67 响应 Body 参数

参数	参数类型	描述
storage_used	Long	已用存储空间
total_namespace_count	Long	命名空间的总数
total_image_count	Long	镜像的总数
namespace_quota	Long	命名空间的配额
image_repo_quota	Long	镜像的配额
replica_policy_quota	Long	镜像同步策略的配额
retention_policy_quota	Long	镜像老化策略的配额
notify_policy_quota	Long	触发器的配额
replica_registry_quota	Long	镜像同步的目标仓库配额
sign_policy_quota	Long	镜像签名策略的配额
replica_policy_count	Long	镜像同步策略总数
retention_policy_count	Long	镜像老化策略的总数
notify_policy_count	Long	触发器的总数
replica_registry_count	Long	镜像同步策略的目标仓库总数
intranet_endpoint_quota	Integer	内网访问控制的配额
intranet_endpoint_count	Integer	内网访问控制的总数
long_term_quota	Integer	长期登录指令的配额
sign_policy_count	Long	镜像签名策略的总数

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-68 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-69 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-70 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-71 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-72 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/statistics
```

响应示例

状态码：200

查询成功

```
{
  "storage_used" : 1073741824,
  "total_namespace_count" : 10,
  "total_image_count" : 100,
  "namespace_quota" : 10000,
  "image_repo_quota" : 5000,
  "replica_policy_quota" : 20,
  "retention_policy_quota" : 10,
  "notify_policy_quota" : 5,
  "replica_registry_quota" : 2,
  "sign_policy_quota" : 10,
  "replica_policy_count" : 10,
  "retention_policy_count" : 5,
  "notify_policy_count" : 2,
  "replica_registry_count" : 1,
  "intranet_endpoint_quota" : 100,
  "intranet_endpoint_count" : 50,
  "long_term_quota" : 500,
  "sign_policy_count" : 5
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceStatisticsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceStatisticsRequest request = new ListInstanceStatisticsRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceStatisticsResponse response = client.listInstanceStatistics(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceStatisticsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_statistics(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.ListInstanceStatisticsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceStatistics(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.2 命名空间管理

4.2.1 创建命名空间 - CreateInstanceNamespace

功能介绍

创建命名空间

接口约束

无

调用方法

请参见**如何调用API**。

版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:createNamespace	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-
		-	swr:EnablePublicNameSpace		

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces

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
instance_id	是	String	企业仓库实例ID

请求参数

表 4-74 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-75 请求 Body 参数

参数	是否必选	参数类型	描述
namespace_name	是	String	命名空间名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。
metadata	是	NamespaceMetadata object	元数据
tags	否	Array of ResourceTag objects	资源标签, 最多支持20个标签

表 4-76 NamespaceMetadata

参数	是否必选	参数类型	描述
public	是	String	是否公开，可选true、false
auto_scan	否	String	上传制品时，是否自动触发制品扫描，可选true、false
prevent_vul	否	String	是否开启制品阻断，可选true、false
severity	否	String	阻断开启的场景下，如果存在漏洞，并且存在的漏洞严重程度高于或者等于此处定义的级别，则无法拉取镜像。可选值为"none", "low", "medium", "high", "critical"。

表 4-77 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。
value	是	String	标签值，每个值最大长度255个unicode字符。

响应参数

状态码：201

版权所有 © 华为云计算技术有限公司

表 4-78 响应 Body 参数

参数	参数类型	描述
id	Integer	命名空间ID

状态码：400

表 4-79 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-80 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-81 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-82 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-83 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces

{
  "namespace_name" : "test",
  "metadata" : {
    "public" : "false"
  }
}
```

响应示例

状态码：201

创建命名空间成功

```
{
  "id" : 1
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceNamespaceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceNamespaceRequest request = new CreateInstanceNamespaceRequest();
        request.withInstanceId("{instance_id}");
        CreateInstanceNamespaceRequestBody body = new CreateInstanceNamespaceRequestBody();
        NamespaceMetadata metadatabody = new NamespaceMetadata();
        metadatabody.withPublic(NamespaceMetadata.PublicEnum.fromValue("false"));
        body.withMetadata(metadatabody);
        body.withNamespaceName("test");
        request.withBody(body);
        try {
            CreateInstanceNamespaceResponse response = client.createInstanceNamespace(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 import *
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceNamespaceRequest()
        request.instance_id = "{instance_id}"
        metadatabody = NamespaceMetadata(
            public="false"
        )
        request.body = CreateInstanceNamespaceRequestBody(
            metadata=metadatabody,
            namespace_name="test"
        )
        response = client.create_instance_namespace(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceNamespaceRequest{}
    request.InstanceId = "{instance_id}"
    metadatabody := &model.NamespaceMetadata{
        Public: model.GetNamespaceMetadataPublicEnum().FALSE,
    }
    request.Body = &model.CreateInstanceNamespaceRequestBody{
        Metadata: metadatabody,
        NamespaceName: "test",
    }
    response, err := client.CreateInstanceNamespace(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	创建命名空间成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.2.2 获取命名空间列表 - ListInstanceNamespaces

功能介绍

获取命名空间列表

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listNamespaces	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces

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
instance_id	是	String	企业仓库实例ID

表 4-85 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
order_column	否	String	按列排序，可设置为updated_at（按更新时间排序）
order_type	否	String	排序方式，与order_column配合使用 取值范围： ● desc：降序 ● asc：升序
name	否	String	命名空间名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。
public	否	String	是否公开，非空且非true默认返回false
with_tag	否	Boolean	是否返回命名空间的标签
tags	否	String	根据标签过滤，多个标签以英文逗号隔开，最多支持20个标签同时过滤查询。格式为key1=value1,key2=value2。如果标签的key/value值中有特殊字符=，请先把 = 替换成 %3D 再进行请求。注意：使用curl命令请求，需要把 = 替换成 %253D(即 %3D 的 % 被编码为 %25)再次请求。

请求参数

表 4-86 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-87 响应 Body 参数

参数	参数类型	描述
namespaces	Array of **Namespace** objects	命名空间列表
total	Integer	命名空间总数

表 4-88 Namespace

参数	参数类型	描述
name	String	命名空间名称
metadata	**NamespaceMetadata** object	元数据
namespace_id	Integer	命名空间ID
created_at	String	创建时间
updated_at	String	更新时间
repo_count	Integer	镜像数量
cve_allowlist	**CVEAllowlist** object	漏洞白名单
tags	Array of **ResourceTag** objects	标签列表

表 4-89 NamespaceMetadata

参数	参数类型	描述
public	String	是否公开，可选true、false
auto_scan	String	上传制品时，是否自动触发制品扫描，可选true、false
prevent_vul	String	是否开启制品阻断，可选true、false

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
severity	String	阻断开启的场景下，如果存在漏洞，并且存在的漏洞严重程度高于或者等于此处定义的级别，则无法拉取镜像。可选值为"none", "low", "medium", "high", "critical"。

表 4-90 CVEAllowlist

参数	参数类型	描述
id	Integer	白名单ID
namespace_id	Integer	漏洞白名单列表所属的命名空间ID
expires_at	Integer	漏洞白名单的有效期时间，如果没有配置，则永不过期
items	Array of **CVEAllowlistItem** objects	漏洞白名单的漏洞项
creation_time	String	漏洞白名单的创建时间
update_time	String	漏洞白名单的更新时间

表 4-91 CVEAllowlistItem

参数	参数类型	描述
cve_id	String	CVE的ID, 比如：CVE-2019-10164

表 4-92 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-93 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-94 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-95 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-96 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-97 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces
```

响应示例

状态码：200

查询命名空间列表成功

```
{
  "total" : 18,
  "namespaces" : [ {
    "metadata" : {
      "public" : "false"
    },
    "name" : "testabc",
    "namespace_id" : 142,
    "repo_count" : 0,
    "created_at" : "2021-02-19T08:23:28.519Z",
    "updated_at" : "2021-02-19T08:23:28.519Z"
  } ]
}
```

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;
```


版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceNamespacesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceNamespacesRequest request = new ListInstanceNamespacesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceNamespacesResponse response = client.listInstanceNamespaces(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
```

 版权所有 © 华为云计算技术有限公司

```
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceNamespacesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_namespaces(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

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

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

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	查询命名空间列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.2.3 获取命名空间详情 - ShowInstanceNamespace

功能介绍

获取命名空间详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getNamespace	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}

表 4-98 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。

请求参数

表 4-99 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-100 响应 Body 参数

参数	参数类型	描述
name	String	命名空间名称
metadata	**NamespaceMetadata** object	元数据
namespace_id	Integer	命名空间ID
created_at	String	创建时间
updated_at	String	更新时间
repo_count	Integer	镜像数量
cve_allowlist	**CVEAllowlist** object	漏洞白名单
tags	Array of **ResourceTag** objects	标签列表

表 4-101 NamespaceMetadata

参数	参数类型	描述
public	String	是否公开，可选true、false
auto_scan	String	上传制品时，是否自动触发制品扫描，可选true、false
prevent_vul	String	是否开启制品阻断，可选true、false
severity	String	阻断开启的场景下，如果存在漏洞，并且存在的漏洞严重程度高于或者等于此处定义的级别，则无法拉取镜像。可选值为"none", "low", "medium", "high", "critical"。

表 4-102 CVEAllowlist

参数	参数类型	描述
id	Integer	白名单ID
namespace_id	Integer	漏洞白名单列表所属的命名空间ID
expires_at	Integer	漏洞白名单的有效期时间，如果没有配置，则永不过期

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
items	Array of CVEAllowlistItem objects	漏洞白名单的漏洞项
creation_time	String	漏洞白名单的创建时间
update_time	String	漏洞白名单的更新时间

表 4-103 CVEAllowlistItem

参数	参数类型	描述
cve_id	String	CVE的ID, 比如：CVE-2019-10164

表 4-104 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-105 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-106 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-107 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-108 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-109 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}
```

响应示例

状态码：200

查询成功

```
{
  "metadata" : {
    "public" : "true"
  },
  "name" : "library",
  "repo_count" : 12,
  "namespace_id" : 1,
  "created_at" : "2020-11-19T02:58:54.809Z",
  "updated_at" : "2020-11-19T02:58:54.809Z"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceNamespaceSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceNamespaceRequest request = new ShowInstanceNamespaceRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        try {
            ShowInstanceNamespaceResponse response = client.showInstanceNamespace(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceNamespaceRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        response = client.show_instance_namespace(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceNamespaceRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    response, err := client.ShowInstanceNamespace(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.2.4 修改命名空间 - UpdateInstanceNamespace

功能介绍

修改命名空间

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateNamespace	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-
		-	swr:EnablePublicNameSpace		

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}

表 4-110 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。

请求参数

表 4-111 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-112 请求 Body 参数

参数	是否必选	参数类型	描述
metadata	是	**NamespaceMetadata** object	元数据
cve_allowlist	否	**UpdateCVEAllowlistRequest** object	漏洞白名单

表 4-113 NamespaceMetadata

参数	是否必选	参数类型	描述
public	是	String	是否公开，可选true、false
auto_scan	否	String	上传制品时，是否自动触发制品扫描，可选true、false
prevent_vul	否	String	是否开启制品阻断，可选true、false

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
severity	否	String	阻断开启的场景下，如果存在漏洞，并且存在的漏洞严重程度高于或者等于此处定义的级别，则无法拉取镜像。可选值为 "none", "low", "medium", "high", "critical"。

表 4-114 UpdateCVEAllowlistRequest

参数	是否必选	参数类型	描述
id	否	Integer	白名单ID，查询命名空间信息时，会返回白名单ID
namespace_id	否	Integer	漏洞白名单列表所属的命名空间ID
expires_at	否	Integer	漏洞白名单的有效期时间，以自 1970 年 1 月 1 日以来的秒数表示；如果没有配置，则永不过期
items	否	Array of **CVEAllowlistItem** objects	漏洞列表，最多支持100条数据，且不能有重复漏洞。

表 4-115 CVEAllowlistItem

参数	是否必选	参数类型	描述
cve_id	否	String	CVE的ID, 比如：CVE-2019-10164

响应参数

状态码：200

表 4-116 响应 Body 参数

参数	参数类型	描述
namespace_name	String	命名空间名称
metadata	**NamespaceMetadata** object	元数据
cve_allowlist	**UpdateCVEAllowlistRequest** object	漏洞白名单

版权所有 © 华为云计算技术有限公司

表 4-117 NamespaceMetadata

参数	参数类型	描述
public	String	是否公开，可选true、false
auto_scan	String	上传制品时，是否自动触发制品扫描，可选true、false
prevent_vul	String	是否开启制品阻断，可选true、false
severity	String	阻断开启的场景下，如果存在漏洞，并且存在的漏洞严重程度高于或者等于此处定义的级别，则无法拉取镜像。可选值为"none", "low", "medium", "high", "critical"。

表 4-118 UpdateCVEAllowlistRequest

参数	参数类型	描述
id	Integer	白名单ID，查询命名空间信息时，会返回白名单ID
namespace_id	Integer	漏洞白名单列表所属的命名空间ID
expires_at	Integer	漏洞白名单的有效期时间，以自 1970 年 1 月 1 日以来的秒数表示；如果没有配置，则永不过期
items	Array of **CVEAllowlistItem** objects	漏洞列表，最多支持100条数据，且不能有重复漏洞。

表 4-119 CVEAllowlistItem

参数	参数类型	描述
cve_id	String	CVE的ID, 比如：CVE-2019-10164

状态码：400

表 4-120 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-121 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-122 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-123 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-124 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}

{
  "metadata" : {
    "public" : "true"
  }
}
```

响应示例

状态码：200

成功修改命名空间

```
{
  "metadata" : {
    "public" : "true"
  },
  "namespace_name" : "library"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateInstanceNamespaceSolution {

    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.
```

 版权所有 © 华为云计算技术有限公司

```
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceNamespaceRequest request = new UpdateInstanceNamespaceRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        UpdateNamespaceRequestBody body = new UpdateNamespaceRequestBody();
        NamespaceMetadata metadatabody = new NamespaceMetadata();
        metadatabody.withPublic(NamespaceMetadata.PublicEnum.fromValue("true"));
        body.withMetadata(metadatabody);
        request.withBody(body);
        try {
            UpdateInstanceNamespaceResponse response = client.updateInstanceNamespace(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceNamespaceRequest()
```

 版权所有 © 华为云计算技术有限公司

```
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        metadatabody = NamespaceMetadata(
            public="true"
        )
        request.body = UpdateNamespaceRequestBody(
            metadata=metadatabody
        )
        response = client.update_instance_namespace(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceNamespaceRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    metadatabody := &model.NamespaceMetadata{
        Public: model.GetNamespaceMetadataPublicEnum().TRUE,
    }
    request.Body = &model.UpdateNamespaceRequestBody{
```

 版权所有 © 华为云计算技术有限公司

```
        Metadata: metadatabody,
    }
    response, err := client.UpdateInstanceNamespace(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	成功修改命名空间
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.2.5 删除命名空间 - DeleteInstanceNamespace

功能介绍

删除命名空间

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

 版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteNamespace	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}

表 4-125 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。

请求参数

表 4-126 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-127 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-128 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-129 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-130 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-131 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceNamespaceSolution {

    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
```


版权所有 © 华为云计算技术有限公司

```
        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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceNamespaceRequest request = new DeleteInstanceNamespaceRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        try {
            DeleteInstanceNamespaceResponse response = client.deleteInstanceNamespace(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceNamespaceRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        response = client.delete_instance_namespace(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceNamespaceRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    response, err := client.DeleteInstanceNamespace(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3 制品仓库管理

4.3.1 获取制品仓库列表 - ListInstanceRepositories

功能介绍

获取制品仓库列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listRepositories	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/repositories

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
instance_id	是	String	企业仓库实例ID

表 4-133 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
order_column	否	String	排序字段，支持created_at、updated_at，默认为created_at
order_type	否	String	排序方式，默认desc 取值范围： • desc：降序 • asc：升序
namespace_id	否	Integer	所属命名空间ID
name	否	String	根据仓库名称或命名空间名称进行模糊查询

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
with_tag	否	Boolean	是否返回仓库的标签
tags	否	String	根据标签过滤，多个标签以英文逗号隔开，最多支持20个标签同时过滤查询。格式为key1=value1,key2=value2。如果标签的key/value值中有特殊字符=，请先把 = 替换成 %3D 再进行请求。注意：使用curl命令请求，需要把 = 替换成 %253D(即 %3D 的 % 被编码为 %25)再次请求。
namespace_ids	否	String	支持按多个所属命名空间ID过滤，多个ID之间用英文逗号分隔。注意：namespace_id与namespace_ids不支持同时传参。

请求参数

表 4-134 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-135 响应 Body 参数

参数	参数类型	描述
total	Integer	仓库总数
repositories	Array of **Repository** objects	仓库列表详情

版权所有 © 华为云计算技术有限公司

表 4-136 Repository

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
namespace_name	String	命名空间名称
namespace_id	Integer	命名空间ID
tag_count	Integer	仓库内的制品版本数量
pull_count	Integer	被下载总次数
artifact_count	Integer	制品包总数
description	String	描述
created_at	String	创建时间
updated_at	String	更新时间
tags	Array of **ResourceTag** objects	标签列表

表 4-137 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-138 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-139 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-140 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-141 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-142 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/repositories
```

响应示例

状态码：200

查询制品仓库列表成功

```
{
  "repositories" : [ {
    "tag_count" : 1,
    "created_at" : "2020-09-19T01:39:58.326Z",
    "description" : "test",
    "id" : 158,
    "name" : "ccccc/test",
    "namespace_id" : 36,
    "pull_count" : 12,
    "artifact_count" : 12,
    "updated_at" : "2020-10-27T07:28:04.051Z"
  } ],
  "total" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRepositoriesSolution {

    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()
```

 版权所有 © 华为云计算技术有限公司

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

    SwrClient client = SwrClient.newBuilder()
            .withCredential(auth)
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
    ListInstanceRepositoriesRequest request = new ListInstanceRepositoriesRequest();
    request.withInstanceId("{instance_id}");
    try {
        ListInstanceRepositoriesResponse response = client.listInstanceRepositories(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRepositoriesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_repositories(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
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRepositoriesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceRepositories(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	查询制品仓库列表成功
400	错误的请求

 版权所有 © 华为云计算技术有限公司

状态码	描述
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3.2 获取命名空间下所有制品仓库列表 - ListNamespaceRepositories

功能介绍

获取命名空间下所有制品仓库列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories

表 4-143 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
namespace_name	是	String	所属命名空间名称

表 4-144 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
order_column	否	String	排序字段，支持created_at、updated_at，默认为created_at
order_type	否	String	排序方式，与order_column配合使用，默认desc 取值范围： ● desc：降序 ● asc：升序

请求参数

表 4-145 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-146 响应 Body 参数

参数	参数类型	描述
total	Integer	仓库总数
repositories	Array of **Repository** objects	仓库列表详情

表 4-147 Repository

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
namespace_name	String	命名空间名称
namespace_id	Integer	命名空间ID
tag_count	Integer	仓库内的制品版本数量
pull_count	Integer	被下载总次数
artifact_count	Integer	制品包总数
description	String	描述
created_at	String	创建时间
updated_at	String	更新时间
tags	Array of **ResourceTag** objects	标签列表

表 4-148 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-149 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-150 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-151 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-152 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-153 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories
```

响应示例

状态码：200

查询制品仓库列表成功

```
{
  "repositories" : [ {
    "tag_count" : 1,
    "created_at" : "2020-09-19T01:39:58.326Z",
    "description" : "test",
    "id" : 158,
    "name" : "ccccc/test",
    "namespace_id" : 36,
    "namespace_name" : "test-namespace",
    "pull_count" : 12,
    "artifact_count" : 12,
    "updated_at" : "2020-10-27T07:28:04.051Z"
  } ],
  "total" : 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;
```

 版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListNamespaceRepositoriesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListNamespaceRepositoriesRequest request = new ListNamespaceRepositoriesRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        try {
            ListNamespaceRepositoriesResponse response = client.listNamespaceRepositories(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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}"
```

 版权所有 © 华为云计算技术有限公司

```
    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListNamespaceRepositoriesRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        response = client.list_namespace_repositories(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListNamespaceRepositoriesRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    response, err := client.ListNamespaceRepositories(request)
```

 版权所有 © 华为云计算技术有限公司

```
    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	查询制品仓库列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3.3 获取制品仓库详情 - ShowInstanceRepository

功能介绍

获取制品仓库详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

 版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getRepository	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}

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
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。

请求参数

表 4-155 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-156 响应 Body 参数

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
namespace_name	String	命名空间名称
namespace_id	Integer	命名空间ID
tag_count	Integer	仓库内的制品版本数量
pull_count	Integer	被下载总次数
artifact_count	Integer	制品包总数
description	String	描述
created_at	String	创建时间
updated_at	String	更新时间
tags	Array of **ResourceTag** objects	标签列表

表 4-157 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-158 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-159 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-160 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-161 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-162 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}
```

响应示例

状态码：200

获取制品仓库详情成功

```
{
  "tag_count" : 1,
  "description" : "test",
  "id" : 158,
  "namespace_name" : "ccccc",
  "name" : "test",
  "namespace_id" : 36,
  "pull_count" : 12,
  "artifact_count" : 12
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceRepositorySolution {

    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
```

 版权所有 © 华为云计算技术有限公司

```
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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceRepositoryRequest request = new ShowInstanceRepositoryRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        try {
            ShowInstanceRepositoryResponse response = client.showInstanceRepository(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceRepositoryRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        response = client.show_instance_repository(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceRepositoryRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    response, err := client.ShowInstanceRepository(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	获取制品仓库详情成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3.4 修改制品仓库 - UpdateInstanceRepository

功能介绍

修改制品仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateRepository	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}

表 4-163 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。

请求参数

表 4-164 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-165 请求 Body 参数

参数	是否必选	参数类型	描述
description	是	String	制品仓库描述

响应参数

状态码：200

修改成功

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-166 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-167 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-168 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-169 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-170 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}

{
  "description" : "description"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateInstanceRepositorySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceRepositoryRequest request = new UpdateInstanceRepositoryRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        UpdateRepoConfigRequestBody body = new UpdateRepoConfigRequestBody();
        body.withDescription("description");
        request.withBody(body);
        try {
            UpdateInstanceRepositoryResponse response = client.updateInstanceRepository(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceRepositoryRequest()
```

 版权所有 © 华为云计算技术有限公司

```
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.body = UpdateRepoConfigRequestBody(
            description="description"
        )
        response = client.update_instance_repository(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceRepositoryRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Body = &model.UpdateRepoConfigRequestBody{
        Description: "description",
    }
    response, err := client.UpdateInstanceRepository(request)
    if err == nil {
```

 版权所有 © 华为云计算技术有限公司

```
        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	修改成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3.5 删除制品仓库 - DeleteInstanceRepository

功能介绍

删除仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

 版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteRepository	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}

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
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。

请求参数

表 4-172 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：204

删除成功

状态码：400

表 4-173 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-174 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-175 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-176 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-177 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
repositories/{repository_name}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceRepositorySolution {

    public static void main(String[] args) {
```

 版权所有 © 华为云计算技术有限公司

```
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceRepositoryRequest request = new DeleteInstanceRepositoryRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        try {
            DeleteInstanceRepositoryResponse response = client.deleteInstanceRepository(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceRepositoryRequest()
        request.instance_id = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        response = client.delete_instance_repository(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceRepositoryRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    response, err := client.DeleteInstanceRepository(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.3.6 获取当前项目下所有企业仓库实例的仓库列表 - ListAllInstanceRepositories

功能介绍

获取当前项目下所有企业仓库实例的仓库列表，不包含共享企业仓库实例下的仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listAllInstanceRepositories	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/repositories

表 4-178 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

表 4-179 Query 参数

参数	是否必选	参数类型	描述
marker	否	String	分页查询时的查询标记，使用上一次接口调用返回的next_marker值，默认值从第一条数据查询。**注意：marker和limit参数需要配套使用。**
limit	否	Integer	条目数量，用于分页查询，默认值为100，最大值为100
name	否	String	仓库名称

请求参数

表 4-180 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-181 响应 Body 参数

参数	参数类型	描述
page_info	**PageInfo** object	分页信息
repositories	Array of **InstanceRepository** objects	仓库列表详情

表 4-182 PageInfo

参数	参数类型	描述
next_marker	String	下一次分页查询的起始标记。如果未返回该值，则表示数据已查询完毕

表 4-183 InstanceRepository

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
namespace_name	String	命名空间名称
namespace_id	Integer	命名空间ID
tag_count	Integer	仓库内的制品版本数量
pull_count	Integer	被下载总次数
artifact_count	Integer	制品包总数
description	String	描述
created_at	String	创建时间
updated_at	String	更新时间
tags	Array of **ResourceTag** objects	标签列表
instance_id	String	企业仓库实例ID
instance_name	String	企业仓库实例名称

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
resource_urn	String	资源URN的值，格式为：swr:::repository:/

表 4-184 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

请求示例

```
GET https://{endpoint}/v2/{project_id}/repositories
```

响应示例

状态码：200

查询仓库列表成功

```
{
  "repositories" : [ {
    "tag_count" : 1,
    "created_at" : "2020-09-19T01:39:58.326Z",
    "description" : "test",
    "id" : 158,
    "name" : "ccccc/test",
    "namespace_id" : 36,
    "pull_count" : 12,
    "artifact_count" : 12,
    "updated_at" : "2020-10-27T07:28:04.051Z",
    "instance_id" : "xx-xx-xx-xx",
    "instance_name" : "test",
    "resource_urn" : "swr:cn-north-4:xxxx:repository:test-swr/swr-test-1215/test2"
  } ],
  "page_info" : {
    "next_marker" : "xxx"
  }
}
```

状态码

状态码	描述
200	查询仓库列表成功

错误码

请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.4 制品版本管理

4.4.1 获取制品版本列表 - ListInstanceArtifacts

功能介绍

获取制品版本列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listArtifacts	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts

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
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。

表 4-186 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
type	否	String	制品类型 取值范围： ● IMAGE：镜像制品
tags	否	String	制品包含版本，模糊匹配条件

请求参数

表 4-187 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-188 响应 Body 参数

参数	参数类型	描述
artifacts	Array of **Artifact** objects	制品列表
total	Integer	制品总数

表 4-189 Artifact

参数	参数类型	描述
digest	String	制品摘要
id	Integer	制品ID
repository_id	Integer	仓库ID
repository_name	String	仓库名称
type	String	制品类型 取值范围： ● IMAGE：镜像制品
namespace_id	Integer	命名空间ID
media_type	String	制品MIME类型
manifest_media_type	String	制品元数据MIME类型
pull_time	String	最近一次拉取时间
push_time	String	最近一次上传时间
size	Integer	制品大小，单位：Byte
tags	Array of **ArtifactTag** objects	制品版本的Tag列表
extra_attrs	Map<String,Object>	制品的一些额外属性，比如制品的构建时间(created)、支持的操作系统(os)、支持CPU架构(architecture)等信息。
references	Array of **Reference** objects	当制品为多架构镜像索引时，显示其引用的所有子制品。

版权所有 © 华为云计算技术有限公司

表 4-190 ArtifactTag

参数	参数类型	描述
id	Long	Tag ID
repository_id	Long	制品仓库ID
artifact_id	Long	制品版本ID
name	String	tag名称
push_time	String	tag的上传时间
pull_time	String	tag的下载时间

表 4-191 Reference

参数	参数类型	描述
parent_id	Long	父制品的ID。
child_id	Long	子制品的ID。
child_digest	String	子制品的digest。
platform	**Platform** object	子制品的平台信息。

表 4-192 Platform

参数	参数类型	描述
architecture	String	制品适用的CPU架构。
os	String	制品适用的操作系统。
os.version	String	制品适用的操作系统版本。
os.features	Array of strings	制品适用的操作系统特性。
variant	String	同一系列 CPU 下的细分型号，例如 Intel i5-13400、i5-13600 属于同代不同 variant。

状态码：400

表 4-193 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-194 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-195 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-196 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-197 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts
```

响应示例

状态码：200

查询制品版本列表成功

```
{
  "total" : 1,
  "artifacts" : [ {
    "digest" : "sha256:88c944609af7a85c8c2a822a5fc50702570d126f9883c0831f5122849f5cebf4",
    "id" : 242,
    "manifest_media_type" : "application/vnd.docker.distribution.manifest.v2+json",
    "media_type" : "application/vnd.docker.container.image.v1+json",
    "namespace_id" : 7,
    "pull_time" : "2025-08-27T12:30:01.396Z",
    "push_time" : "2025-07-23T08:27:47.327Z",
    "repository_id" : 37,
    "repository_name" : "hwofficial/vault",
    "size" : 38937129,
    "tags" : [ {
      "artifact_id" : 242,
      "id" : 294,
      "name" : "1.0.3",
      "pull_time" : "2025-08-19T09:29:05.492Z",
      "push_time" : "2025-07-23T08:27:47.489Z",
      "repository_id" : 37
    } ],
    "type" : "IMAGE"
  } ]
}
```

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;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceArtifactsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceArtifactsRequest request = new ListInstanceArtifactsRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        try {
            ListInstanceArtifactsResponse response = client.listInstanceArtifacts(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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}"
```

 版权所有 © 华为云计算技术有限公司

```
    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListInstanceArtifactsRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        response = client.list_instance_artifacts(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceArtifactsRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
```

 版权所有 © 华为云计算技术有限公司

```
    request.RepositoryName = "{repository_name}"
    response, err := client.ListInstanceArtifacts(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	查询制品版本列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.4.2 获取制品版本详情 - ShowInstanceArtifact

功能介绍

获取制品版本详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。

版权所有 © 华为云计算技术有限公司

- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getArtifact	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}

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
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

表 4-199 Query 参数

参数	是否必选	参数类型	描述
with_scan_overview	否	Boolean	是否返回制品扫描摘要

版权所有 © 华为云计算技术有限公司

请求参数

表 4-200 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-201 响应 Body 参数

参数	参数类型	描述
digest	String	制品摘要
id	Integer	制品ID
repository_id	Integer	仓库ID
repository_name	String	仓库名称
type	String	制品类型 取值范围： • IMAGE：镜像制品
namespace_id	Integer	命名空间ID
media_type	String	制品MIME类型
manifest_media_type	String	制品元数据MIME类型
pull_time	String	最近一次拉取时间
push_time	String	最近一次上传时间
size	Integer	制品大小，单位：Byte
tags	Array of **ArtifactTag** objects	制品版本的Tag列表
extra_attrs	Map<String,Object>	制品的一些额外属性，比如制品的构建时间(created)、支持的操作系统(os)、支持CPU架构(architecture)等信息。
references	Array of **Reference** objects	当制品为多架构镜像索引时，显示其引用的所有子制品。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
scan_overview	**ScanOverview** object	制品扫描摘要

表 4-202 ArtifactTag

参数	参数类型	描述
id	Long	Tag ID
repository_id	Long	制品仓库ID
artifact_id	Long	制品版本ID
name	String	tag名称
push_time	String	tag的上传时间
pull_time	String	tag的下载时间

表 4-203 Reference

参数	参数类型	描述
parent_id	Long	父制品的ID。
child_id	Long	子制品的ID。
child_digest	String	子制品的digest。
platform	**Platform** object	子制品的平台信息。

表 4-204 Platform

参数	参数类型	描述
architecture	String	制品适用的CPU架构。
os	String	制品适用的操作系统。
os.version	String	制品适用的操作系统版本。
os.features	Array of strings	制品适用的操作系统特性。
variant	String	同一系列 CPU 下的细分型号，例如 Intel i5-13400、i5-13600 属于同代不同 variant。

版权所有 © 华为云计算技术有限公司

表 4-205 ScanOverview

参数	参数类型	描述
{自定义key}	Map<String,NativeReportSummary>	制品扫描摘要,支持制品扫描报告类型为application/vnd.security.vulnerability.report;version=1.1的制品扫描摘要

表 4-206 NativeReportSummary

参数	参数类型	描述
report_id	String	报告ID
scan_status	String	制品扫描状态 取值范围： • Pending：扫描等待中 • Running：扫描中 • Error：扫描失败 • Success：扫描成功
severity	String	制品扫描报告的总体严重程度，None(无评分), Low(低危), Medium(中危), High(高危), Critical(严重), Security(安全)
duration	Integer	生成报告所花费的秒数时间
summary	VulnerabilitySummary object	漏洞信息摘要
start_time	String	报告生成的开始时间
end_time	String	报告生成的结束时间
complete_percent	Integer	扫描的完成百分比，该值介于0和100之间
scanner	Scanner object	扫描器的信息

表 4-207 VulnerabilitySummary

参数	参数类型	描述
total	Integer	发现的漏洞总数
fixable	Integer	可修复漏洞的数量
summary	Map<String,Integer>	不同级别的漏洞数量

版权所有 © 华为云计算技术有限公司

表 4-208 Scanner

参数	参数类型	描述
name	String	扫描器的名称
vendor	String	扫描器的提供商
version	String	扫描器的版本号

状态码：400

表 4-209 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-210 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-211 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-212 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-213 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}
```

响应示例

状态码：200

查询制品版本详情成功

```
{
  "digest" : "sha256:d086845138397686ea70e11f069031370534dcc1617d02ae9b501504f91af50a",
  "id" : 448,
  "manifest_media_type" : "application/vnd.docker.distribution.manifest.v2+json",
  "media_type" : "application/vnd.docker.container.image.v1+json",
  "pull_time" : "2022-11-12T00:53:48.696Z",
  "push_time" : "2022-11-03T10:20:23.753Z",
```


版权所有 © 华为云计算技术有限公司

```
  "repository_id" : 49,
  "size" : 127999644,
  "tags" : [ {
    "artifact_id" : 448,
    "id" : 352,
    "name" : "v1.0.3.test1",
    "pull_time" : "2022-11-12T00:53:48.696Z",
    "push_time" : "2022-11-03T10:20:23.870Z",
    "repository_id" : 49
  } ],
  "type" : "IMAGE",
  "repository_name" : "namespace/repo-name",
  "namespace_id" : 1,
  "scan_overview" : {
    "application/vnd.security.vulnerability.report; version=1.1" : {
      "report_id" : "12345",
      "scan_status" : "Success",
      "severity" : "High",
      "duration" : 10,
      "summary" : {
        "total" : 5,
        "fixable" : 3
      }
    }
  }
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceArtifactSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceArtifactRequest request = new ShowInstanceArtifactRequest();
        request.withInstanceId("{instance_id}");
```

 版权所有 © 华为云计算技术有限公司

```
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        request.withReference("{reference}");
        try {
            ShowInstanceArtifactResponse response = client.showInstanceArtifact(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceArtifactRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        response = client.show_instance_artifact(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
```

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceArtifactRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    response, err := client.ShowInstanceArtifact(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	查询制品版本详情成功
400	错误的请求
401	鉴权失败

 版权所有 © 华为云计算技术有限公司

状态码	描述
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.4.3 删除制品版本 - DeleteInstanceArtifact

功能介绍

删除制品版本

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteArtifact	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}

版权所有 © 华为云计算技术有限公司

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
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

请求参数

表 4-215 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

成功删除制品版本

状态码：400

表 4-216 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-217 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-218 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-219 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-220 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
repositories/{repository_name}/artifacts/{reference}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceArtifactSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceArtifactRequest request = new DeleteInstanceArtifactRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        request.withReference("{reference}");
        try {
```

 版权所有 © 华为云计算技术有限公司

```
            DeleteInstanceArtifactResponse response = client.deleteInstanceArtifact(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceArtifactRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        response = client.delete_instance_artifact(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceArtifactRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    response, err := client.DeleteInstanceArtifact(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	成功删除制品版本
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源

 版权所有 © 华为云计算技术有限公司

状态码	描述
500	内部错误

错误码

请参见**错误码**。

4.4.4 获取制品附件列表 - ListInstanceAccessories

功能介绍

获取制品附件列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listAccessories	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/accessories

版权所有 © 华为云计算技术有限公司

表 4-221 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

表 4-222 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
type	否	String	附件的类型 取值范围： ● signature.cosign：cosign附件

请求参数

表 4-223 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-224 响应 Body 参数

参数	参数类型	描述
accessories	Array of **Accessory** objects	制品附件列表
total	Integer	制品附件总数

表 4-225 Accessory

参数	参数类型	描述
id	Long	附件ID
artifact_id	Long	附件制品ID
subject_artifact_id	Long	附件所属的父制品ID.
size	Long	附件的大小
digest	String	附件的sha256值
type	String	附件的类型 取值范围： • signature.cosign：cosign附件
created_at	String	附件的创建时间

状态码：400

表 4-226 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-227 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-228 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-229 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-230 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/accessories
```

响应示例

状态码：200

查询制品附件列表成功

```
{
  "total" : 1,
  "accessories" : [ {
    "artifact_id" : 1545,
    "created_at" : "2022-11-12T01:23:53.522Z",
    "digest" : "sha256:41e7031f751aa3b8eaeaa94cf73af943a77bc50f26bb36043c710253a6345a8a",
    "id" : 2,
    "size" : 558,
    "subject_artifact_id" : 448,
    "type" : "signature.cosign"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceAccessoriesSolution {

    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)
```

 版权所有 © 华为云计算技术有限公司

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

    SwrClient client = SwrClient.newBuilder()
            .withCredential(auth)
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
    ListInstanceAccessoriesRequest request = new ListInstanceAccessoriesRequest();
    request.withInstanceId("{instance_id}");
    request.withNamespaceName("{namespace_name}");
    request.withRepositoryName("{repository_name}");
    request.withReference("{reference}");
    try {
        ListInstanceAccessoriesResponse response = client.listInstanceAccessories(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceAccessoriesRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        response = client.list_instance_accessories(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

 版权所有 © 华为云计算技术有限公司

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceAccessoriesRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    response, err := client.ListInstanceAccessories(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询制品附件列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.4.5 获取制品附加信息 - ShowInstanceArtifactAddition

功能介绍

获取制品附加信息

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getTagAddition	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/additions/{addition}

表 4-231 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要
addition	是	String	制品附加信息

请求参数

表 4-232 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-233 响应 Body 参数

参数	参数类型	描述
build_histories	Array of **BuildHistory** objects	构建历史列表
total	Integer	构建历史条数

表 4-234 BuildHistory

参数	参数类型	描述
created_at	String	构建时间
created_by	String	构建命令
empty_layer	Boolean	是否空层
mediaType	String	镜像层格式
size	Integer	镜像层大小
digest	String	镜像层sha256信息

状态码：400

表 4-235 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-236 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-237 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-238 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-239 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/additions/{addition}
```

响应示例

状态码：200

查询制品附加信息成功

```
{
  "total" : 2,
  "build_histories" : [ {
    "mediaType" : "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size" : 699243,
    "digest" : "sha256:a5b4cfae24aaa36d4b7f3c9ee3b096c91dbc0a9290b383f5146225f8a5ee0b83",
    "created_by" : "/bin/sh -c #(nop) ADD file:aa56bc8f2fea9c0c81ca085bfa273ad1a3b0d46f51b8c9c61b483340c902024f in / ",
    "empty_layer" : false,
    "created_at" : "2021-01-01T00:00:00Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceArtifactAdditionSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceArtifactAdditionRequest request = new ShowInstanceArtifactAdditionRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
```

 版权所有 © 华为云计算技术有限公司

```
        request.withReference("{reference}");
        request.withAddition(ShowInstanceArtifactAdditionRequest.AdditionEnum.fromValue("{addition}"));
        try {
            ShowInstanceArtifactAdditionResponse response = client.showInstanceArtifactAddition(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceArtifactAdditionRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        request.addition = "{addition}"
        response = client.show_instance_artifact_addition(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
```

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceArtifactAdditionRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    request.Addition = model.GetShowInstanceArtifactAdditionRequestAdditionEnum().ADDITION
    response, err := client.ShowInstanceArtifactAddition(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	查询制品附加信息成功
400	错误的请求

 版权所有 © 华为云计算技术有限公司

状态码	描述
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.4.6 获取仓库实例的所有制品版本列表 - ListInstanceAllArtifacts

功能介绍

获取仓库实例的所有制品版本列表

接口约束

此接口只在企业仓库实例版本大于25.6.0以上的版本才支持

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listAllArtifacts	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/artifacts

版权所有 © 华为云计算技术有限公司

表 4-240 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-241 Query 参数

参数	是否必选	参数类型	描述
marker	否	Integer	分页查询时的查询标记，使用上一次接口调用返回的next_marker值。默认值为1。**注意：marker和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：marker和limit参数需要配套使用。**

请求参数

表 4-242 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-243 响应 Body 参数

参数	参数类型	描述
artifacts	Array of **Artifact** objects	制品列表
next_marker	Integer	下一次分页查询的起始ID。如果未返回该值，则表示数据已查询完毕

版权所有 © 华为云计算技术有限公司

表 4-244 Artifact

参数	参数类型	描述
digest	String	制品摘要
id	Integer	制品ID
repository_id	Integer	仓库ID
repository_name	String	仓库名称
type	String	制品类型 取值范围： • IMAGE：镜像制品
namespace_id	Integer	命名空间ID
media_type	String	制品MIME类型
manifest_media_type	String	制品元数据MIME类型
pull_time	String	最近一次拉取时间
push_time	String	最近一次上传时间
size	Integer	制品大小，单位：Byte
tags	Array of **ArtifactTag** objects	制品版本的Tag列表
extra_attrs	Map<String,Object>	制品的一些额外属性，比如制品的构建时间(created)、支持的操作系统(os)、支持CPU架构(architecture)等信息。
references	Array of **Reference** objects	当制品为多架构镜像索引时，显示其引用的所有子制品。

表 4-245 ArtifactTag

参数	参数类型	描述
id	Long	Tag ID
repository_id	Long	制品仓库ID
artifact_id	Long	制品版本ID
name	String	tag名称
push_time	String	tag的上传时间
pull_time	String	tag的下载时间

版权所有 © 华为云计算技术有限公司

表 4-246 Reference

参数	参数类型	描述
parent_id	Long	父制品的ID。
child_id	Long	子制品的ID。
child_digest	String	子制品的digest。
platform	**Platform** object	子制品的平台信息。

表 4-247 Platform

参数	参数类型	描述
architecture	String	制品适用的CPU架构。
os	String	制品适用的操作系统。
os.version	String	制品适用的操作系统版本。
os.features	Array of strings	制品适用的操作系统特性。
variant	String	同一系列 CPU 下的细分型号，例如 Intel i5-13400、i5-13600 属于同代不同 variant。

状态码：400

表 4-248 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-249 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-250 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-251 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-252 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/artifacts
```

响应示例

状态码：200

查询制品版本列表成功

```
{
  "next_marker" : 10,
  "artifacts" : [ {
    "id" : 3804,
    "pull_time" : "0001-01-01T00:00:00.000Z",
    "push_time" : "2021-02-03T10:22:11.840Z",
    "repository_id" : 402,
    "digest" : "sha256:3ee2a6bffb6dfec67be23fd0f6c0e8a48620277582e01800b4ad3c03ad0d9489",
    "manifest_media_type" : "application/vnd.docker.distribution.manifest.v2+json",
    "media_type" : "application/vnd.docker.container.image.v1+json",
    "namespace_id" : 87,
    "size" : 8049952,
    "type" : "IMAGE"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceAllArtifactsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceAllArtifactsRequest request = new ListInstanceAllArtifactsRequest();
        request.withInstanceId("{instance_id}");
```

 版权所有 © 华为云计算技术有限公司

```
        try {
            ListInstanceAllArtifactsResponse response = client.listInstanceAllArtifacts(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceAllArtifactsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_all_artifacts(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

 版权所有 © 华为云计算技术有限公司

```
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are 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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceAllArtifactsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceAllArtifacts(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	查询制品版本列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.4.7 获取制品仓库的 Tag 列表 - ListInstanceTags

功能介绍

获取制品仓库的Tag列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listTags	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/tags

表 4-253 路径参数

参数	是否必选	参数类型	描述
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。

表 4-254 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
is_accessory	否	Boolean	是否返回制品附件，默认为true

请求参数

表 4-255 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-256 响应 Body 参数

参数	参数类型	描述
total	Integer	制品总数

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
tags	Array of **RepositoryTag** objects	制品Tag列表

表 4-257 RepositoryTag

参数	参数类型	描述
name	String	版本名称
artifact_id	Integer	关联的制品ID
digest	String	关联的制品摘要
id	Integer	版本ID
repository_id	Integer	制品仓库仓库ID
type	String	制品类型 取值范围： ● IMAGE：镜像制品
namespace_id	Integer	命名空间ID
media_type	String	制品MIME类型
manifest_media_type	String	制品元数据MIME类型
pull_time	String	最近一次拉取时间
push_time	String	最近一次上传时间
size	Integer	制品大小，单位：Byte

状态码：400

表 4-258 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-259 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-260 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-261 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-262 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/tags
```

响应示例

状态码：200

查询制品仓库的Tag列表成功

```
{
  "total" : 1,
  "tags" : [ {
    "artifact_id" : 1978,
    "id" : 3804,
    "name" : "latest",
    "pull_time" : "0001-01-01T00:00:00.000Z",
    "push_time" : "2021-02-03T10:22:11.840Z",
    "repository_id" : 402,
    "digest" : "sha256:3ee2a6bffb6dfec67be23fd0f6c0e8a48620277582e01800b4ad3c03ad0d9489",
    "manifest_media_type" : "application/vnd.docker.distribution.manifest.v2+json",
    "media_type" : "application/vnd.docker.container.image.v1+json",
    "namespace_id" : 87,
    "size" : 8049952,
    "type" : "IMAGE"
  } ]
}
```

状态码

状态码	描述
200	查询制品仓库的Tag列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.4.8 删除制品的 Tag - DeleteInstanceTag

功能介绍

删除制品Tag

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

DELETE /v2/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/tags/{tag_name}

表 4-263 路径参数

参数	是否必选	参数类型	描述
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
tag_name	是	String	Tag名称

版权所有 © 华为云计算技术有限公司

请求参数

表 4-264 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除制品Tag成功

状态码：400

表 4-265 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-266 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-267 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-268 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-269 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/tags/{tag_name}
```

响应示例

无

版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除制品Tag成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.4.9 获取制品 Manifest 信息 - ShowInstanceArtifactManifest

功能介绍

获取制品的Manifest信息

接口约束

此接口只在注册表实例版本26.6.0及以上的版本才支持

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getArtifactManifest	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/manifest

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
instance_id	是	String	注册表实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

请求参数

表 4-271 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-272 响应 Body 参数

参数	参数类型	描述
-	String	

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-273 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-274 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-275 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-276 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-277 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/manifest
```

响应示例

状态码：200

查询制品Manifest信息成功

```
{
  "schemaVersion" : 2,
  "mediaType" : "application/vnd.docker.distribution.manifest.v2+json",
  "config" : {
    "mediaType" : "application/vnd.docker.container.image.v1+json",
    "size" : 1216,
    "digest" : "sha256:172acc6b1fe2930754a9f9d94703a7d571a1eb38cf54f93f47a5a47383baab20"
  },
  "layers" : [ {
    "mediaType" : "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size" : 96,
    "digest" : "sha256:da9a91b252b0ddcdb64ac42b23b8c1ae6ef3f020987988ebc882d9ec98144df0"
  } ]
}
```

状态码

状态码	描述
200	查询制品Manifest信息成功
400	错误的请求

版权所有 © 华为云计算技术有限公司

状态码	描述
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.5 访问凭证管理

4.5.1 获取临时访问凭证 - CreateInstanceTempCredential

功能介绍

获取临时访问凭证

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createTempCredential	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	sts::createServiceBearerToken

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/instances/{instance_id}/temp-credential

表 4-278 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-279 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：201

表 4-280 响应 Body 参数

参数	参数类型	描述
auth_token	String	临时访问凭证密码
created_at	String	创建时间
expire_date	String	过期时间
token_id	String	临时访问凭证ID
user_id	String	临时访问凭证的用户ID

状态码：400

表 4-281 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-282 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-283 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-284 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-285 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/temp-credential
```

响应示例

状态码：201

成功

```
{
  "auth_token" : "abc123xyz789",
  "created_at" : "2023-10-20T15:30:00Z",
  "expire_date" : "2023-10-21T15:30:00Z",
  "token_id" : "550e8400-e29b-41d4-a716-446655440000",
  "user_id" : "test_user_123"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceTempCredentialSolution {

    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}";
```

 版权所有 © 华为云计算技术有限公司

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceTempCredentialRequest request = new CreateInstanceTempCredentialRequest();
        request.withInstanceId("{instance_id}");
        try {
            CreateInstanceTempCredentialResponse response = client.createInstanceTempCredential(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceTempCredentialRequest()
        request.instance_id = "{instance_id}"
        response = client.create_instance_temp_credential(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceTempCredentialRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.CreateInstanceTempCredential(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	成功

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.5.2 创建长期访问凭证 - CreateInstanceLtCredential

功能介绍

创建长期访问凭证

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createLTCredential	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/instances/{instance_id}/long-term-credential

表 4-286 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-287 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-288 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	长期访问凭证名称，小写字母或数字开头，后面跟小写字母、数字、点、下划线或中划线（其中点、下划线、中划线不能直接连续），小写字母或数字结尾，1-64个字符。

响应参数

状态码：201

表 4-289 响应 Body 参数

参数	参数类型	描述
auth_token	String	访问凭证密码
created_at	String	创建时间
expire_date	String	过期时间

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
token_id	String	访问凭证ID
user_id	String	访问凭证用户名

状态码：400

表 4-290 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-291 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-292 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-293 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-294 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/long-term-credential

{
  "name" : "token_name"
}
```

响应示例

状态码：201

创建成功

```
{
  "auth_token" : "abc123xyz789",
  "created_at" : "2023-10-20T15:30:00Z",
  "expire_date" : "2023-10-21T15:30:00Z",
  "token_id" : "550e8400-e29b-41d4-a716-446655440000",
  "user_id" : "test_user_123"
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceLtCredentialSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceLtCredentialRequest request = new CreateInstanceLtCredentialRequest();
        request.withInstanceId("{instance_id}");
        CreateLongTermCredentialRequestBody body = new CreateLongTermCredentialRequestBody();
        body.withName("token_name");
        request.withBody(body);
        try {
            CreateInstanceLtCredentialResponse response = client.createInstanceLtCredential(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceLtCredentialRequest()
        request.instance_id = "{instance_id}"
        request.body = CreateLongTermCredentialRequestBody(
            name="token_name"
        )
        response = client.create_instance_lt_credential(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
```

 版权所有 © 华为云计算技术有限公司

```
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceLtCredentialRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.CreateLongTermCredentialRequestBody{
        Name: "token_name",
    }
    response, err := client.CreateInstanceLtCredential(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	创建成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.5.3 启用/停用长期访问凭证 - UpdateInstanceLtCredential

功能介绍

启用/停用长期访问凭证

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateLTCredential	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}

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
instance_id	是	String	企业仓库实例ID
credential_id	是	String	访问凭证ID，即token_id

请求参数

表 4-296 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-297 请求 Body 参数

参数	是否必选	参数类型	描述
enable	是	Boolean	是否启用

响应参数

状态码：200

成功更新访问凭证信息

状态码：400

表 4-298 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-299 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-300 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-301 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-302 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}

{
  "enable" : false
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateInstanceLtCredentialSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceLtCredentialRequest request = new UpdateInstanceLtCredentialRequest();
        request.withInstanceId("{instance_id}");
        request.withCredentialId("{credential_id}");
        UpdateLongTermCredentialRequestBody body = new UpdateLongTermCredentialRequestBody();
        body.withEnable(false);
        request.withBody(body);
        try {
            UpdateInstanceLtCredentialResponse response = client.updateInstanceLtCredential(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 import *

if __name__ == "__main__":
```

 版权所有 © 华为云计算技术有限公司

```
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceLtCredentialRequest()
        request.instance_id = "{instance_id}"
        request.credential_id = "{credential_id}"
        request.body = UpdateLongTermCredentialRequestBody(
            enable=False
        )
        response = client.update_instance_lt_credential(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceLtCredentialRequest{}
    request.InstanceId = "{instance_id}"
    request.CredentialId = "{credential_id}"
    request.Body = &model.UpdateLongTermCredentialRequestBody{
        Enable: false,
    }
    response, err := client.UpdateInstanceLtCredential(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	成功更新访问凭证信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.5.4 删除长期访问凭证 - DeleteInstanceLtCredential

功能介绍

删除长期访问凭证

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteLTCredential	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}

表 4-303 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
credential_id	是	String	访问凭证ID，即token_id

请求参数

表 4-304 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：204

删除成功

状态码：400

表 4-305 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-306 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-307 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-308 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-309 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceLtCredentialSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceLtCredentialRequest request = new DeleteInstanceLtCredentialRequest();
        request.withInstanceId("{instance_id}");
        request.withCredentialId("{credential_id}");
        try {
            DeleteInstanceLtCredentialResponse response = client.deleteInstanceLtCredential(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceLtCredentialRequest()
        request.instance_id = "{instance_id}"
        request.credential_id = "{credential_id}"
        response = client.delete_instance_lt_credential(request)
```

 版权所有 © 华为云计算技术有限公司

```
        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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceLtCredentialRequest{}
    request.InstanceId = "{instance_id}"
    request.CredentialId = "{credential_id}"
    response, err := client.DeleteInstanceLtCredential(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.5.5 查询长期访问凭证列表 - ListInstanceLtCredentials

功能介绍

查询长期访问凭证列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listLTCredentials	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/long-term-credentials

表 4-310 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-311 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为100，最大值为100。**注意：offset和limit参数需要配套使用。**
self_only	否	Boolean	当self_only为false时拥有te_admin角色的用户可以查询实例下所有的长期登录凭证信息

请求参数

表 4-312 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-313 响应 Body 参数

参数	参数类型	描述
total	Integer	长期访问凭据总数
auth_tokens	Array of **AuthToken** objects	长期访问凭证列表

表 4-314 AuthToken

参数	参数类型	描述
created_at	String	创建时间
enable	Boolean	是否启用
expire_date	String	过期时间
instance_id	String	实例ID
name	String	访问凭证名称
token_id	String	访问凭证ID
user_id	String	访问凭证用户名
user_profile	String	用户访问凭据信息

状态码：400

表 4-315 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-316 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-317 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-318 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-319 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/long-term-credentials
```

响应示例

状态码：200

查询长期访问凭证列表成功

```
{
  "auth_tokens" : [ {
    "created_at" : "2021-03-16T03:05:41.863418Z",
    "enable" : true,
    "expire_date" : "2031-03-16T03:05:41.863418Z",
    "instance_id" : "0241f1da-7cbb-11eb-8050-fa163ed64718",
    "name" : "test2",
    "token_id" : "7e1b2fbd-8604-11eb-bb8c-fa163ed64718",
    "user_id" : "ff4f44473a1d4ab3847906410af2d152"
  }, {
    "created_at" : "2021-03-08T02:33:11.384714Z",
    "enable" : true,
    "expire_date" : "2031-03-08T02:33:11.384714Z",
    "instance_id" : "0241f1da-7cbb-11eb-8050-fa163ed64718",
    "name" : "test",
    "token_id" : "a039c15d-7fb6-11eb-aefc-fa163ed64718",
    "user_id" : "0b8d109de280d4841ff2c015890293e9"
  } ],
  "total" : 2
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceLtCredentialsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
```

 版权所有 © 华为云计算技术有限公司

```
            .withCredential(auth)
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
    ListInstanceLtCredentialsRequest request = new ListInstanceLtCredentialsRequest();
    request.withInstanceId("{instance_id}");
    try {
        ListInstanceLtCredentialsResponse response = client.listInstanceLtCredentials(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceLtCredentialsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_lt_credentials(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceLtCredentialsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceLtCredentials(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	查询长期访问凭证列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源

 版权所有 © 华为云计算技术有限公司

状态码	描述
500	内部错误

错误码

请参见**错误码**。

4.6 标签管理

4.6.1 按照标签检索资源列表 - ListInstanceResourceInstances

功能介绍

按照标签检索资源列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listResourceInstances	List	instance *	-	-	-
		-	g:TagKeys		

URI

POST /v2/{project_id}/{resource_type}/resource-instances/filter

版权所有 © 华为云计算技术有限公司

表 4-320 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： ● instances：Instance实例 ● namespaces：命名空间 ● repositories：仓库

表 4-321 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为200，最小值为1，最大值为200。**注意：offset和limit参数需要配套使用。**

请求参数

表 4-322 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-323 请求 Body 参数

参数	是否必选	参数类型	描述
tags	否	Array of **TagsFilter** objects	包含标签，最多包含50个key，每个key下面的value最多10个，每个key对应的value可以为空数组但结构体不能缺失。Key不能重复，同一个key中values不能重复。结果返回包含所有标签的资源列表，key之间是与的关系，key-value结构中value是或的关系。无tag过滤条件时返回全量数据。

表 4-324 TagsFilter

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。 key不能为空。
values	是	Array of strings	标签值列表，每个值最大长度255个unicode字符，校验和使用之前先trim 前后半角空格。value可为空数组但不可缺省。如果values为空列表，则表示any_value（查询任意value）。value之间为或的关系。。

响应参数

状态码：200

表 4-325 响应 Body 参数

参数	参数类型	描述
resources	Array of **Resource** objects	资源列表
total_count	Integer	资源数量

表 4-326 Resource

参数	参数类型	描述
resource_id	String	资源id

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
resource_detail	Object	资源详情
resource_name	String	资源名称
tags	Array of ResourceTag objects	资源标签列表
sys_tags	Array of ResourceTag objects	-| 系统标签列表。仅op_service权限才可以可以获取此字段：目前只包含一个resource_tag 结构体key：_sys_enterprise_project_idvalue：企业项目id，0表示默认企业项目非op_service场景不能返回此字段。

表 4-327 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-328 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-329 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-330 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-331 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-332 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/resource-instances/filter

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

响应示例

状态码：200

查询资源列表

```
{
  "resources" : [ {
    "resource_detail" : null,
    "resource_id" : "5aa119a8-d25b-45a7-8d1b-88e127885632",
    "resource_name" : "test-instance",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    }, {
      "key" : "key2",
      "value" : "value1"
    } ],
    "sys_tags" : [ {
      "key" : "_sys_enterprise_project_id",
      "value" : "5aa119a8-d25b-45a7-8d1b-88e127885635"
    } ]
  } ],
  "total_count" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class ListInstanceResourceInstancesSolution {

    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");
```

 版权所有 © 华为云计算技术有限公司

```
        String projectId = "{project_id}";

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceResourceInstancesRequest request = new ListInstanceResourceInstancesRequest();

request.withResourceType(ListInstanceResourceInstancesRequest.ResourceTypeEnum.fromValue("{resource_t
ype}"));
        ListResourceInstancesRequestBody body = new ListResourceInstancesRequestBody();
        List<String> listTagsValues = new ArrayList<>();
        listTagsValues.add("*value1");
        listTagsValues.add("value2");
        List<TagsFilter> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new TagsFilter()
                .withKey("key1")
                .withValues(listTagsValues)
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            ListInstanceResourceInstancesResponse response = client.listInstanceResourceInstances(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
```

 版权所有 © 华为云计算技术有限公司

```
        .build()

    try:
        request = ListInstanceResourceInstancesRequest()
        request.resource_type = "{resource_type}"
        listValuesTags = [
            "*value1",
            "value2"
        ]
        listTagsbody = [
            TagsFilter(
                key="key1",
                values=listValuesTags
            )
        ]
        request.body = ListResourceInstancesRequestBody(
            tags=listTagsbody
        )
        response = client.list_instance_resource_instances(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceResourceInstancesRequest{}
    request.ResourceType =
model.GetListInstanceResourceInstancesRequestResourceTypeEnum().RESOURCE_TYPE
    var listValuesTags = []string{
        "*value1",
        "value2",
    }
    var listTagsbody = []model.TagsFilter{
        {
            Key: "key1",
            Values: listValuesTags,
        },
    }
    request.Body = &model.ListResourceInstancesRequestBody{
        Tags: &listTagsbody,
    }
    response, err := client.ListInstanceResourceInstances(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	查询资源列表
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.2 按照标签检索资源数量 - ShowInstanceResourceInstancesCount

功能介绍

按照标签检索资源数量

版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getResourceInstancesCount	Read	instance *	-	-	-
		-	g:TagKeys		

URI

POST /v2/{project_id}/{resource_type}/resource-instances/count

表 4-333 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例 • namespaces：命名空间 • repositories：仓库

版权所有 © 华为云计算技术有限公司

请求参数

表 4-334 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-335 请求 Body 参数

参数	是否必选	参数类型	描述
tags	否	Array of **TagsFilter** objects	包含标签，最多包含50个key，每个key下面的value最多10个，每个key对应的value可以为空数组但结构体不能缺失。Key不能重复，同一个key中values不能重复。结果返回包含所有标签的资源列表，key之间是与的关系，key-value结构中value是或的关系。无tag过滤条件时返回全量数据。

表 4-336 TagsFilter

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。 key不能为空。
values	是	Array of strings	标签值列表，每个值最大长度255个unicode字符，校验和使用之前先trim 前后半角空格。value可为空数组但不可缺省。如果values为空列表，则表示any_value（查询任意value）。value之间为或的关系。。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-337 响应 Body 参数

参数	参数类型	描述
total_count	Integer	资源数量

状态码：400

表 4-338 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-339 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-340 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-341 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-342 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/resource-instances/count

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

响应示例

状态码：200

查询资源数量

```
{
  "total_count" : 1
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class ShowInstanceResourceInstancesCountSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceResourceInstancesCountRequest request = new ShowInstanceResourceInstancesCountRequest();

request.withResourceType(ShowInstanceResourceInstancesCountRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        ListResourceInstancesRequestBody body = new ListResourceInstancesRequestBody();
        List<String> listTagsValues = new ArrayList<>();
        listTagsValues.add("*value1");
        listTagsValues.add("value2");
        List<TagsFilter> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new TagsFilter()
                .withKey("key1")
                .withValues(listTagsValues)
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            ShowInstanceResourceInstancesCountResponse response = client.showInstanceResourceInstancesCount(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());
        }
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceResourceInstancesCountRequest()
        request.resource_type = "{resource_type}"
        listValuesTags = [
            "*value1",
            "value2"
        ]
        listTagsbody = [
            TagsFilter(
                key="key1",
                values=listValuesTags
            )
        ]
        request.body = ListResourceInstancesRequestBody(
            tags=listTagsbody
        )
        response = client.show_instance_resource_instances_count(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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
```

 版权所有 © 华为云计算技术有限公司

```
variables and decrypted during use to ensure security.
    // In this example, AK and SK are 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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceResourceInstancesCountRequest{}
    request.ResourceType =
model.GetShowInstanceResourceInstancesCountRequestResourceTypeEnum().RESOURCE_TYPE
    var listValuesTags = []string{
        "*value1",
        "value2",
    }
    var listTagsbody = []model.TagsFilter{
        {
            Key: "key1",
            Values: listValuesTags,
        },
    }
    request.Body = &model.ListResourceInstancesRequestBody{
        Tags: &listTagsbody,
    }
    response, err := client.ShowInstanceResourceInstancesCount(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	查询资源数量

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.3 批量添加资源标签 - CreateInstanceResourceTags

功能介绍

批量添加资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createResourceTags	Tagging	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
		-	• g:RequestTag/<tag-key> • g:TagKeys		

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/tags/create

表 4-343 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例 • namespaces：命名空间 • repositories：仓库
resource_id	是	String	资源id

请求参数

表 4-344 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-345 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of **ResourceTag** objects	批量操作标签

版权所有 © 华为云计算技术有限公司

表 4-346 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。
value	是	String	标签值，每个值最大长度255个unicode字符。

响应参数

状态码：204

批量添加资源标签成功

状态码：400

表 4-347 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-348 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-349 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-350 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-351 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags/create

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


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceResourceTagsRequest request = new CreateInstanceResourceTagsRequest();

request.withResourceType(CreateInstanceResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");
        CreateResourceTagsRequestBody body = new CreateResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key")
                .withValue("value3")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            CreateInstanceResourceTagsResponse response = client.createInstanceResourceTags(request);
            System.out.println(response.toString());
```

 版权所有 © 华为云计算技术有限公司

```
        } 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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1",
                value="value1"
            ),
            ResourceTag(
                key="key",
                value="value3"
            )
        ]
        request.body = CreateResourceTagsRequestBody(
            tags=listTagsbody
        )
        response = client.create_instance_resource_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceResourceTagsRequest{}
    request.ResourceType = model.GetCreateInstanceResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    var listTagsbody = []model.ResourceTag{
        {
            Key: "key1",
            Value: "value1",
        },
        {
            Key: "key",
            Value: "value3",
        },
    }
    request.Body = &model.CreateResourceTagsRequestBody{
        Tags: listTagsbody,
    }
    response, err := client.CreateInstanceResourceTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	批量添加资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.4 批量删除资源标签 - DeleteInstanceResourceTags

功能介绍

批量删除资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteResourceTags	Tagging	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-
		-	• g:RequestTag/<tag-key> • g:TagKeys		

URI

DELETE /v2/{project_id}/{resource_type}/{resource_id}/tags/delete

表 4-352 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例 • namespaces：命名空间 • repositories：仓库
resource_id	是	String	资源id

请求参数

表 4-353 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-354 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of **ResourceTag** objects	批量操作标签

表 4-355 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。
value	是	String	标签值，每个值最大长度255个unicode字符。

响应参数

状态码：204

批量删除资源标签成功

状态码：400

表 4-356 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-357 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-358 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-359 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-360 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags/delete

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

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class DeleteInstanceResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceResourceTagsRequest request = new DeleteInstanceResourceTagsRequest();

request.withResourceType(DeleteInstanceResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");
        DeleteResourceTagsRequestBody body = new DeleteResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
```

 版权所有 © 华为云计算技术有限公司

```
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key")
                .withValue("value3")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            DeleteInstanceResourceTagsResponse response = client.deleteInstanceResourceTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1",
                value="value1"
            ),
            ResourceTag(
                key="key",
                value="value3"
            )
        ]
```

 版权所有 © 华为云计算技术有限公司

```
        request.body = DeleteResourceTagsRequestBody(
            tags=listTagsbody
        )
        response = client.delete_instance_resource_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceResourceTagsRequest{}
    request.ResourceType = model.GetDeleteInstanceResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    var listTagsbody = []model.ResourceTag{
        {
            Key: "key1",
            Value: "value1",
        },
        {
            Key: "key",
            Value: "value3",
```

 版权所有 © 华为云计算技术有限公司

```
        },
    }
    request.Body = &model.DeleteResourceTagsRequestBody{
        Tags: listTagsbody,
    }
    response, err := client.DeleteInstanceResourceTags(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	批量删除资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.5 查询项目下所有实例标签 - ListInstanceProjectTags

功能介绍

查询项目下所有实例标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getProjectTags	Read	-	-	-	-

URI

GET /v2/{project_id}/instances/tags

表 4-361 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

请求参数

表 4-362 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-363 响应 Body 参数

参数	参数类型	描述
tags	Array of **ProjectTag** objects	标签列表

版权所有 © 华为云计算技术有限公司

表 4-364 ProjectTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
values	Array of strings	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-365 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-366 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-367 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-368 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-369 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/tags
```

响应示例

状态码：200

查询项目标签成功

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

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceProjectTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceProjectTagsRequest request = new ListInstanceProjectTagsRequest();
        try {
            ListInstanceProjectTagsResponse response = client.listInstanceProjectTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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
```

 版权所有 © 华为云计算技术有限公司

```
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListInstanceProjectTagsRequest()
        response = client.list_instance_project_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceProjectTagsRequest{}
    response, err := client.ListInstanceProjectTags(request)
```

 版权所有 © 华为云计算技术有限公司

```
    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	查询项目标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.6 查询某种资源类型的标签列表 - ListResourcesTagsByProjectId

功能介绍

查询某种资源类型的标签列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。

版权所有 © 华为云计算技术有限公司

- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/{resource_type}/tags

表 4-370 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： ● instances：Instance实例 ● namespaces：命名空间 ● repositories：仓库

表 4-371 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为200，最小值为1，最大值为200。**注意：offset和limit参数需要配套使用。**
instance_id	否	String	当resource_type为repositories或namespaces时，支持通过instance_id进行过滤查询，用于获取指定注册表实例下对应资源类型的标签列表。

请求参数

表 4-372 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-373 响应 Body 参数

参数	参数类型	描述
tags	Array of **ProjectTag** objects	标签列表

表 4-374 ProjectTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
values	Array of strings	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-375 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-376 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：403

表 4-377 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-378 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-379 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/{resource_type}/tags
```

响应示例

状态码：200

查询某种资源类型的标签列表成功

版权所有 © 华为云计算技术有限公司

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

状态码

状态码	描述
200	查询某种资源类型的标签列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.7 查询资源标签 - ListInstanceResourceTags

功能介绍

查询资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getResourceTags	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/{resource_type}/{resource_id}/tags

表 4-380 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例 • namespaces：命名空间 • repositories：仓库
resource_id	是	String	资源id

请求参数

表 4-381 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-382 响应 Body 参数

参数	参数类型	描述
tags	Array of ResourceTag objects	资源标签列表

表 4-383 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-384 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-385 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-386 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-387 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-388 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags
```

响应示例

状态码：200

查询资源标签成功

```
{
  "tags" : [ {
```


版权所有 © 华为云计算技术有限公司

```
    "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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceResourceTagsRequest request = new ListInstanceResourceTagsRequest();

request.withResourceType(ListInstanceResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");
        try {
            ListInstanceResourceTagsResponse response = client.listInstanceResourceTags(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        response = client.list_instance_resource_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
        return
    }

    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceResourceTagsRequest{}
    request.ResourceType = model.GetListInstanceResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    response, err := client.ListInstanceResourceTags(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	查询资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.8 按照标签检索子资源列表 - ListSubResourceInstances

功能介绍

按照标签检索子资源列表

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/resource-instances/filter

表 4-389 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例
resource_id	是	String	资源id
sub_resource_type	是	String	资源类型 取值范围： • namespaces：命名空间

表 4-390 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为200，最小值为1，最大值为200。**注意：offset和limit参数需要配套使用。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-391 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-392 请求 Body 参数

参数	是否必选	参数类型	描述
tags	否	Array of **TagsFilter** objects	包含标签，最多包含50个key，每个key下面的value最多10个，每个key对应的value可以为空数组但结构体不能缺失。Key不能重复，同一个key中values不能重复。结果返回包含所有标签的资源列表，key之间是与的关系，key-value结构中value是或的关系。无tag过滤条件时返回全量数据。

表 4-393 TagsFilter

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。 key不能为空。
values	是	Array of strings	标签值列表，每个值最大长度255个unicode字符，校验和使用之前先trim 前后半角空格。value可为空数组但不可缺省。如果values为空列表，则表示any_value（查询任意value）。value之间为或的关系。。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-394 响应 Body 参数

参数	参数类型	描述
resources	Array of **Resource** objects	资源列表
total_count	Integer	资源数量

表 4-395 Resource

参数	参数类型	描述
resource_id	String	资源id
resource_detail	Object	资源详情
resource_name	String	资源名称
tags	Array of **ResourceTag** objects	资源标签列表
sys_tags	Array of **ResourceTag** objects	-| 系统标签列表。仅op_service权限才可以可以获取此字段：目前只包含一个resource_tag 结构体key：_sys_enterprise_project_idvalue：企业项目id，0表示默认企业项目非op_service场景不能返回此字段。

表 4-396 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-397 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-398 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-399 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-400 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-401 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/resource-instances/filter

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

响应示例

状态码：200

查询子资源列表

```
{
  "resources" : [ {
    "resource_detail" : null,
    "resource_id" : "5aa119a8-d25b-45a7-8d1b-88e127885632",
    "resource_name" : "test-instance",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    }, {
      "key" : "key2",
      "value" : "value1"
    } ],
    "sys_tags" : [ {
      "key" : "_sys_enterprise_project_id",
      "value" : "5aa119a8-d25b-45a7-8d1b-88e127885635"
    } ]
  } ],
  "total_count" : 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;
```

 版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class ListSubResourceInstancesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListSubResourceInstancesRequest request = new ListSubResourceInstancesRequest();

request.withResourceType(ListSubResourceInstancesRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");

request.withSubResourceType(ListSubResourceInstancesRequest.SubResourceTypeEnum.fromValue("{sub_resource_type}"));
        ListResourceInstancesRequestBody body = new ListResourceInstancesRequestBody();
        List<String> listTagsValues = new ArrayList<>();
        listTagsValues.add("*value1");
        listTagsValues.add("value2");
        List<TagsFilter> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new TagsFilter()
                .withKey("key1")
                .withValues(listTagsValues)
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            ListSubResourceInstancesResponse response = client.listSubResourceInstances(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSubResourceInstancesRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        request.sub_resource_type = "{sub_resource_type}"
        listValuesTags = [
            "*value1",
            "value2"
        ]
        listTagsbody = [
            TagsFilter(
                key="key1",
                values=listValuesTags
            )
        ]
        request.body = ListResourceInstancesRequestBody(
            tags=listTagsbody
        )
        response = client.list_sub_resource_instances(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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.
```

 版权所有 © 华为云计算技术有限公司

```
    // In this example, AK and SK are 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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListSubResourceInstancesRequest{}
    request.ResourceType = model.GetListSubResourceInstancesRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    request.SubResourceType =
model.GetListSubResourceInstancesRequestSubResourceTypeEnum().SUB_RESOURCE_TYPE
    var listValuesTags = []string{
        "*value1",
        "value2",
    }
    var listTagsbody = []model.TagsFilter{
        {
            Key: "key1",
            Values: listValuesTags,
        },
    }
    request.Body = &model.ListResourceInstancesRequestBody{
        Tags: &listTagsbody,
    }
    response, err := client.ListSubResourceInstances(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	查询子资源列表

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.9 按照标签检索子资源数量 - ShowSubResourceInstancesCount

功能介绍

按照标签检索子资源数量

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/resource-instances/count

表 4-402 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
resource_type	是	String	资源类型 取值范围： • instances：Instance实例
resource_id	是	String	资源id
sub_resource_type	是	String	资源类型 取值范围： • namespaces：命名空间

请求参数

表 4-403 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-404 请求 Body 参数

参数	是否必选	参数类型	描述
tags	否	Array of **TagsFilter** objects	包含标签，最多包含50个key，每个key下面的value最多10个，每个key对应的value可以为空数组但结构体不能缺失。Key不能重复，同一个key中values不能重复。结果返回包含所有标签的资源列表，key之间是与的关系，key-value结构中value是或的关系。无tag过滤条件时返回全量数据。

表 4-405 TagsFilter

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。 key不能为空。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
values	是	Array of strings	标签值列表，每个值最大长度255个unicode字符，校验和使用之前先trim 前后半角空格。value可为空数组但不可缺省。如果values为空列表，则表示any_value（查询任意value）。value之间为或的关系。。

响应参数

状态码：200

表 4-406 响应 Body 参数

参数	参数类型	描述
total_count	Integer	资源数量

状态码：400

表 4-407 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-408 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：403

表 4-409 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-410 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-411 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/resource-instances/count

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


版权所有 © 华为云计算技术有限公司

```
  } ]
}
```

响应示例

状态码：200

查询子资源数量

```
{
  "total_count" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class ShowSubResourceInstancesCountSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowSubResourceInstancesCountRequest request = new ShowSubResourceInstancesCountRequest();

request.withResourceType(ShowSubResourceInstancesCountRequest.ResourceTypeEnum.fromValue("{resourc
e_type}"));
        request.withResourceId("{resource_id}");

request.withSubResourceType(ShowSubResourceInstancesCountRequest.SubResourceTypeEnum.fromValue("{
sub_resource_type}"));
        ListResourceInstancesRequestBody body = new ListResourceInstancesRequestBody();
        List<String> listTagsValues = new ArrayList<>();
        listTagsValues.add("*value1");
        listTagsValues.add("value2");
        List<TagsFilter> listbodyTags = new ArrayList<>();
```

 版权所有 © 华为云计算技术有限公司

```
        listbodyTags.add(
            new TagsFilter()
                .withKey("key1")
                .withValues(listTagsValues)
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            ShowSubResourceInstancesCountResponse response =
client.showSubResourceInstancesCount(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowSubResourceInstancesCountRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        request.sub_resource_type = "{sub_resource_type}"
        listValuesTags = [
            "*value1",
            "value2"
        ]
        listTagsbody = [
            TagsFilter(
                key="key1",
                values=listValuesTags
            )
        ]
        request.body = ListResourceInstancesRequestBody(
            tags=listTagsbody
```

 版权所有 © 华为云计算技术有限公司

```
        )
        response = client.show_sub_resource_instances_count(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowSubResourceInstancesCountRequest{}
    request.ResourceType = model.GetShowSubResourceInstancesCountRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    request.SubResourceType = model.GetShowSubResourceInstancesCountRequestSubResourceTypeEnum().SUB_RESOURCE_TYPE
    var listValuesTags = []string{
        "*value1",
        "value2",
    }
    var listTagsbody = []model.TagsFilter{
        {
            Key: "key1",
            Values: listValuesTags,
```

```
        },
    }
    request.Body = &model.ListResourceInstancesRequestBody{
        Tags: &listTagsbody,
    }
    response, err := client.ShowSubResourceInstancesCount(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	查询子资源数量
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.10 批量添加子资源标签 - CreateSubResourceTags

功能介绍

批量添加子资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/{sub_resource_id}/tags/create

表 4-412 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例
resource_id	是	String	资源id
sub_resource_type	是	String	资源类型 取值范围： • namespaces：命名空间
sub_resource_id	是	String	子资源id

请求参数

表 4-413 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-414 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of **ResourceTag** objects	批量操作标签

版权所有 © 华为云计算技术有限公司

表 4-415 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。
value	是	String	标签值，每个值最大长度255个unicode字符。

响应参数

状态码：204

批量添加子资源标签成功

状态码：400

表 4-416 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-417 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-418 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-419 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-420 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/
{sub_resource_id}/tags/create

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


版权所有 © 华为云计算技术有限公司

```
 }]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateSubResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateSubResourceTagsRequest request = new CreateSubResourceTagsRequest();

request.withResourceType(CreateSubResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");

request.withSubResourceType(CreateSubResourceTagsRequest.SubResourceTypeEnum.fromValue("{sub_resource_type}"));
        request.withSubResourceId("{sub_resource_id}");
        CreateResourceTagsRequestBody body = new CreateResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key")
```

 版权所有 © 华为云计算技术有限公司

```
                .withValue("value3")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            CreateSubResourceTagsResponse response = client.createSubResourceTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateSubResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        request.sub_resource_type = "{sub_resource_type}"
        request.sub_resource_id = "{sub_resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1",
                value="value1"
            ),
            ResourceTag(
                key="key",
                value="value3"
            )
        ]
        request.body = CreateResourceTagsRequestBody(
            tags=listTagsbody
        )
        response = client.create_sub_resource_tags(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateSubResourceTagsRequest{}
    request.ResourceType = model.GetCreateSubResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    request.SubResourceType = model.GetCreateSubResourceTagsRequestSubResourceTypeEnum().SUB_RESOURCE_TYPE
    request.SubResourceId = "{sub_resource_id}"
    var listTagsbody = []model.ResourceTag{
        {
            Key: "key1",
            Value: "value1",
        },
        {
            Key: "key",
            Value: "value3",
        },
    }
    request.Body = &model.CreateResourceTagsRequestBody{
```

 版权所有 © 华为云计算技术有限公司

```
        Tags: listTagsbody,
    }
    response, err := client.CreateSubResourceTags(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	批量添加子资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.11 批量删除子资源标签 - DeleteSubResourceTags

功能介绍

批量删除子资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

 版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

DELETE /v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/{sub_resource_id}/tags/delete

表 4-421 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例
resource_id	是	String	资源id
sub_resource_type	是	String	资源类型 取值范围： • namespaces：命名空间
sub_resource_id	是	String	子资源id

请求参数

表 4-422 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-423 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of **ResourceTag** objects	批量操作标签

版权所有 © 华为云计算技术有限公司

表 4-424 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	标签键，最大长度128个unicode字符。
value	是	String	标签值，每个值最大长度255个unicode字符。

响应参数

状态码：204

批量删除子资源标签成功

状态码：400

表 4-425 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-426 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-427 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-428 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-429 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/
{sub_resource_id}/tags/delete

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


版权所有 © 华为云计算技术有限公司

```
 }]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class DeleteSubResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteSubResourceTagsRequest request = new DeleteSubResourceTagsRequest();

request.withResourceType(DeleteSubResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");

request.withSubResourceType(DeleteSubResourceTagsRequest.SubResourceTypeEnum.fromValue("{sub_resource_type}"));
        request.withSubResourceId("{sub_resource_id}");
        DeleteResourceTagsRequestBody body = new DeleteResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key")
```

 版权所有 © 华为云计算技术有限公司

```
                .withValue("value3")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            DeleteSubResourceTagsResponse response = client.deleteSubResourceTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteSubResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        request.sub_resource_type = "{sub_resource_type}"
        request.sub_resource_id = "{sub_resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1",
                value="value1"
            ),
            ResourceTag(
                key="key",
                value="value3"
            )
        ]
        request.body = DeleteResourceTagsRequestBody(
            tags=listTagsbody
        )
        response = client.delete_sub_resource_tags(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteSubResourceTagsRequest{}
    request.ResourceType = model.GetDeleteSubResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    request.SubResourceType = model.GetDeleteSubResourceTagsRequestSubResourceTypeEnum().SUB_RESOURCE_TYPE
    request.SubResourceId = "{sub_resource_id}"
    var listTagsbody = []model.ResourceTag{
        {
            Key: "key1",
            Value: "value1",
        },
        {
            Key: "key",
            Value: "value3",
        },
    }
    request.Body = &model.DeleteResourceTagsRequestBody{
```

 版权所有 © 华为云计算技术有限公司

```
        Tags: listTagsbody,
    }
    response, err := client.DeleteSubResourceTags(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	批量删除子资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.12 查询实例下所有命名空间标签 - ListNamespaceTags

功能介绍

查询实例下所有命名空间标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces-tags

表 4-430 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	实例id

请求参数

表 4-431 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-432 响应 Body 参数

参数	参数类型	描述
tags	Array of **ProjectTag** objects	标签列表

表 4-433 ProjectTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。
values	Array of strings	标签值，每个值最大长度255个unicode字符。

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-434 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-435 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-436 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-437 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-438 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces-tags
```

响应示例

状态码：200

查询实例下所有命名空间标签成功

```
{
  "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.swr.v2.region.SwrRegion;
```


版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListNamespaceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListNamespaceTagsRequest request = new ListNamespaceTagsRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListNamespaceTagsResponse response = client.listNamespaceTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
```

 版权所有 © 华为云计算技术有限公司

```
        .build()

    try:
        request = ListNamespaceTagsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_namespace_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

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

 版权所有 © 华为云计算技术有限公司

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	查询实例下所有命名空间标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.6.13 查询子资源标签 - ListSubResourceTags

功能介绍

查询子资源标签

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/{sub_resource_id}/tags

版权所有 © 华为云计算技术有限公司

表 4-439 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
resource_type	是	String	资源类型 取值范围： • instances：Instance实例
resource_id	是	String	资源id
sub_resource_type	是	String	资源类型 取值范围： • namespaces：命名空间
sub_resource_id	是	String	子资源id

请求参数

表 4-440 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-441 响应 Body 参数

参数	参数类型	描述
tags	Array of **ResourceTag** objects	资源标签列表

表 4-442 ResourceTag

参数	参数类型	描述
key	String	标签键，最大长度128个unicode字符。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
value	String	标签值，每个值最大长度255个unicode字符。

状态码：400

表 4-443 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-444 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-445 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-446 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-447 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/{resource_type}/{resource_id}/{sub_resource_type}/{sub_resource_id}/tags
```

响应示例

状态码：200

查询子资源标签成功

```
{
  "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;
```

 版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListSubResourceTagsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListSubResourceTagsRequest request = new ListSubResourceTagsRequest();

request.withResourceType(ListSubResourceTagsRequest.ResourceTypeEnum.fromValue("{resource_type}"));
        request.withResourceId("{resource_id}");

request.withSubResourceType(ListSubResourceTagsRequest.SubResourceTypeEnum.fromValue("{sub_resource_type}"));
        request.withSubResourceId("{sub_resource_id}");
        try {
            ListSubResourceTagsResponse response = client.listSubResourceTags(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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.
```

 版权所有 © 华为云计算技术有限公司

```
    # In this example, AK and SK are 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSubResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        request.sub_resource_type = "{sub_resource_type}"
        request.sub_resource_id = "{sub_resource_id}"
        response = client.list_sub_resource_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
```

```
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListSubResourceTagsRequest{}
    request.ResourceType = model.GetListSubResourceTagsRequestResourceTypeEnum().RESOURCE_TYPE
    request.ResourceId = "{resource_id}"
    request.SubResourceType =
model.GetListSubResourceTagsRequestSubResourceTypeEnum().SUB_RESOURCE_TYPE
    request.SubResourceId = "{sub_resource_id}"
    response, err := client.ListSubResourceTags(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	查询子资源标签成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.7 域名管理

4.7.1 增加域名 - AddDomainName

功能介绍

增加域名

接口约束

无

版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:addDomainName	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	scm:cert:export

URI

POST /v2/{project_id}/instances/{instance_id}/domainname

表 4-448 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-449 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-450 请求 Body 参数

参数	是否必选	参数类型	描述
domain_name	是	String	域名，只能由字母、数字、中划线、星号组成， 星号只能在开头，中划线不能在开头或末尾，至少包含两个字符串，单个字符串不 超过63个字符，字符串间以点分隔，且总 长度不超过100个字符。例如：example.com 或 *.example.com。
certificate_id	是	String	SCM服务的证书ID

响应参数

状态码：201

表 4-451 响应 Body 参数

参数	参数类型	描述
domain_name_info	**DomainNameInfo** object	域名信息

表 4-452 DomainNameInfo

参数	参数类型	描述
uid	String	域名ID
domain_id	String	租户ID
project_id	String	项目ID
instance_id	String	实例ID
type	String	域名类型 取值范围： • default：默认域名 • custom：自定义域名
domain_name	String	域名，只能由字母、数字、中划线、星号组成， 星号只能在开头，中划线不能在开头或末 尾，至少包含两个字符串，单个字符串不 超过63个字符，字符串间以点分隔，且总 长度不超过100个字符。例如： example.com 或 *.example.com。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
certificate_id	String	SCM服务的证书ID
created_at	String	增加域名的时间
updated_at	String	更新域名的时间

状态码：400

表 4-453 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-454 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-455 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-456 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-457 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/domainname

{
  "domain_name" : "cn-north-1.myhuaweicloud.com",
  "certificate_id" : "scs101"
}
```

响应示例

状态码：201

增加域名成功

```
{
  "domain_name_info" : {
    "uid" : "42892305-c7a8-4206-9b94-ad5e47aaca15",
    "domain_id" : "xxxx",
    "project_id" : "xxxx",
    "instance_id" : "f33480c6-3174-11f0-a271-0255ac1018a6",
    "domain_name" : "test.myhuaweicloud.com",
    "certificate_id" : "scs17",
    "type" : "custom",
    "created_at" : "2025-05-22T20:28:30.467856+08:00",
    "updated_at" : "2025-05-22T20:28:30.467857+08:00"
  }
}
```


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class AddDomainNameSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        AddDomainNameRequest request = new AddDomainNameRequest();
        request.withInstanceId("{instance_id}");
        AddDomainNameRequestBody body = new AddDomainNameRequestBody();
        body.withCertificateId("scs101");
        body.withDomainName("cn-north-1.myhuaweicloud.com");
        request.withBody(body);
        try {
            AddDomainNameResponse response = client.addDomainName(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
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddDomainNameRequest()
        request.instance_id = "{instance_id}"
        request.body = AddDomainNameRequestBody(
            certificate_id="scs101",
            domain_name="cn-north-1.myhuaweicloud.com"
        )
        response = client.add_domain_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

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.AddDomainNameRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.AddDomainNameRequestBody{
        CertificateId: "scs101",
        DomainName: "cn-north-1.myhuaweicloud.com",
    }
    response, err := client.AddDomainName(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	增加域名成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.7.2 查询所有域名列表 - ListDomainNames

功能介绍

查询当前实例的所有域名

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listDomainNames	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/domainname

表 4-458 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-459 Query 参数

参数	是否必选	参数类型	描述
uid	否	String	域名ID
domain_name	否	String	域名

版权所有 © 华为云计算技术有限公司

请求参数

表 4-460 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-461 响应 Body 参数

参数	参数类型	描述
domain_name_infos	Array of **DomainNameInfo** objects	域名信息

表 4-462 DomainNameInfo

参数	参数类型	描述
uid	String	域名ID
domain_id	String	租户ID
project_id	String	项目ID
instance_id	String	实例ID
type	String	域名类型 取值范围： ● default：默认域名 ● custom：自定义域名
domain_name	String	域名，只能由字母、数字、中划线、星号组成， 星号只能在开头，中划线不能在开头或末 尾，至少包含两个字符串，单个字符串不 超过63个字符，字符串间以点分隔，且总 长度不超过100个字符。例如： example.com 或 *.example.com。
certificate_id	String	SCM服务的证书ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
created_at	String	增加域名的时间
updated_at	String	更新域名的时间

状态码：400

表 4-463 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-464 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-465 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-466 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-467 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/domainname

[ {
  "domain_name" : "cn-north-1.myhuaweicloud.com",
  "certificate_id" : "scs101"
} ]
```

响应示例

状态码：200

查询域名列表成功

```
{
  "domain_name_infos" : [ {
    "uid" : "f33480c6-3174-11f0-a271-0255ac1018a6",
    "domain_id" : "xxxx",
    "project_id" : "xxxx",
    "instance_id" : "f33480c6-3174-11f0-a271-0255ac1018a6",
    "domain_name" : "test.swr-pro.myhuaweicloud.com",
    "type" : "default",
    "created_at" : "2025-05-15T18:11:21.522131+08:00",
    "updated_at" : "2025-05-22T20:32:38.280566+08:00"
  }, {
    "uid" : "42892305-c7a8-4206-9b94-ad5e47aaca15",
    "domain_id" : "xxxx",
    "project_id" : "xxxx",
```


版权所有 © 华为云计算技术有限公司

```
  "instance_id" : "f33480c6-3174-11f0-a271-0255ac1018a6",
  "domain_name" : "test.cn-north-7.myhuaweicloud.com",
  "certificate_id" : "scs1730805603352",
  "type" : "custom",
  "created_at" : "2025-05-22T20:28:30.467856+08:00",
  "updated_at" : "2025-05-22T20:28:30.467857+08:00"
 } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListDomainNamesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListDomainNamesRequest request = new ListDomainNamesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListDomainNamesResponse response = client.listDomainNames(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListDomainNamesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_domain_names(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
```

 版权所有 © 华为云计算技术有限公司

```
    }

    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListDomainNamesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListDomainNames(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	查询域名列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.7.3 更新域名 - UpdateDomainName

功能介绍

更新域名

接口约束

无

版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateDomainName	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}

表 4-468 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
domainname_id	是	String	域名ID

请求参数

表 4-469 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

表 4-470 请求 Body 参数

参数	是否必选	参数类型	描述
certificate_id	是	String	SCM服务的证书ID

响应参数

状态码：200

表 4-471 响应 Body 参数

参数	参数类型	描述
domain_name_info	**DomainNameInfo** object	域名信息

表 4-472 DomainNameInfo

参数	参数类型	描述
uid	String	域名ID
domain_id	String	租户ID
project_id	String	项目ID
instance_id	String	实例ID
type	String	域名类型 取值范围： • default：默认域名 • custom：自定义域名
domain_name	String	域名，只能由字母、数字、中划线、星号组成， 星号只能在开头，中划线不能在开头或末 尾，至少包含两个字符串，单个字符串不 超过63个字符，字符串间以点分隔，且总 长度不超过100个字符。例如： example.com 或 *.example.com。
certificate_id	String	SCM服务的证书ID
created_at	String	增加域名的时间
updated_at	String	更新域名的时间

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-473 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-474 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-475 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-476 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-477 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}

{
  "domain_name" : "cn-north-1.myhuaweicloud.com",
  "certificate_id" : "scs101"
}
```

响应示例

状态码：200

更新域名成功

```
{
  "domain_name_info" : {
    "uid" : "42892305-c7a8-4206-9b94-ad5e47aaca15",
    "domain_id" : "xxxx",
    "project_id" : "xxxx",
    "instance_id" : "f33480c6-3174-11f0-a271-0255ac1018a6",
    "domain_name" : "test.myhuaweicloud.com",
    "certificate_id" : "scs17",
    "type" : "custom",
    "created_at" : "2025-05-22T20:28:30.467856+08:00",
    "updated_at" : "2025-05-22T20:28:30.467857+08:00"
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

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


版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateDomainNameSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateDomainNameRequest request = new UpdateDomainNameRequest();
        request.withInstanceId("{instance_id}");
        request.withDomainnameId("{domainname_id}");
        UpdateDomainNameRequestBody body = new UpdateDomainNameRequestBody();
        body.withCertificateId("scs101");
        request.withBody(body);
        try {
            UpdateDomainNameResponse response = client.updateDomainName(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateDomainNameRequest()
        request.instance_id = "{instance_id}"
        request.domainname_id = "{domainname_id}"
        request.body = UpdateDomainNameRequestBody(
            certificate_id="scs101"
        )
        response = client.update_domain_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

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
```

```
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateDomainNameRequest{}
    request.InstanceId = "{instance_id}"
    request.DomainnameId = "{domainname_id}"
    request.Body = &model.UpdateDomainNameRequestBody{
        CertificateId: "scs101",
    }
    response, err := client.UpdateDomainName(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	更新域名成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.7.4 删除域名 - DeleteDomainName

功能介绍

删除域名，SWR企业仓库的默认域名无法删除。

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteDomainName	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}

表 4-478 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
domainname_id	是	String	域名ID

请求参数

表 4-479 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

版权所有 © 华为云计算技术有限公司

删除域名成功

状态码：400

表 4-480 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-481 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-482 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-483 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-484 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteDomainNameSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteDomainNameRequest request = new DeleteDomainNameRequest();
        request.withInstanceId("{instance_id}");
        request.withDomainnameId("{domainname_id}");
        try {
            DeleteDomainNameResponse response = client.deleteDomainName(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteDomainNameRequest()
        request.instance_id = "{instance_id}"
        request.domainname_id = "{domainname_id}"
        response = client.delete_domain_name(request)
```

 版权所有 © 华为云计算技术有限公司

```
        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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteDomainNameRequest{}
    request.InstanceId = "{instance_id}"
    request.DomainnameId = "{domainname_id}"
    response, err := client.DeleteDomainName(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除域名成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.8 特性开关

4.8.1 查询特性开关信息 - ListFeatureGates

功能介绍

查询特性开关信息

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/instances/{instance_id}/feature-gates

版权所有 © 华为云计算技术有限公司

表 4-485 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-486 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-487 响应 Body 参数

参数	参数类型	描述
enableDomainName	Boolean	是否开启域名管理
enableCombinationRetention	Boolean	老化策略是否支持多条规则
enableArtifactScanning	Boolean	是否支持制品扫描
enableArtifactGc	Boolean	是否支持制品清理
enableRepositoryTag	Boolean	仓库是否支持标签
enableNamespaceTag	Boolean	创建命名空间时是否支持传入标签以及查询命名空间列表是否支持根据标签过滤
enableArtifactMetadata	Boolean	是否支持查询制品的Manifest信息

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-488 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-489 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-490 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-491 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-492 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/feature-gates
```

响应示例

状态码：200

查询特性开关信息成功

```
{
  "enableDomainName" : true,
  "enableCombinationRetention" : false,
  "enableArtifactScanning" : true,
  "enableArtifactGc" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListFeatureGatesSolution {
```


版权所有 © 华为云计算技术有限公司

```
    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListFeatureGatesRequest request = new ListFeatureGatesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListFeatureGatesResponse response = client.listFeatureGates(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListFeatureGatesRequest()
        request.instance_id = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
        response = client.list_feature_gates(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListFeatureGatesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListFeatureGates(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询特性开关信息成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.8.2 查询全局特性开关信息 - ListGlobalFeatureGates

功能介绍

查询全局特性开关信息

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/feature-gates

表 4-493 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-494 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-495 响应 Body 参数

参数	参数类型	描述
enableUserDefObs	Boolean	是否支持使用用户的obs桶
enableEnterprise	Boolean	是否支持支持企业项目
cerAvailable	Boolean	是否支持SWR企业版功能
enableIntranetAccessSwitch	Boolean	是否展示创建内网访问的开关按钮
enableConsoleAgency	Boolean	是否支持Console侧的一键委托
enableV5CustomAgency	Boolean	是否支持IAM新平面的信任委托
agencyName	String	委托名称

状态码：400

表 4-496 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-497 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-498 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-499 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-500 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/feature-gates
```

响应示例

状态码：200

查询全局特性开关信息成功

```
{
  "enableUserDefObs" : true,
  "enableEnterprise" : false,
  "cerAvailable" : false,
  "enableIntranetAccessSwitch" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListGlobalFeatureGatesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
```

 版权所有 © 华为云计算技术有限公司

```
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListGlobalFeatureGatesRequest request = new ListGlobalFeatureGatesRequest();
        try {
            ListGlobalFeatureGatesResponse response = client.listGlobalFeatureGates(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListGlobalFeatureGatesRequest()
        response = client.list_global_feature_gates(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListGlobalFeatureGatesRequest{}
    response, err := client.ListGlobalFeatureGates(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	查询全局特性开关信息成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.9 访问控制管理

4.9.1 新增内网访问 - CreateInstanceInternalEndpoint

功能介绍

新增内网访问

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createInternalEndpoint	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	vpcep:endpoints:create
		-	• swr:VpcId • swr:SubnetId		

URI

POST /v2/{project_id}/instances/{instance_id}/internal-endpoints

版权所有 © 华为云计算技术有限公司

表 4-501 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-502 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-503 请求 Body 参数

参数	是否必选	参数类型	描述
project_id	是	String	vpc和子网所在项目的编号
vpc_id	是	String	vpc编号ID
subnet_id	是	String	子网编号ID
description	否	String	描述

响应参数

状态码：201

表 4-504 响应 Body 参数

参数	参数类型	描述
id	String	访问控制ID

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-505 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-506 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-507 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-508 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-509 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/internal-endpoints

{
  "project_id" : "prj-12345678",
  "vpc_id" : "vpc-abc12345",
  "subnet_id" : "subnet-def56789",
  "description" : "用于内部网络访问的配置"
}
```

响应示例

状态码：201

新增内网访问成功

```
{
  "id" : "57aeff10-32c1-11f0-a5e2-0255ac1018a9"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
```


版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceInternalEndpointSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceInternalEndpointRequest request = new CreateInstanceInternalEndpointRequest();
        request.withInstanceId("{instance_id}");
        CreateIntranetAccessRequestBody body = new CreateIntranetAccessRequestBody();
        body.withDescription("用于内部网络访问的配置");
        body.withSubnetId("subnet-def56789");
        body.withVpcId("vpc-abc12345");
        body.withProjectId("prj-12345678");
        request.withBody(body);
        try {
            CreateInstanceInternalEndpointResponse response = client.createInstanceInternalEndpoint(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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}"
```

 版权所有 © 华为云计算技术有限公司

```
    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateInstanceInternalEndpointRequest()
        request.instance_id = "{instance_id}"
        request.body = CreateIntranetAccessRequestBody(
            description="用于内部网络访问的配置",
            subnet_id="subnet-def56789",
            vpc_id="vpc-abc12345",
            project_id="prj-12345678"
        )
        response = client.create_instance_internal_endpoint(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.CreateInstanceInternalEndpointRequest{}
    request.InstanceId = "{instance_id}"
    descriptionCreateIntranetAccessRequestBody:= "用于内部网络访问的配置"
    request.Body = &model.CreateIntranetAccessRequestBody{
        Description: &descriptionCreateIntranetAccessRequestBody,
        SubnetId: "subnet-def56789",
        VpcId: "vpc-abc12345",
        ProjectId: "prj-12345678",
    }
    response, err := client.CreateInstanceInternalEndpoint(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	新增内网访问成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.9.2 获取内网访问列表 - ListInstanceInternalEndpoints

功能介绍

获取内网访问列表

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listInternalEndpoints	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/internal-endpoints

表 4-510 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-511 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为100，最大值为100。**注意：offset和limit参数需要配套使用。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-512 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-513 响应 Body 参数

参数	参数类型	描述
internal_endpoints	Array of **InternalEndpoint** objects	内网访问列表
total	Integer	内网访问总数

表 4-514 InternalEndpoint

参数	参数类型	描述
id	String	内网访问ID
vpcep_endpoint_id	String	VPC端点的ID
project_id	String	项目ID
project_name	String	项目名称
vpc_id	String	VPC的ID
vpc_name	String	VPC的名称
vpc_cidr	String	VPC的CIDR范围
subnet_id	String	子网的ID
subnet_name	String	子网的名称
subnet_cidr	String	子网的CIDR范围
endpoint_ip	String	端点的IP地址
description	String	访问控制的描述信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
status	String	访问控制的状态 取值范围： ● Creating：创建中 ● Running：使用中 ● CreateError：创建错误 ● Deleting：删除中 ● DeleteError：删除错误
status_text	String	访问控制的状态信息
created_at	String	访问控制的创建时间

状态码：400

表 4-515 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-516 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-517 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-518 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-519 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/internal-endpoints
```

响应示例

状态码：200

获取内网访问列表成功

```
{
  "total" : 1,
```


版权所有 © 华为云计算技术有限公司

```
 "internal_endpoints" : [ {
   "id" : "ep-12345678",
   "vpcep_endpoint_id" : "vpcep-abc12345",
   "project_id" : "prj-abc12345",
   "project_name" : "example-project",
   "vpc_id" : "vpc-12345678",
   "vpc_name" : "example-vpc",
   "vpc_cidr" : "10.0.0.0/16",
   "subnet_id" : "subnet-5678abcd",
   "subnet_name" : "example-subnet",
   "subnet_cidr" : "10.0.1.0/24",
   "endpoint_ip" : "10.0.1.100",
   "description" : "用于内部网络通信的端点",
   "status" : "Running",
   "status_text" : "端点已成功创建并运行",
   "created_at" : "2023-10-01T12:34:56Z"
 } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceInternalEndpointsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceInternalEndpointsRequest request = new ListInstanceInternalEndpointsRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceInternalEndpointsResponse response = client.listInstanceInternalEndpoints(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceInternalEndpointsRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_internal_endpoints(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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}"
```

 版权所有 © 华为云计算技术有限公司

```
    auth, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceInternalEndpointsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceInternalEndpoints(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	获取内网访问列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.9.3 查询内网访问详情 - ShowInstanceInternalEndpoint

功能介绍

查询内网访问详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getInternalEndpoint	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}

表 4-520 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
internal_endpoints_id	是	String	内网访问ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-521 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-522 响应 Body 参数

参数	参数类型	描述
id	String	内网访问ID
vpcep_endpoint_id	String	VPC端点的ID
project_id	String	项目ID
project_name	String	项目名称
vpc_id	String	VPC的ID
vpc_name	String	VPC的名称
vpc_cidr	String	VPC的CIDR范围
subnet_id	String	子网的ID
subnet_name	String	子网的名称
subnet_cidr	String	子网的CIDR范围
endpoint_ip	String	端点的IP地址
description	String	访问控制的描述信息
status	String	访问控制的状态 取值范围： ● Creating：创建中 ● Running：使用中 ● CreateError：创建错误 ● Deleting：删除中 ● DeleteError：删除错误
status_text	String	访问控制的状态信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
created_at	String	访问控制的创建时间

状态码：400

表 4-523 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-524 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-525 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-526 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-527 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}
```

响应示例

状态码：200

查询内网访问详情成功

```
{
  "id" : "ep-12345678",
  "vpcep_endpoint_id" : "vpcep-abc12345",
  "project_id" : "prj-abc12345",
  "project_name" : "example-project",
  "vpc_id" : "vpc-12345678",
  "vpc_name" : "example-vpc",
  "vpc_cidr" : "10.0.0.0/16",
  "subnet_id" : "subnet-5678abcd",
  "subnet_name" : "example-subnet",
  "subnet_cidr" : "10.0.1.0/24",
  "endpoint_ip" : "10.0.1.100",
  "description" : "用于内部网络通信的端点",
  "status" : "Running",
  "status_text" : "端点已成功创建并运行",
  "created_at" : "2023-10-01T12:34:56Z"
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceInternalEndpointSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceInternalEndpointRequest request = new ShowInstanceInternalEndpointRequest();
        request.withInstanceId("{instance_id}");
        request.withInternalEndpointsId("{internal_endpoints_id}");
        try {
            ShowInstanceInternalEndpointResponse response = client.showInstanceInternalEndpoint(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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.
```

 版权所有 © 华为云计算技术有限公司

```
    # In this example, AK and SK are 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceInternalEndpointRequest()
        request.instance_id = "{instance_id}"
        request.internal_endpoints_id = "{internal_endpoints_id}"
        response = client.show_instance_internal_endpoint(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceInternalEndpointRequest{}
    request.InstanceId = "{instance_id}"
    request.InternalEndpointsId = "{internal_endpoints_id}"
    response, err := client.ShowInstanceInternalEndpoint(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	查询内网访问详情成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.9.4 删除内网访问 - DeleteInstanceInternalEndpoint

功能介绍

删除内网访问

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteInternalEndpoint	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	vpcep:endpoints:delete

URI

DELETE /v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}

表 4-528 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
internal_endpoints_id	是	String	内网访问ID

请求参数

表 4-529 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-530 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-531 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-532 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-533 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-534 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceInternalEndpointSolution {

    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");
```


版权所有 © 华为云计算技术有限公司

```
        String projectId = "{project_id}";

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceInternalEndpointRequest request = new DeleteInstanceInternalEndpointRequest();
        request.withInstanceId("{instance_id}");
        request.withInternalEndpointsId("{internal_endpoints_id}");
        try {
            DeleteInstanceInternalEndpointResponse response = client.deleteInstanceInternalEndpoint(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

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

 版权所有 © 华为云计算技术有限公司

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceInternalEndpointRequest{}
    request.InstanceId = "{instance_id}"
    request.InternalEndpointsId = "{internal_endpoints_id}"
    response, err := client.DeleteInstanceInternalEndpoint(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.9.5 开启或关闭公网访问 - CreateInstanceEndpointPolicy

功能介绍

开启或关闭公网访问

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateEndpointPolicyStatus	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/instances/{instance_id}/endpoint-policy

表 4-535 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-536 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-537 请求 Body 参数

参数	是否必选	参数类型	描述
enable	是	Boolean	true为开启，false为关闭。只有Disable、EnableFailed开启失败或者关闭时才能开启，只有Enable、DisableFailed时才能关闭。

响应参数

状态码：202

执行命令被接受

状态码：400

表 4-538 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-539 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-540 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-541 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-542 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/endpoint-policy

{
  "enable" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceEndpointPolicySolution {

    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);
```

 版权所有 © 华为云计算技术有限公司

```
        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceEndpointPolicyRequest request = new CreateInstanceEndpointPolicyRequest();
        request.withInstanceId("{instance_id}");
        CreateEndpointPolicyRequestBody body = new CreateEndpointPolicyRequestBody();
        body.withEnable(true);
        request.withBody(body);
        try {
            CreateInstanceEndpointPolicyResponse response = client.createInstanceEndpointPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceEndpointPolicyRequest()
        request.instance_id = "{instance_id}"
        request.body = CreateEndpointPolicyRequestBody(
            enable=True
        )
        response = client.create_instance_endpoint_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceEndpointPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.CreateEndpointPolicyRequestBody{
        Enable: true,
    }
    response, err := client.CreateInstanceEndpointPolicy(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
202	执行命令被接受
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.9.6 更新公网访问白名单 - UpdateInstanceEndpointPolicy

功能介绍

更新公网访问白名单，更新方式为全量更新方式

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateEndpointPolicy	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

PUT /v2/{project_id}/instances/{instance_id}/endpoint-policy

表 4-543 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-544 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-545 请求 Body 参数

参数	是否必选	参数类型	描述
ip_list	是	Array of **IpInfo** objects	白名单列表

表 4-546 IpInfo

参数	是否必选	参数类型	描述
ip	是	String	允许访问的ip或网段
description	否	String	描述

响应参数

状态码：200

更新公网访问白名单成功

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-547 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-548 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-549 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-550 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-551 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/endpoint-policy

{
  "ip_list" : [ {
    "ip" : "10.1.1.1/32",
    "description" : ""
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class UpdateInstanceEndpointPolicySolution {
```


版权所有 © 华为云计算技术有限公司

```
    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceEndpointPolicyRequest request = new UpdateInstanceEndpointPolicyRequest();
        request.withInstanceId("{instance_id}");
        UpdateWhiteListRequestBody body = new UpdateWhiteListRequestBody();
        List<IpInfo> listbodyIpList = new ArrayList<>();
        listbodyIpList.add(
            new IpInfo()
                .withIp("10.1.1.1/32")
                .withDescription("")
        );
        body.withIpList(listbodyIpList);
        request.withBody(body);
        try {
            UpdateInstanceEndpointPolicyResponse response = client.updateInstanceEndpointPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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)
```

 版权所有 © 华为云计算技术有限公司

```
    client = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceEndpointPolicyRequest()
        request.instance_id = "{instance_id}"
        listIpListbody = [
            IpInfo(
                ip="10.1.1.1/32",
                description=""
            )
        ]
        request.body = UpdateWhiteListRequestBody(
            ip_list=listIpListbody
        )
        response = client.update_instance_endpoint_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceEndpointPolicyRequest{}
    request.InstanceId = "{instance_id}"
    descriptionIpList:= ""
    var listIpListbody = []model.IpInfo{
        {
            Ip: "10.1.1.1/32",
            Description: &descriptionIpList,
        },
    }
    request.Body = &model.UpdateWhiteListRequestBody{
        IpList: listIpListbody,
    }
    response, err := client.UpdateInstanceEndpointPolicy(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	更新公网访问白名单成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.9.7 获取公网访问信息 - ShowInstanceEndpointPolicy

功能介绍

获取公网访问信息

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getEndpointPolicy	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/endpoint-policy

表 4-552 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-553 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-554 响应 Body 参数

参数	参数类型	描述
status	String	公网访问状态 取值范围： • Enable：启用 • Enabling：启用中 • EnableFailed：启用失败 • Disable：禁用 • Disabling：禁用中 • DisableFailed：禁用失败
ip_list	Array of **IpInfo** objects	白名单列表

表 4-555 IpInfo

参数	参数类型	描述
ip	String	允许访问的ip或网段
description	String	描述

状态码：400

表 4-556 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-557 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-558 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-559 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-560 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/endpoint-policy
```

响应示例

状态码：200

查询成功

```
{
  "ip_list" : [ {
    "ip" : "10.1.1.1/32",
    "description" : ""
  } ],
  "status" : "Enable"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceEndpointPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
```

 版权所有 © 华为云计算技术有限公司

```
                .build();
        ShowInstanceEndpointPolicyRequest request = new ShowInstanceEndpointPolicyRequest();
        request.withInstanceId("{instance_id}");
        try {
            ShowInstanceEndpointPolicyResponse response = client.showInstanceEndpointPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceEndpointPolicyRequest()
        request.instance_id = "{instance_id}"
        response = client.show_instance_endpoint_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceEndpointPolicyRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ShowInstanceEndpointPolicy(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.10 触发器管理

4.10.1 创建触发器 - CreateInstanceWebhook

功能介绍

创建触发器

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:createWebhook	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies

版权所有 © 华为云计算技术有限公司

表 4-561 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称

请求参数

表 4-562 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-563 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	触发器名称，由字母、汉字、数字、下划线（_）、中划线 (-)组成，1-256个字符。
description	否	String	触发器描述
targets	是	Array of **Target** objects	触发器目标参数
scope_rules	是	Array of **ScopeRule** objects	作用范围规则
event_types	是	Array of strings	触发器触发条件 取值范围： • PUSH_ARTIFACT：上传镜像
enabled	是	Boolean	是否启用

版权所有 © 华为云计算技术有限公司

表 4-564 Target

参数	是否必选	参数类型	描述
type	是	String	通知类型 取值范围： ● http：通过web url通知
address	是	String	触发地址，不可修改
auth_header	否	String	请求头，格式为key1:value1;key2:value2
skip_cert_verify	否	Boolean	是否跳过证书认证
address_type	是	String	触发地址类型，可选internal(内网)，internet(公网)。internal必须为内网ip形式。

表 4-565 ScopeRule

参数	是否必选	参数类型	描述
tag_selectors	是	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**RuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-566 RuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？： 匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

响应参数

状态码：201

表 4-567 响应 Body 参数

参数	参数类型	描述
id	Integer	触发器ID

状态码：400

表 4-568 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-569 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-570 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-571 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-572 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies

{
  "name" : "mywebhook",
  "description" : "my describe",
  "targets" : [ {
    "type" : "http",
    "address" : "https://100.0.0.1:80",
    "auth_header" : "param1:value1",
    "skip_cert_verify" : false,
    "address_type" : "internal"
  } ],
  "scope_rules" : [ {
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "{myrepo}"
      } ]
    },
    "repo_scope_mode" : "selection"
  } ],
  "event_types" : [ "PUSH_ARTIFACT" ],
  "enabled" : true
}
```

响应示例

状态码：201

创建触发器成功

```
{
  "id" : 34
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateInstanceWebhookSolution {
```

 版权所有 © 华为云计算技术有限公司

```
    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceWebhookRequest request = new CreateInstanceWebhookRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        CreateWebhookPolicyRequestBody body = new CreateWebhookPolicyRequestBody();
        List<String> listbodyEventTypes = new ArrayList<>();
        listbodyEventTypes.add("PUSH_ARTIFACT");
        List<RuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("{myrepo}")
        );
        Map<String, List<RuleSelector>> listScopeRulesScopeSelectors = new HashMap<>();
        listScopeRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<RuleSelector> listScopeRulesTagSelectors = new ArrayList<>();
        listScopeRulesTagSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
        );
        List<ScopeRule> listbodyScopeRules = new ArrayList<>();
        listbodyScopeRules.add(
            new ScopeRule()
                .withTagSelectors(listScopeRulesTagSelectors)
                .withScopeSelectors(listScopeRulesScopeSelectors)
                .withRepoScopeMode("selection")
        );
        List<Target> listbodyTargets = new ArrayList<>();
        listbodyTargets.add(
            new Target()
                .withType("http")
                .withAddress("https://100.0.0.1:80")
                .withAuthHeader("param1:value1")
                .withSkipCertVerify(false)
                .withAddressType("internal")
        );
        body.withEnabled(true);
        body.withEventTypes(listbodyEventTypes);
        body.withScopeRules(listbodyScopeRules);
        body.withTargets(listbodyTargets);
        body.withDescription("my describe");
        body.withName("mywebhook");
        request.withBody(body);
        try {
            CreateInstanceWebhookResponse response = client.createInstanceWebhook(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceWebhookRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        listEventTypesbody = [
            "PUSH_ARTIFACT"
        ]
        listScopeSelectorsScopeSelectors = [
            RuleSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="{myrepo}"
            )
        ]
        listScopeSelectorsScopeRules = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsScopeRules = [
            RuleSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**"
            )
        ]
        listScopeRulesbody = [
            ScopeRule(
                tag_selectors=listTagSelectorsScopeRules,
                scope_selectors=listScopeSelectorsScopeRules,
                repo_scope_mode="selection"
```

 版权所有 © 华为云计算技术有限公司

```
            )
        ]
        listTargetsbody = [
            Target(
                type="http",
                address="https://100.0.0.1:80",
                auth_header="param1:value1",
                skip_cert_verify=False,
                address_type="internal"
            )
        ]
        request.body = CreateWebhookPolicyRequestBody(
            enabled=True,
            event_types=listEventTypesbody,
            scope_rules=listScopeRulesbody,
            targets=listTargetsbody,
            description="my describe",
            name="mywebhook"
        )
        response = client.create_instance_webhook(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
```

```
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceWebhookRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    var listEventTypesbody = []string{
        "PUSH_ARTIFACT",
    }
    var listScopeSelectorsScopeSelectors = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "{myrepo}",
        },
    }
    var listScopeSelectorsScopeRules = map[string][](model.RuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsScopeRules = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    var listScopeRulesbody = []model.ScopeRule{
        {
            TagSelectors: listTagSelectorsScopeRules,
            ScopeSelectors: listScopeSelectorsScopeRules,
            RepoScopeMode: "selection",
        },
    }
    authHeaderTargets:= "param1:value1"
    skipCertVerifyTargets:= false
    var listTargetsbody = []model.Target{
        {
            Type: "http",
            Address: "https://100.0.0.1:80",
            AuthHeader: &authHeaderTargets,
            SkipCertVerify: &skipCertVerifyTargets,
            AddressType: "internal",
        },
    }
    descriptionCreateWebhookPolicyRequestBody:= "my describe"
    request.Body = &model.CreateWebhookPolicyRequestBody{
        Enabled: true,
        EventTypes: listEventTypesbody,
        ScopeRules: listScopeRulesbody,
        Targets: listTargetsbody,
        Description: &descriptionCreateWebhookPolicyRequestBody,
        Name: "mywebhook",
    }
    response, err := client.CreateInstanceWebhook(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
201	创建触发器成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.10.2 获取触发器列表 - ListInstanceWebhooks

功能介绍

获取触发器列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listWebhooks	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/webhook/policies

表 4-573 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-574 Query 参数

参数	是否必选	参数类型	描述
order_column	否	String	排序字段，支持created_at、updated_at、name，默认为created_at
order_type	否	String	排序方式，支持desc、asc，默认desc;order_type需要与order_column配合使用
name	否	String	策略名称，模糊查询
namespace_id	否	Integer	所属命名空间ID
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-575 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-576 响应 Body 参数

参数	参数类型	描述
policies	Array of **WebhookPolicyDetail** objects	触发器列表
total	Integer	触发器总数

表 4-577 WebhookPolicyDetail

参数	参数类型	描述
id	Integer	触发器ID
name	String	触发器名称
description	String	触发器描述
targets	Array of **Target** objects	触发目标
event_types	Array of strings	触发器触发条件 取值范围： • PUSH_ARTIFACT：上传镜像
enabled	Boolean	是否启用，可选true或false
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名
creator	String	创建者
created_at	String	创建时间
updated_at	String	更新时间
scope_rules	Array of **ScopeRule** objects	触发作用范围规则

版权所有 © 华为云计算技术有限公司

表 4-578 Target

参数	参数类型	描述
type	String	通知类型 取值范围： ● http：通过web url通知
address	String	触发地址，不可修改
auth_header	String	请求头，格式为key1:value1;key2:value2
skip_cert_verify	Boolean	是否跳过证书认证
address_type	String	触发地址类型，可选internal(内网)，internet(公网)。internal必须为内网ip形式。

表 4-579 ScopeRule

参数	参数类型	描述
tag_selectors	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<**RuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-580 RuleSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
extras	String	预留字段

状态码：400

表 4-581 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-582 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-583 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-584 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-585 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/webhook/policies
```

响应示例

状态码：200

查询触发器列表成功

```
{
  "total" : 1,
  "policies" : [ {
    "id" : 39,
    "name" : "te3st123",
    "description" : "",
    "namespace_id" : 1,
    "targets" : [ {
      "type" : "http",
      "address" : "https://baidu.com",
      "auth_header" : "key:value",
      "skip_cert_verify" : false,
      "address_type" : "internet"
    } ],
    "event_types" : [ "PUSH_ARTIFACT" ],
    "creator" : "admin",
    "created_at" : "2021-02-18T12:05:36.218939Z",
    "updated_at" : "2021-02-18T12:05:36.218939Z",
    "enabled" : true,
    "namespace_name" : "library",
    "scope_rules" : [ {
      "repo_scope_mode" : "regular",
```


版权所有 © 华为云计算技术有限公司

```
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : ""
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**",
        "extras" : ""
      } ]
    }
  } ]
 } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceWebhooksSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceWebhooksRequest request = new ListInstanceWebhooksRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceWebhooksResponse response = client.listInstanceWebhooks(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceWebhooksRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_webhooks(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
```

 版权所有 © 华为云计算技术有限公司

```
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceWebhooksRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceWebhooks(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	查询触发器列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.10.3 获取触发器详情 - ShowInstanceWebhook

功能介绍

获取触发器详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getWebhook	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}

表 4-586 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	触发器ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-587 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-588 响应 Body 参数

参数	参数类型	描述
id	Integer	触发器ID
name	String	触发器名称
description	String	触发器描述
targets	Array of **Target** objects	触发目标
event_types	Array of strings	触发器触发条件 取值范围： ● PUSH_ARTIFACT：上传镜像
enabled	Boolean	是否启用，可选true或false
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名
creator	String	创建者
created_at	String	创建时间
updated_at	String	更新时间
scope_rules	Array of **ScopeRule** objects	触发作用范围规则

版权所有 © 华为云计算技术有限公司

表 4-589 Target

参数	参数类型	描述
type	String	通知类型 取值范围： ● http：通过web url通知
address	String	触发地址，不可修改
auth_header	String	请求头，格式为key1:value1;key2:value2
skip_cert_verify	Boolean	是否跳过证书认证
address_type	String	触发地址类型，可选internal(内网)，internet(公网)。internal必须为内网ip形式。

表 4-590 ScopeRule

参数	参数类型	描述
tag_selectors	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<**RuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-591 RuleSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
extras	String	预留字段

状态码：400

表 4-592 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-593 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-594 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-595 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-596 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}
```

响应示例

状态码：200

查询触发器详情成功

```
{
  "id" : 39,
  "name" : "te3st123",
  "description" : "",
  "namespace_id" : 1,
  "targets" : [ {
    "type" : "http",
    "address" : "https://baidu.com",
    "auth_header" : "key:value",
    "skip_cert_verify" : false,
    "address_type" : "internet"
  } ],
  "event_types" : [ "PUSH_ARTIFACT" ],
  "creator" : "admin",
  "created_at" : "2021-02-18T12:05:36.218939Z",
  "updated_at" : "2021-02-18T12:05:36.218939Z",
  "enabled" : true,
  "namespace_name" : "library",
  "scope_rules" : [ {
    "repo_scope_mode" : "regular",
    "tag_selectors" : [ {
```

 版权所有 © 华为云计算技术有限公司

```
    "kind" : "doublestar",
    "decoration" : "matches",
    "pattern" : "**",
    "extras" : ""
   } ],
   "scope_selectors" : {
    "repository" : [ {
     "kind" : "doublestar",
     "decoration" : "repoMatches",
     "pattern" : "**",
     "extras" : ""
    } ]
   }
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceWebhookSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceWebhookRequest request = new ShowInstanceWebhookRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ShowInstanceWebhookResponse response = client.showInstanceWebhook(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceWebhookRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.show_instance_webhook(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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}"
```

 版权所有 © 华为云计算技术有限公司

```
    auth, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceWebhookRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ShowInstanceWebhook(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	查询触发器详情成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.10.4 修改触发器 - UpdateInstanceWebhook

功能介绍

修改触发器

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateWebhook	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}

表 4-597 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	触发器ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-598 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-599 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	触发器名称，由字母、汉字、数字、下划线（_）、中划线 (-)组成，1-256个字符。
description	否	String	触发器描述
targets	是	Array of **Target** objects	触发器目标参数
scope_rules	是	Array of **ScopeRule** objects	作用范围规则
event_types	否	Array of strings	触发器触发条件 取值范围： • PUSH_ARTIFACT：上传镜像
enabled	是	Boolean	是否启用

表 4-600 Target

参数	是否必选	参数类型	描述
type	是	String	通知类型 取值范围： • http：通过web url通知
address	是	String	触发地址，不可修改
auth_header	否	String	请求头，格式为key1:value1;key2:value2
skip_cert_verify	否	Boolean	是否跳过证书认证

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
address_type	是	String	触发地址类型，可选internal(内网)，internet(公网)。internal必须为内网ip形式。

表 4-601 ScopeRule

参数	是否必选	参数类型	描述
tag_selectors	是	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**RuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-602 RuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

成功更新触发器信息

状态码：400

表 4-603 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-604 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-605 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-606 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-607 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}

{
  "event_types" : [ "PUSH_ARTIFACT" ],
  "targets" : [ {
    "type" : "http",
    "address" : "https://test.com",
    "auth_header" : "xxxxx",
    "skip_cert_verify" : false,
    "address_type" : "internet"
  } ],
  "name" : "test111",
  "description" : "xxxx",
  "scope_rules" : [ {
    "repo_scope_mode" : "regular",
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    }
  } ],
  "enabled" : true
}
```


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateInstanceWebhookSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceWebhookRequest request = new UpdateInstanceWebhookRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        UpdateWebhookPolicyRequestBody body = new UpdateWebhookPolicyRequestBody();
        List<String> listbodyEventTypes = new ArrayList<>();
        listbodyEventTypes.add("PUSH_ARTIFACT");
        List<RuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<RuleSelector>> listScopeRulesScopeSelectors = new HashMap<>();
        listScopeRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<RuleSelector> listScopeRulesTagSelectors = new ArrayList<>();
        listScopeRulesTagSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
```

 版权所有 © 华为云计算技术有限公司

```
                .withDecoration("matches")
                .withPattern("**")
        );
        List<ScopeRule> listbodyScopeRules = new ArrayList<>();
        listbodyScopeRules.add(
            new ScopeRule()
                .withTagSelectors(listScopeRulesTagSelectors)
                .withScopeSelectors(listScopeRulesScopeSelectors)
                .withRepoScopeMode("regular")
        );
        List<Target> listbodyTargets = new ArrayList<>();
        listbodyTargets.add(
            new Target()
                .withType("http")
                .withAddress("https://test.com")
                .withAuthHeader("xxxxx")
                .withSkipCertVerify(false)
                .withAddressType("internet")
        );
        body.withEnabled(true);
        body.withEventTypes(listbodyEventTypes);
        body.withScopeRules(listbodyScopeRules);
        body.withTargets(listbodyTargets);
        body.withDescription("xxxx");
        body.withName("test111");
        request.withBody(body);
        try {
            UpdateInstanceWebhookResponse response = client.updateInstanceWebhook(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()
```

 版权所有 © 华为云计算技术有限公司

```
try:
    request = UpdateInstanceWebhookRequest()
    request.instance_id = "{instance_id}"
    request.namespace_name = "{namespace_name}"
    request.policy_id = {policy_id}
    listEventTypesbody = [
        "PUSH_ARTIFACT"
    ]
    listScopeSelectorsScopeSelectors = [
        RuleSelector(
            kind="doublestar",
            decoration="repoMatches",
            pattern="**"
        )
    ]
    listScopeSelectorsScopeRules = {
        "repository": listScopeSelectorsScopeSelectors
    }
    listTagSelectorsScopeRules = [
        RuleSelector(
            kind="doublestar",
            decoration="matches",
            pattern="**"
        )
    ]
    listScopeRulesbody = [
        ScopeRule(
            tag_selectors=listTagSelectorsScopeRules,
            scope_selectors=listScopeSelectorsScopeRules,
            repo_scope_mode="regular"
        )
    ]
    listTargetsbody = [
        Target(
            type="http",
            address="https://test.com",
            auth_header="xxxxx",
            skip_cert_verify=False,
            address_type="internet"
        )
    ]
    request.body = UpdateWebhookPolicyRequestBody(
        enabled=True,
        event_types=listEventTypesbody,
        scope_rules=listScopeRulesbody,
        targets=listTargetsbody,
        description="xxxx",
        name="test111"
    )
    response = client.update_instance_webhook(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceWebhookRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    var listEventTypesbody = []string{
        "PUSH_ARTIFACT",
    }
    var listScopeSelectorsScopeSelectors = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsScopeRules = map[string][](model.RuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsScopeRules = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    var listScopeRulesbody = []model.ScopeRule{
        {
            TagSelectors: listTagSelectorsScopeRules,
            ScopeSelectors: listScopeSelectorsScopeRules,
            RepoScopeMode: "regular",
        },
    }
    authHeaderTargets:= "xxxxx"
    skipCertVerifyTargets:= false
    var listTargetsbody = []model.Target{
```

 版权所有 © 华为云计算技术有限公司

```
        {
            Type: "http",
            Address: "https://test.com",
            AuthHeader: &authHeaderTargets,
            SkipCertVerify: &skipCertVerifyTargets,
            AddressType: "internet",
        },
    }
    descriptionUpdateWebhookPolicyRequestBody:= "xxxx"
    request.Body = &model.UpdateWebhookPolicyRequestBody{
        Enabled: true,
        EventTypes: &listEventTypesbody,
        ScopeRules: listScopeRulesbody,
        Targets: listTargetsbody,
        Description: &descriptionUpdateWebhookPolicyRequestBody,
        Name: "test111",
    }
    response, err := client.UpdateInstanceWebhook(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	成功更新触发器信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.10.5 删除触发器 - DeleteInstanceWebhook

功能介绍

删除触发器

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteWebhook	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}

表 4-608 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	触发器ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-609 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

表 4-610 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-611 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-612 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-613 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-614 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}
```

响应示例

无

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceWebhookSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceWebhookRequest request = new DeleteInstanceWebhookRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            DeleteInstanceWebhookResponse response = client.deleteInstanceWebhook(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceWebhookRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.delete_instance_webhook(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceWebhookRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.DeleteInstanceWebhook(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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.10.6 获取触发器执行任务列表 - ListInstanceWebhookJobs

功能介绍

获取触发器执行任务列表

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listWebhookJobs	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}/jobs

表 4-615 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	触发器ID

表 4-616 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
status	否	String	状态 取值范围： • Initialized：已初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止

请求参数

表 4-617 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-618 响应 Body 参数

参数	参数类型	描述
jobs	Array of **Job** objects	触发器执行任务列表
total	Integer	触发器执行任务总数

表 4-619 Job

参数	参数类型	描述
id	Integer	任务ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
policy_id	Integer	触发器ID
event_type	String	事件类型 取值范围： ● PUSH_ARTIFACT：上传镜像
notify_type	String	通知类型 取值范围： ● http：通过web url通知
status	String	任务状态 取值范围： ● Pending：准备中 ● InProgress：运行中 ● Succeed：运行成功 ● Failed：运行失败
status_text	String	状态原因
job_detail	String	任务详情
created_at	String	创建时间
updated_at	String	更新时间

状态码：400

表 4-620 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-621 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-622 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-623 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-624 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/
policies/{policy_id}/jobs
```

响应示例

状态码：200

查询触发器执行任务列表成功

```
{
  "total" : 2,
  "jobs" : [ {
    "id" : 30173,
    "policy_id" : 16,
    "event_type" : "PUSH_ARTIFACT",
    "notify_type" : "http",
    "status" : "InProgress",
    "job_detail" : "",
    "created_at" : "2020-12-02T00:28:06.145239Z",
    "updated_at" : "2020-12-04T09:16:56.339681Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceWebhookJobsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceWebhookJobsRequest request = new ListInstanceWebhookJobsRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
```

 版权所有 © 华为云计算技术有限公司

```
        request.withPolicyId({policy_id});
        try {
            ListInstanceWebhookJobsResponse response = client.listInstanceWebhookJobs(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceWebhookJobsRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.list_instance_webhook_jobs(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceWebhookJobsRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ListInstanceWebhookJobs(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	查询触发器执行任务列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源

 版权所有 © 华为云计算技术有限公司

状态码	描述
500	内部错误

错误码

请参见**错误码**。

4.11 任务管理

4.11.1 获取任务列表 - ListInstanceJobs

功能介绍

获取任务列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listJobs	List	instance *	-	-	-

URI

GET /v2/{project_id}/jobs

版权所有 © 华为云计算技术有限公司

表 4-625 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

表 4-626 Query 参数

参数	是否必选	参数类型	描述
status	否	String	任务状态 取值范围： ● Creating：创建中 ● Initializing：初始化 ● Running：运行中 ● Failed：失败 ● Success：成功
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用。**
limit	否	Integer	返回条数，默认为100，最大值为100。**注意：offset和limit参数需要配套使用。**

请求参数

表 4-627 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-628 响应 Body 参数

参数	参数类型	描述
jobs	Array of **JobDetail** objects	任务列表
total	Integer	任务总数

表 4-629 JobDetail

参数	参数类型	描述
id	String	任务ID
instance_id	String	相关实例ID
resource_id	String	资源ID
resource_name	String	资源名称
type	String	类型 取值范围： • CreateInstance：创建实例
status	String	状态 取值范围： • Initializing：初始化 • Creating：创建中 • Running：运行中 • Failed：失败 • Success：成功
reason	String	失败原因
created_at	String	实例创建时间
updated_at	String	实例更新时间

状态码：400

表 4-630 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-631 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-632 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-633 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-634 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/jobs
```

响应示例

状态码：200

查询任务列表成功

```
{
  "total" : 1,
  "jobs" : [ {
    "id" : "36db69c5-b2d5-11eb-b044-0255ac101219",
    "instance_id" : "36ce7f50-b2d5-11eb-b044-0255ac101219",
    "resource_id" : "36ce7f50-b2d5-11eb-b044-0255ac101219",
    "resource_name" : "test-public",
    "type" : "CreateInstance",
    "status" : "Failed",
    "reason" : "failed to select cce cluster",
    "created_at" : "2021-05-12T03:50:38.305978Z",
    "updated_at" : "2021-05-12T03:50:43.315473Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceJobsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceJobsRequest request = new ListInstanceJobsRequest();
        try {
            ListInstanceJobsResponse response = client.listInstanceJobs(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceJobsRequest()
        response = client.list_instance_jobs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceJobsRequest{}
    response, err := client.ListInstanceJobs(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询任务列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.11.2 获取任务详情 - ShowInstanceJob

功能介绍

获取任务详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

URI

GET /v2/{project_id}/jobs/{job_id}

表 4-635 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
job_id	是	String	任务ID

请求参数

表 4-636 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-637 响应 Body 参数

参数	参数类型	描述
id	String	任务ID
instance_id	String	相关实例ID
resource_id	String	资源ID
resource_name	String	资源名称
type	String	类型 取值范围： ● CreateInstance：创建实例
status	String	状态 取值范围： ● Initializing：初始化 ● Creating：创建中 ● Running：运行中 ● Failed：失败 ● Success：成功
reason	String	失败原因
created_at	String	实例创建时间
updated_at	String	实例更新时间

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-638 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-639 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-640 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-641 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-642 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/jobs/{job_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : "36db69c5-b2d5-11eb-b044-0255ac101219",
  "instance_id" : "36ce7f50-b2d5-11eb-b044-0255ac101219",
  "resource_id" : "36ce7f50-b2d5-11eb-b044-0255ac101219",
  "resource_name" : "test-public",
  "type" : "CreateInstance",
  "status" : "Failed",
  "reason" : "failed to select cce cluster",
  "created_at" : "2021-05-12T03:50:38.305978Z",
  "updated_at" : "2021-05-12T03:50:43.315473Z"
}
```

SDK 代码示例

SDK代码示例如下。

Java

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


版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceJobSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceJobRequest request = new ShowInstanceJobRequest();
        request.withJobId("{job_id}");
        try {
            ShowInstanceJobResponse response = client.showInstanceJob(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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}"
```

 版权所有 © 华为云计算技术有限公司

```
credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = ShowInstanceJobRequest()
    request.job_id = "{job_id}"
    response = client.show_instance_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceJobRequest{}
    request.JobId = "{job_id}"
    response, err := client.ShowInstanceJob(request)
    if err == nil {
```

 版权所有 © 华为云计算技术有限公司

```
        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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.11.3 删除任务 - DeleteInstanceJob

功能介绍

删除任务

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

 版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteJob	Write	instance *	-	-	-

URI

DELETE /v2/{project_id}/jobs/{job_id}

表 4-643 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
job_id	是	String	任务ID

请求参数

表 4-644 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

表 4-645 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-646 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-647 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-648 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

版权所有 © 华为云计算技术有限公司

表 4-649 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/jobs/{job_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceJobSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceJobRequest request = new DeleteInstanceJobRequest();
        request.withJobId("{job_id}");
        try {
```

 版权所有 © 华为云计算技术有限公司

```
            DeleteInstanceJobResponse response = client.deleteInstanceJob(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceJobRequest()
        request.job_id = "{job_id}"
        response = client.delete_instance_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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
```

 版权所有 © 华为云计算技术有限公司

```
variables and decrypted during use to ensure security.
    // In this example, AK and SK are 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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceJobRequest{}
    request.JobId = "{job_id}"
    response, err := client.DeleteInstanceJob(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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.12 镜像同步管理

4.12.1 创建镜像同步的目标仓库 - CreateInstanceRegistry

功能介绍

创建镜像同步的目标仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createRegistry	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/registries

表 4-650 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
instance_id	是	String	企业仓库实例ID

请求参数

表 4-651 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-652 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	目标仓库名称，1-64字符组成，只能包含英文大小写、数字、汉字、中划线和下划线。
description	否	String	目标仓库描述
type	是	String	仓库类型 取值范围： • swr-pro：开源harbor仓库 • swr-pro-internal：SWR企业版仓库 • huawei-SWR：SWR基础版仓库
project_id	否	String	企业仓库实例所属的项目ID，当type为swr-pro-internal时必填
region_id	否	String	区域ID，当type为swr-pro-internal时必填
instance_id	否	String	企业仓库实例ID，当type为swr-pro-internal时必填
url	是	String	目标仓库的地址
credential	是	**Credential** object	仓库认证信息

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
dns_conf	否	DnsConf object	域名解析的配置信息，当仓库类型为开源harbor仓库时，是必选项。
insecure	是	Boolean	是否验证远程证书，目标仓库选择SWR企业版仓库时仅支持填true。

表 4-653 Credential

参数	是否必选	参数类型	描述
type	是	String	认证方式 取值范围： • basic Basic认证方式
access_key	是	String	访问ID
access_secret	是	String	访问密码

表 4-654 DnsConf

参数	是否必选	参数类型	描述
hosts	否	Map<String,String>	主机映射，键为域名，值为对应的 IP 地址

响应参数

状态码：201

表 4-655 响应 Body 参数

参数	参数类型	描述
id	Integer	命名空间ID

状态码：400

表 4-656 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-657 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-658 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-659 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-660 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

- 创建目标仓库，仓库类型为huawei-SWR(SWR基础版仓库)

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries

{
  "name" : "SWR",
  "description" : "swr.cn-north-4.myhuaweicloud.com",
  "type" : "huawei-SWR",
  "url" : "https://swr.cn-north-4.myhuaweicloud.com",
  "credential" : {
    "type" : "basic",
    "access_key" : "access_key",
    "access_secret" : "access_secret"
  },
  "insecure" : true
}
```

- 创建目标仓库，仓库类型为swr-pro-internal(SWR企业版仓库)

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries

{
  "name" : "swr-pro-internal",
  "type" : "swr-pro-internal",
  "project_id" : "project_id",
  "region_id" : "cn-north-7",
  "instance_id" : "instance_id",
  "url" : "https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
  "credential" : {
    "type" : "basic",
    "access_key" : "access_key",
    "access_secret" : "access_secret"
  },
  "insecure" : true
}
```

- 创建目标仓库，仓库类型为swr-pro(开源harbor仓库)

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries

{
  "name" : "swr-pro",
  "type" : "swr-pro",
  "project_id" : "project_id",
  "region_id" : "cn-north-7",
  "instance_id" : "instance_id",
  "url" : "https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
  "credential" : {
    "type" : "basic",
    "access_key" : "access_key",
```


版权所有 © 华为云计算技术有限公司

```
    "access_secret" : "access_secret"
  },
  "dns_conf" : {
    "hosts" : {
      "test241102-f6ktf2.swr-pro.myhuaweicloud.com" : "192.168.14.14"
    }
  },
  "insecure" : true
}
```

响应示例

状态码：201

创建目标仓库成功

```
{
  "id" : 34
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 创建目标仓库，仓库类型为huawei-SWR(SWR基础版仓库)

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceRegistrySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRegistryRequest request = new CreateInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        CreateRegistryRequestBody body = new CreateRegistryRequestBody();
        Credential credentialbody = new Credential();
        credentialbody.withType("basic")
```

 版权所有 © 华为云计算技术有限公司

```
            .withAccessKey("access_key")
            .withAccessSecret("access_secret");
        body.withInsecure(true);
        body.withCredential(credentialbody);
        body.withUrl("https://swr.cn-north-4.myhuaweicloud.com");
        body.withType("huawei-SWR");
        body.withDescription("swr.cn-north-4.myhuaweicloud.com");
        body.withName("SWR");
        request.withBody(body);
        try {
            CreateInstanceRegistryResponse response = client.createInstanceRegistry(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());
        }
    }
}
```

- 创建目标仓库，仓库类型为swr-pro-internal(SWR企业版仓库)

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class CreateInstanceRegistrySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRegistryRequest request = new CreateInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        CreateRegistryRequestBody body = new CreateRegistryRequestBody();
        Credential credentialbody = new Credential();
        credentialbody.withType("basic")
            .withAccessKey("access_key")
            .withAccessSecret("access_secret");
        body.withInsecure(true);
        body.withCredential(credentialbody);
```

```
        body.withUrl("https://test241102-f6ktf2.swr-pro.myhuaweicloud.com");
        body.withInstanceId("instance_id");
        body.withRegionId("cn-north-7");
        body.withProjectId("project_id");
        body.withType("swr-pro-internal");
        body.withName("swr-pro-internal");
        request.withBody(body);
        try {
            CreateInstanceRegistryResponse response = client.createInstanceRegistry(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());
        }
    }
}
```

- 创建目标仓库，仓库类型为swr-pro(开源harbor仓库)

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class CreateInstanceRegistrySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRegistryRequest request = new CreateInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        CreateRegistryRequestBody body = new CreateRegistryRequestBody();
        Map<String, String> listDnsConfHosts = new HashMap<>();
        listDnsConfHosts.put("test241102-f6ktf2.swr-pro.myhuaweicloud.com", "192.168.14.14");
        DnsConf dnsConfbody = new DnsConf();
        dnsConfbody.withHosts(listDnsConfHosts);
        Credential credentialbody = new Credential();
        credentialbody.withType("basic")
```

 版权所有 © 华为云计算技术有限公司

```
            .withAccessKey("access_key")
            .withAccessSecret("access_secret");
        body.withInsecure(true);
        body.withDnsConf(dnsConfbody);
        body.withCredential(credentialbody);
        body.withUrl("https://test241102-f6ktf2.swr-pro.myhuaweicloud.com");
        body.withInstanceId("instance_id");
        body.withRegionId("cn-north-7");
        body.withProjectId("project_id");
        body.withType("swr-pro");
        body.withName("swr-pro");
        request.withBody(body);
        try {
            CreateInstanceRegistryResponse response = client.createInstanceRegistry(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

- 创建目标仓库，仓库类型为huawei-SWR(SWR基础版仓库)

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        credentialbody = Credential(
            type="basic",
            access_key="access_key",
            access_secret="access_secret"
        )
        request.body = CreateRegistryRequestBody(
            insecure=True,
            credential=credentialbody,
```

 版权所有 © 华为云计算技术有限公司

```
            url="https://swr.cn-north-4.myhuaweicloud.com",
            type="huawei-SWR",
            description="swr.cn-north-4.myhuaweicloud.com",
            name="SWR"
        )
        response = client.create_instance_registry(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 创建目标仓库，仓库类型为swr-pro-internal(SWR企业版仓库)

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        credentialbody = Credential(
            type="basic",
            access_key="access_key",
            access_secret="access_secret"
        )
        request.body = CreateRegistryRequestBody(
            insecure=True,
            credential=credentialbody,
            url="https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
            instance_id="instance_id",
            region_id="cn-north-7",
            project_id="project_id",
            type="swr-pro-internal",
            name="swr-pro-internal"
        )
        response = client.create_instance_registry(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 创建目标仓库，仓库类型为swr-pro(开源harbor仓库)

```
# coding: utf-8

import os
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        listHostsDnsConf = {
            "test241102-f6ktf2.swr-pro.myhuaweicloud.com": "192.168.14.14"
        }
        dnsConfbody = DnsConf(
            hosts=listHostsDnsConf
        )
        credentialbody = Credential(
            type="basic",
            access_key="access_key",
            access_secret="access_secret"
        )
        request.body = CreateRegistryRequestBody(
            insecure=True,
            dns_conf=dnsConfbody,
            credential=credentialbody,
            url="https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
            instance_id="instance_id",
            region_id="cn-north-7",
            project_id="project_id",
            type="swr-pro",
            name="swr-pro"
        )
        response = client.create_instance_registry(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

- 创建目标仓库，仓库类型为huawei-SWR(SWR基础版仓库)

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRegistryRequest{}
    request.InstanceId = "{instance_id}"
    credentialbody := &model.Credential{
        Type: "basic",
        AccessKey: "access_key",
        AccessSecret: "access_secret",
    }
    descriptionCreateRegistryRequestBody:= "swr.cn-north-4.myhuaweicloud.com"
    request.Body = &model.CreateRegistryRequestBody{
        Insecure: true,
        Credential: credentialbody,
        Url: "https://swr.cn-north-4.myhuaweicloud.com",
        Type: "huawei-SWR",
        Description: &descriptionCreateRegistryRequestBody,
        Name: "SWR",
    }
    response, err := client.CreateInstanceRegistry(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建目标仓库，仓库类型为swr-pro-internal(SWR企业版仓库)

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRegistryRequest{}
    request.InstanceId = "{instance_id}"
    credentialbody := &model.Credential{
        Type: "basic",
        AccessKey: "access_key",
        AccessSecret: "access_secret",
    }
    instanceIdCreateRegistryRequestBody:= "instance_id"
    regionIdCreateRegistryRequestBody:= "cn-north-7"
    projectIdCreateRegistryRequestBody:= "project_id"
    request.Body = &model.CreateRegistryRequestBody{
        Insecure: true,
        Credential: credentialbody,
        Url: "https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
        InstanceId: &instanceIdCreateRegistryRequestBody,
        RegionId: &regionIdCreateRegistryRequestBody,
        ProjectId: &projectIdCreateRegistryRequestBody,
        Type: "swr-pro-internal",
        Name: "swr-pro-internal",
    }
    response, err := client.CreateInstanceRegistry(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建目标仓库，仓库类型为swr-pro(开源harbor仓库)

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
```

 版权所有 © 华为云计算技术有限公司

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRegistryRequest{}
    request.InstanceId = "{instance_id}"
    var listHostsDnsConf = map[string]string{
        "test241102-f6ktf2.swr-pro.myhuaweicloud.com": "192.168.14.14",
    }
    dnsConfbody := &model.DnsConf{
        Hosts: listHostsDnsConf,
    }
    credentialbody := &model.Credential{
        Type: "basic",
        AccessKey: "access_key",
        AccessSecret: "access_secret",
    }
    instanceIdCreateRegistryRequestBody:= "instance_id"
    regionIdCreateRegistryRequestBody:= "cn-north-7"
    projectIdCreateRegistryRequestBody:= "project_id"
    request.Body = &model.CreateRegistryRequestBody{
        Insecure: true,
        DnsConf: dnsConfbody,
        Credential: credentialbody,
        Url: "https://test241102-f6ktf2.swr-pro.myhuaweicloud.com",
        InstanceId: &instanceIdCreateRegistryRequestBody,
        RegionId: &regionIdCreateRegistryRequestBody,
        ProjectId: &projectIdCreateRegistryRequestBody,
        Type: "swr-pro",
        Name: "swr-pro",
    }
    response, err := client.CreateInstanceRegistry(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```

 版权所有 © 华为云计算技术有限公司

```
    } 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	创建目标仓库成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.2 获取镜像同步的目标仓库列表 - ListInstanceRegistries

功能介绍

获取镜像同步的目标仓库列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listRegistries	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/registries

表 4-661 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-662 Query 参数

参数	是否必选	参数类型	描述
order_column	否	String	排序字段，支持created_at、updated_at、name，默认为created_at
order_type	否	String	排序方式，支持desc、asc，默认desc; 注意：order_type需要与order_column配合使用
name	否	String	名称，模糊查询
type	否	String	仓库类型 取值范围： • swr-pro：开源harbor仓库 • swr-pro-internal：SWR企业版仓库 • huawei-SWR：SWR基础版仓库
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-663 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-664 响应 Body 参数

参数	参数类型	描述
registries	Array of **Registry** objects	同步仓库列表
total	Integer	同步仓库总数

表 4-665 Registry

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
description	String	仓库描述
region_id	String	区域ID
instance_id	String	实例ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
type	String	仓库类型 取值范围： ● swr-pro：开源harbor仓库 ● swr-pro-internal：SWR企业版仓库 ● huawei-SWR：SWR基础版仓库
url	String	仓库地址
credential	**Credential** object	仓库认证信息
insecure	Boolean	是否验证远程证书
status	String	仓库状态 取值范围： ● healthy：健康 ● unhealthy：不健康
created_at	String	创建时间
updated_at	String	更新时间

表 4-666 Credential

参数	参数类型	描述
type	String	认证方式 取值范围： ● basic Basic认证方式
access_key	String	访问ID
access_secret	String	访问密码

状态码：400

表 4-667 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-668 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-669 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-670 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-671 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries
```

响应示例

状态码：200

查询镜像同步的目标仓库列表成功

```
{
  "total" : 1,
  "registries" : [ {
    "id" : 24,
    "name" : "test-swr",
    "description" : "",
    "type" : "huawei-SWR",
    "url" : "https://10.185.79.188",
    "credential" : {
      "type" : "basic",
      "access_key" : "cn-cmcc1@R5172VW6FEW6MZFP2W1S",
      "access_secret" : ""
    },
    "insecure" : true,
    "status" : "healthy",
    "created_at" : "2021-02-05T09:31:00.298426Z",
    "updated_at" : "2021-02-19T12:19:51.912632Z",
    "region_id" : "",
    "instance_id" : ""
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRegistriesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRegistriesRequest request = new ListInstanceRegistriesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceRegistriesResponse response = client.listInstanceRegistries(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRegistriesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_registries(request)
        print(response)
    except exceptions.ClientRequestException as e:
```

 版权所有 © 华为云计算技术有限公司

```
        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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRegistriesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceRegistries(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询镜像同步的目标仓库列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.3 获取镜像同步的目标仓库详情 - ShowInstanceRegistry

功能介绍

获取镜像同步的目标仓库详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getRegistry	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/registries/{registry_id}

表 4-672 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
registry_id	是	Integer	同步仓库ID

请求参数

表 4-673 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-674 响应 Body 参数

参数	参数类型	描述
id	Integer	仓库ID
name	String	仓库名称
description	String	仓库描述
region_id	String	区域ID
instance_id	String	实例ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
type	String	仓库类型 取值范围： ● swr-pro：开源harbor仓库 ● swr-pro-internal：SWR企业版仓库 ● huawei-SWR：SWR基础版仓库
url	String	仓库地址
credential	**Credential** object	仓库认证信息
insecure	Boolean	是否验证远程证书
status	String	仓库状态 取值范围： ● healthy：健康 ● unhealthy：不健康
created_at	String	创建时间
updated_at	String	更新时间

表 4-675 Credential

参数	参数类型	描述
type	String	认证方式 取值范围： ● basic Basic认证方式
access_key	String	访问ID
access_secret	String	访问密码

状态码：400

表 4-676 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-677 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-678 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-679 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-680 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries/{registry_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : 5,
  "name" : "test",
  "description" : "",
  "type" : "swr-pro",
  "url" : "http://192.168.0.123:30000",
  "region_id" : "cn-north-7",
  "instance_id" : "abc",
  "credential" : {
    "type" : "basic",
    "access_key" : "admin",
    "access_secret" : "*****"
  },
  "insecure" : true,
  "status" : "healthy",
  "created_at" : "2020-10-16T08:36:06.973474Z",
  "updated_at" : "2020-12-03T11:23:39.106998Z"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceRegistrySolution {

    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
```

 版权所有 © 华为云计算技术有限公司

```
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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceRegistryRequest request = new ShowInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        request.withRegistryId({registry_id});
        try {
            ShowInstanceRegistryResponse response = client.showInstanceRegistry(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        request.registry_id = {registry_id}
        response = client.show_instance_registry(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
```

 版权所有 © 华为云计算技术有限公司

```
        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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceRegistryRequest{}
    request.InstanceId = "{instance_id}"
    request.RegistryId = int32({registry_id})
    response, err := client.ShowInstanceRegistry(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.4 修改镜像同步的目标仓库 - UpdateInstanceRegistry

功能介绍

修改镜像同步的目标仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateRegistry	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

PUT /v2/{project_id}/instances/{instance_id}/registries/{registry_id}

表 4-681 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
registry_id	是	Integer	同步仓库ID

请求参数

表 4-682 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-683 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	目标仓库名称，1-64字符组成，只能包含英文大小写、数字、汉字、中划线和下划线。
description	否	String	目标仓库描述
type	是	String	仓库类型，swr-pro(开源harbor仓库)、swr-pro-internal(SWR企业版仓库)、huawei-SWR(SWR基础版仓库)
project_id	否	String	企业仓库实例所属的项目ID，当type为swr-pro-internal时必填
region_id	否	String	区域ID，当type为swr-pro-internal时必填
instance_id	否	String	企业仓库实例ID，当type为swr-pro-internal时必填

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	是	String	目标仓库的地址
credential	是	**Credential** object	仓库认证信息
dns_conf	否	**DnsConf** object	域名解析的配置信息
insecure	是	Boolean	是否验证远程证书，目标仓库选择SWR企业版仓库时仅支持填true。

表 4-684 Credential

参数	是否必选	参数类型	描述
type	是	String	认证方式 取值范围： • basic Basic认证方式
access_key	是	String	访问ID
access_secret	是	String	访问密码

表 4-685 DnsConf

参数	是否必选	参数类型	描述
hosts	否	Map<String,String>	主机映射，键为域名，值为对应的 IP 地址

响应参数

状态码：200

成功更新信息

状态码：400

表 4-686 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-687 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-688 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-689 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-690 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries/{registry_id}

{
  "description" : "",
  "insecure" : true,
  "name" : "test112",
  "type" : "swr-pro-internal",
  "url" : "https://testbasic-vwnajp.swr-pro.myhuaweicloud.com",
  "credential" : {
    "type" : "basic",
    "access_key" : "AKIAIOSFODNN7...",
    "access_secret" : "wJalrXUtnFEM..."
  },
  "project_id" : "test-project-id",
  "region_id" : "cn-north-7",
  "instance_id" : "test-instance-id"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateInstanceRegistrySolution {

    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
```

 版权所有 © 华为云计算技术有限公司

```
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceRegistryRequest request = new UpdateInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        request.withRegistryId({registry_id});
        UpdateRegistryRequestBody body = new UpdateRegistryRequestBody();
        Credential credentialbody = new Credential();
        credentialbody.withType("basic")
            .withAccessKey("AKIAIOSFODNN7...")
            .withAccessSecret("wJalrXUtnFEM...");
        body.withInsecure(true);
        body.withCredential(credentialbody);
        body.withUrl("https://testbasic-vwnajp.swr-pro.myhuaweicloud.com");
        body.withInstanceId("test-instance-id");
        body.withRegionId("cn-north-7");
        body.withProjectId("test-project-id");
        body.withType("swr-pro-internal");
        body.withDescription("");
        body.withName("test112");
        request.withBody(body);
        try {
            UpdateInstanceRegistryResponse response = client.updateInstanceRegistry(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        request.registry_id = {registry_id}
        credentialbody = Credential(
            type="basic",
            access_key="AKIAIOSFODNN7...",
            access_secret="wJalrXUtnFEM..."
        )
        request.body = UpdateRegistryRequestBody(
            insecure=True,
            credential=credentialbody,
            url="https://testbasic-vwnajp.swr-pro.myhuaweicloud.com",
            instance_id="test-instance-id",
            region_id="cn-north-7",
            project_id="test-project-id",
            type="swr-pro-internal",
            description="",
            name="test112"
        )
        response = client.update_instance_registry(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
hcClient, err := swr.SwrClientBuilder().
     WithRegion(region.ValueOf("<YOUR REGION>")).
     WithCredential(auth).
     SafeBuild()


if err != nil {
    fmt.Println(err)
    return
}

client := swr.NewSwrClient(hcClient)

request := &model.UpdateInstanceRegistryRequest{}
request.InstanceId = "{instance_id}"
request.RegistryId = int32({registry_id})
credentialbody := &model.Credential{
    Type: "basic",
    AccessKey: "AKIAIOSFODNN7...",
    AccessSecret: "wJalrXUtnFEM...",
}
instanceIdUpdateRegistryRequestBody:= "test-instance-id"
regionIdUpdateRegistryRequestBody:= "cn-north-7"
projectIdUpdateRegistryRequestBody:= "test-project-id"
descriptionUpdateRegistryRequestBody:= ""
request.Body = &model.UpdateRegistryRequestBody{
    Insecure: true,
    Credential: credentialbody,
    Url: "https://testbasic-vwnajp.swr-pro.myhuaweicloud.com",
    InstanceId: &instanceIdUpdateRegistryRequestBody,
    RegionId: &regionIdUpdateRegistryRequestBody,
    ProjectId: &projectIdUpdateRegistryRequestBody,
    Type: "swr-pro-internal",
    Description: &descriptionUpdateRegistryRequestBody,
    Name: "test112",
}
response, err := client.UpdateInstanceRegistry(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	成功更新信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.12.5 删除镜像同步的目标仓库 - DeleteInstanceRegistry

功能介绍

删除同步仓库

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteRegistry	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/registries/{registry_id}

表 4-691 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
registry_id	是	Integer	同步仓库ID

请求参数

表 4-692 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

表 4-693 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-694 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：403

表 4-695 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-696 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-697 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/registries/{registry_id}
```

响应示例

无

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceRegistrySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceRegistryRequest request = new DeleteInstanceRegistryRequest();
        request.withInstanceId("{instance_id}");
        request.withRegistryId({registry_id});
        try {
            DeleteInstanceRegistryResponse response = client.deleteInstanceRegistry(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 import *
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceRegistryRequest()
        request.instance_id = "{instance_id}"
        request.registry_id = {registry_id}
        response = client.delete_instance_registry(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceRegistryRequest{}
    request.InstanceId = "{instance_id}"
    request.RegistryId = int32({registry_id})
    response, err := client.DeleteInstanceRegistry(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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.6 创建镜像同步策略 - CreateInstanceReplicationPolicy

功能介绍

创建镜像同步策略

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:createReplicationPolicy	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId • swr:TargetRegion • swr:SourceRegion • swr:TargetUrls • swr:SourceUrls	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/replication/policies

表 4-698 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-699 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-700 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	策略名称, 1-256字符组成，只能包含英文大小写、数字、汉字、中划线和下划线。
description	否	String	策略描述
src_registry	否	ReplicationRegistry object	Pull-based时必选，src_registry跟dest_registry必须传一个
dest_registry	否	ReplicationRegistry object	Push-based时必选，src_registry跟dest_registry必须传一个
dest_namespace	否	String	目标命名空间名，默认值为空值
dest_namespace_replace_count	否	Integer	在镜像复制过程中，针对多路径镜像名称，可通过设置替换层级来优化仓库目录结构。支持的值为-1,0,1,2,3。dest_namespace_replace_count不传值，则默认为-1。假设原始镜像路径为 a/b/c/d/img，其中a为原始镜像的命名空间名称，目标命名空间为 ns，不同替换级别的效果如下： '-1：替换所有级': 'a/b/c/d/img' -> 'ns/img' '0：无替换': 'a/b/c/d/img' -> 'ns/a/b/c/d/img '1：替换1级': 'a/b/c/d/img' -> 'ns/b/c/d/img' '2：替换2级': 'a/b/c/d/img' -> 'ns/c/d/img' '3：替换3级': 'a/b/c/d/img' -> 'ns/d/img'

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
filters	是	Array of **Filter** objects	过滤器（目前只支持harbor类型的仓库）
repo_scope_mode	是	String	repo的选择方式，regular，selection
trigger	是	**TriggerConfig** object	触发配置
override	否	Boolean	是否覆盖，默认为false
enabled	是	Boolean	是否启用，默认为false

表 4-701 ReplicationRegistry

参数	是否必选	参数类型	描述
id	是	Long	The registry ID.

表 4-702 Filter

参数	是否必选	参数类型	描述
type	是	String	过滤类型 取值范围： • name：镜像 • tag：版本
value	是	String	过滤类型对应的正则表达式(type为name时，对应的是命名空间及制品仓库，例如library/**； type为tag时，对应的是制品版本，例如：**repo和tag**； 正则表达式有多个时，用英文逗号分隔，且在最外层加{}，如library/{test,test*,*test})

版权所有 © 华为云计算技术有限公司

表 4-703 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-704 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

响应参数

状态码：201

表 4-705 响应 Body 参数

参数	参数类型	描述
id	Integer	命名空间ID

状态码：400

表 4-706 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-707 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-708 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-709 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-710 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

- 创建推动到目标仓库的同步规则

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies

{
  "name" : "harbor-pull",
  "dest_registry" : {
    "id" : 3
  },
  "dest_namespace" : "aa",
  "filters" : [ {
    "type" : "name",
    "value" : "library/**"
  }, {
    "type" : "tag",
    "value" : "**"
  } ],
  "repo_scope_mode" : "regular",
  "trigger" : {
    "type" : "manual"
  },
  "override" : true,
  "enabled" : true
}
```

- 创建从目标仓库拉取的同步规则

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies

{
  "name" : "swr-pull",
  "src_registry" : {
    "id" : 12
  },
  "dest_namespace" : "aa",
  "filters" : [ {
    "type" : "name",
    "value" : "aa/**"
  }, {
    "type" : "tag",
    "value" : "**"
  } ],
  "repo_scope_mode" : "regular",
  "trigger" : {
    "type" : "manual"
  },
  "override" : true,
  "enabled" : true
}
```

响应示例

状态码：201

创建策略成功

 版权所有 © 华为云计算技术有限公司

```
{
  "id" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 创建推动到目标仓库的同步规则

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceReplicationPolicyRequest request = new CreateInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        CreateReplicationPolicyRequestBody body = new CreateReplicationPolicyRequestBody();
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"));
        List<Filter> listbodyFilters = new ArrayList<>();
        listbodyFilters.add(
            new Filter()
                .withType("name")
                .withValue("library/**")
        );
        listbodyFilters.add(
            new Filter()
                .withType("tag")
                .withValue("**")
        );
        ReplicationRegistry destRegistrybody = new ReplicationRegistry();
        destRegistrybody.withId(3L);
        body.withEnabled(true);
        body.withOverride(true);
```

 版权所有 © 华为云计算技术有限公司

```
        body.withTrigger(triggerbody);
        body.withRepoScopeMode("regular");
        body.withFilters(listbodyFilters);
        body.withDestNamespace("aa");
        body.withDestRegistry(destRegistrybody);
        body.withName("harbor-pull");
        request.withBody(body);
        try {
            CreateInstanceReplicationPolicyResponse response =
client.createInstanceReplicationPolicy(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());
        }
    }
}
```

- 创建从目标仓库拉取的同步规则

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class CreateInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceReplicationPolicyRequest request = new CreateInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        CreateReplicationPolicyRequestBody body = new CreateReplicationPolicyRequestBody();
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"));
        List<Filter> listbodyFilters = new ArrayList<>();
        listbodyFilters.add(
            new Filter()
```

 版权所有 © 华为云计算技术有限公司

```
                .withType("name")
                .withValue("aa/**")
        );
        listbodyFilters.add(
            new Filter()
                .withType("tag")
                .withValue("**")
        );
        ReplicationRegistry srcRegistrybody = new ReplicationRegistry();
        srcRegistrybody.withId(12L);
        body.withEnabled(true);
        body.withOverride(true);
        body.withTrigger(triggerbody);
        body.withRepoScopeMode("regular");
        body.withFilters(listbodyFilters);
        body.withDestNamespace("aa");
        body.withSrcRegistry(srcRegistrybody);
        body.withName("swr-pull");
        request.withBody(body);
        try {
            CreateInstanceReplicationPolicyResponse response =
client.createInstanceReplicationPolicy(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

- 创建推动到目标仓库的同步规则

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
        triggerbody = TriggerConfig(
            type="manual"
        )
        listFiltersbody = [
            Filter(
                type="name",
                value="library/**"
            ),
            Filter(
                type="tag",
                value="**"
            )
        ]
        destRegistrybody = ReplicationRegistry(
            id=3
        )
        request.body = CreateReplicationPolicyRequestBody(
            enabled=True,
            override=True,
            trigger=triggerbody,
            repo_scope_mode="regular",
            filters=listFiltersbody,
            dest_namespace="aa",
            dest_registry=destRegistrybody,
            name="harbor-pull"
        )
        response = client.create_instance_replication_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)
```

- 创建从目标仓库拉取的同步规则

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
        triggerbody = TriggerConfig(
            type="manual"
        )
        listFiltersbody = [
            Filter(
                type="name",
```

 版权所有 © 华为云计算技术有限公司

```
                value="aa/**"
            ),
            Filter(
                type="tag",
                value="**"
            )
        ]
        srcRegistrybody = ReplicationRegistry(
            id=12
        )
        request.body = CreateReplicationPolicyRequestBody(
            enabled=True,
            override=True,
            trigger=triggerbody,
            repo_scope_mode="regular",
            filters=listFiltersbody,
            dest_namespace="aa",
            src_registry=srcRegistrybody,
            name="swr-pull"
        )
        response = client.create_instance_replication_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

- 创建推动到目标仓库的同步规则

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
    }
    var listFiltersbody = []model.Filter{
        {
            Type: "name",
            Value: "library/**",
        },
        {
            Type: "tag",
            Value: "**",
        },
    }
    destRegistrybody := &model.ReplicationRegistry{
        Id: int64(3),
    }
    overrideCreateReplicationPolicyRequestBody:= true
    destNamespaceCreateReplicationPolicyRequestBody:= "aa"
    request.Body = &model.CreateReplicationPolicyRequestBody{
        Enabled: true,
        Override: &overrideCreateReplicationPolicyRequestBody,
        Trigger: triggerbody,
        RepoScopeMode: "regular",
        Filters: listFiltersbody,
        DestNamespace: &destNamespaceCreateReplicationPolicyRequestBody,
        DestRegistry: destRegistrybody,
        Name: "harbor-pull",
    }
    response, err := client.CreateInstanceReplicationPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 创建从目标仓库拉取的同步规则

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

 版权所有 © 华为云计算技术有限公司

```
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
    }
    var listFiltersbody = []model.Filter{
        {
            Type: "name",
            Value: "aa/**",
        },
        {
            Type: "tag",
            Value: "**",
        },
    }
    srcRegistrybody := &model.ReplicationRegistry{
        Id: int64(12),
    }
    overrideCreateReplicationPolicyRequestBody:= true
    destNamespaceCreateReplicationPolicyRequestBody:= "aa"
    request.Body = &model.CreateReplicationPolicyRequestBody{
        Enabled: true,
        Override: &overrideCreateReplicationPolicyRequestBody,
        Trigger: triggerbody,
        RepoScopeMode: "regular",
        Filters: listFiltersbody,
        DestNamespace: &destNamespaceCreateReplicationPolicyRequestBody,
        SrcRegistry: srcRegistrybody,
        Name: "swr-pull",
    }
    response, err := client.CreateInstanceReplicationPolicy(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
201	创建策略成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.7 获取镜像同步策略列表 - ListInstanceReplicationPolicies

功能介绍

获取镜像同步策略列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listReplicationPolicies	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/replication/policies

表 4-711 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-712 Query 参数

参数	是否必选	参数类型	描述
order_column	否	String	排序字段，支持created_at、updated_at、name，默认为created_at
order_type	否	String	排序方式，与order_column配合使用，默认desc 取值范围： ● desc：降序 ● asc：升序
name	否	String	名称，模糊查询
registry_id	否	Integer	仓库ID
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-713 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-714 响应 Body 参数

参数	参数类型	描述
policies	Array of **ReplicationPolicy** objects	同步策略列表
total	Integer	同步策略总数

表 4-715 ReplicationPolicy

参数	参数类型	描述
id	Integer	策略ID
name	String	策略名称
description	String	策略描述
src_registry	**ReplicationRegistry** object	Pull-based时必选, src_registry跟dest_registry二选一
dest_registry	**ReplicationRegistry** object	Push-based时必选， src_registry跟dest_registry二选一
dest_namespace	String	目标命名空间名，默认为空值

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
dest_namespace_replace_count	Integer	在镜像复制过程中，针对多路径镜像名称，可通过设置替换层级来优化仓库目录结构。支持的值为-1,0,1,2,3。dest_namespace_replace_count不传值，则默认为-1。假设原始镜像路径为a/b/c/d/img，其中a为原始镜像的命名空间名称，目标命名空间为 ns，不同替换级别的效果如下： '-1：替换所有级': 'a/b/c/d/img' -> 'ns/img' '0：无替换': 'a/b/c/d/img' -> 'ns/a/b/c/d/img '1：替换1级': 'a/b/c/d/img' -> 'ns/b/c/d/img' '2：替换2级': 'a/b/c/d/img' -> 'ns/c/d/img' '3：替换3级': 'a/b/c/d/img' -> 'ns/d/img'
filters	Array of **Filter** objects	源资源过滤器
repo_scope_mode	String	repo过滤类型
trigger	**TriggerConfig** object	触发配置
override	Boolean	是否覆盖
enabled	Boolean	是否启用
created_at	String	创建时间
updated_at	String	更新时间

表 4-716 ReplicationRegistry

参数	参数类型	描述
id	Long	The registry ID.

版权所有 © 华为云计算技术有限公司

表 4-717 Filter

参数	参数类型	描述
type	String	过滤类型 取值范围： • name：镜像 • tag：版本
value	String	过滤类型对应的正则表达式(type为name时，对应的是命名空间及制品仓库，例如library/**； type为tag时，对应的是制品版本，例如：**repo和**tag； 正则表达式有多个时，用英文逗号分隔，且在最外层加{}，如library/{test,test*,*test})

表 4-718 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	**TriggerSetting** object	触发设置，只有type为scheduled的时候，才需要设置

表 4-719 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

状态码：400

表 4-720 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-721 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-722 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-723 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-724 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies
```

响应示例

状态码：200

查询镜像同步策略列表成功

```
{
  "total" : 10,
  "policies" : [ {
    "id" : 20,
    "name" : "asfdf",
    "description" : "",
    "src_registry" : {
      "id" : 0
    },
    "dest_registry" : {
      "id" : 11
    },
    "repo_scope_mode" : "regular",
    "filters" : [ {
      "type" : "name",
      "value" : "root"
    }, {
      "type" : "tag",
      "value" : "**"
    } ],
    "dest_namespace" : "",
    "trigger" : {
      "type" : "manual",
      "trigger_settings" : {
        "cron" : ""
      }
    },
    "override" : true,
    "enabled" : true,
    "created_at" : "2020-12-03T11:55:14.338485Z",
    "updated_at" : "2020-12-03T11:55:14.338485Z"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceReplicationPoliciesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceReplicationPoliciesRequest request = new ListInstanceReplicationPoliciesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceReplicationPoliciesResponse response = client.listInstanceReplicationPolicies(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceReplicationPoliciesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_replication_policies(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.ListInstanceReplicationPoliciesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceReplicationPolicies(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	查询镜像同步策略列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.8 获取镜像同步策略详情 - ShowInstanceReplicationPolicy

功能介绍

获取镜像同步策略详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getReplicationPolicy	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}

表 4-725 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
policy_id	是	Integer	策略ID

请求参数

表 4-726 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-727 响应 Body 参数

参数	参数类型	描述
id	Integer	策略ID
name	String	策略名称
description	String	策略描述
src_registry	**ReplicationRegistry** object	Pull-based时必选, src_registry跟dest_registry二选一
dest_registry	**ReplicationRegistry** object	Push-based时必选， src_registry跟dest_registry二选一
dest_namespace	String	目标命名空间名，默认为空值
dest_namespace_replace_count	Integer	在镜像复制过程中，针对多路径镜像名称，可通过设置替换层级来优化仓库目录结构。支持的值为-1,0,1,2,3。dest_namespace_replace_count不传值，则默认为-1。假设原始镜像路径为a/b/c/d/img，其中a为原始镜像的命名空间名称，目标命名空间为 ns，不同替换级别的效果如下： '-1：替换所有级': 'a/b/c/d/img' -> 'ns/img' '0：无替换': 'a/b/c/d/img' -> 'ns/a/b/c/d/img '1：替换1级': 'a/b/c/d/img' -> 'ns/b/c/d/img' '2：替换2级': 'a/b/c/d/img' -> 'ns/c/d/img' '3：替换3级': 'a/b/c/d/img' -> 'ns/d/img'
filters	Array of **Filter** objects	源资源过滤器
repo_scope_mode	String	repo过滤类型
trigger	**TriggerConfig** object	触发配置
override	Boolean	是否覆盖
enabled	Boolean	是否启用
created_at	String	创建时间
updated_at	String	更新时间

版权所有 © 华为云计算技术有限公司

表 4-728 ReplicationRegistry

参数	参数类型	描述
id	Long	The registry ID.

表 4-729 Filter

参数	参数类型	描述
type	String	过滤类型 取值范围： • name：镜像 • tag：版本
value	String	过滤类型对应的正则表达式(type为name时，对应的是命名空间及制品仓库，例如library/**； type为tag时，对应的是制品版本，例如：**repo和**tag； 正则表达式有多个时，用英文逗号分隔，且在最外层加{}，如library/{test,test*,*test})

表 4-730 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	**TriggerSetting** object	触发设置，只有type为scheduled的时候，才需要设置

表 4-731 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-732 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-733 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-734 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-735 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-736 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : 20,
  "name" : "asfdf",
  "description" : "",
  "src_registry" : {
    "id" : 0
  },
  "dest_registry" : {
    "id" : 11
  },
  "dest_namespace" : "",
  "filters" : [ {
    "type" : "name",
    "value" : "root"
  }, {
    "type" : "tag",
    "value" : "**"
  } ],
  "trigger" : {
    "type" : "manual",
    "trigger_settings" : {
      "cron" : ""
    }
  },
  "repo_scope_mode" : "regular",
  "override" : true,
  "enabled" : true,
  "created_at" : "2020-12-03T11:55:14.338485Z",
```


版权所有 © 华为云计算技术有限公司

```
  "updated_at" : "2020-12-03T11:55:14.338485Z"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceReplicationPolicyRequest request = new ShowInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withPolicyId({policy_id});
        try {
            ShowInstanceReplicationPolicyResponse response = client.showInstanceReplicationPolicy(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
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
        request.policy_id = {policy_id}
        response = client.show_instance_replication_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
```

 版权所有 © 华为云计算技术有限公司

```
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.PolicyId = int32({policy_id})
    response, err := client.ShowInstanceReplicationPolicy(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.9 修改镜像同步策略 - UpdateInstanceReplicationPolicy

功能介绍

修改镜像同步策略

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateReplicationPolicy	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId • swr:TargetRegion • swr:SourceRegion • swr:TargetUrls • swr:SourceUrls	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}

表 4-737 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
policy_id	是	Integer	同步策略ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-738 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-739 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	策略名称, 1-256字符组成，只能包含英文大小写、数字、汉字、中划线和下划线。
description	否	String	策略描述
src_registry	否	**ReplicationRegistry** object	Pull-based时必选, src_registry跟dest_registry二选一
dest_registry	否	**ReplicationRegistry** object	Push-based时必选，src_registry跟dest_registry二选一
dest_namespace	否	String	目标命名空间名，默认为空值
filters	是	Array of **Filter** objects	过滤器
repo_scope_mode	是	String	repo的选择方式，regular，selection
trigger	是	**TriggerConfig** object	触发配置
override	否	Boolean	是否覆盖，默认为false
enabled	是	Boolean	是否启用，默认为false

表 4-740 ReplicationRegistry

参数	是否必选	参数类型	描述
id	是	Long	The registry ID.

版权所有 © 华为云计算技术有限公司

表 4-741 Filter

参数	是否必选	参数类型	描述
type	是	String	过滤类型 取值范围： • name：镜像 • tag：版本
value	是	String	过滤类型对应的正则表达式(type为name时，对应的是命名空间及制品仓库，例如library/**； type为tag时，对应的是制品版本，例如：**repo和**tag； 正则表达式有多个时，用英文逗号分隔，且在最外层加{}，如library/{test,test*,*test})

表 4-742 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-743 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

响应参数

状态码：200

成功更新策略信息

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-744 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-745 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-746 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-747 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-748 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}

{
  "description" : "test",
  "dest_namespace" : "",
  "name" : "test",
  "dest_registry" : {
    "id" : 6
  },
  "trigger" : {
    "trigger_settings" : {
      "cron" : ""
    },
    "type" : "manual"
  },
  "filters" : [ {
    "type" : "name",
    "value" : "**/**"
  }, {
    "type" : "tag",
    "value" : "**"
  } ],
  "override" : true,
  "enabled" : true,
  "repo_scope_mode" : "regular"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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

public class UpdateInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceReplicationPolicyRequest request = new UpdateInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withPolicyId({policy_id});
        UpdateReplicationPolicyRequestBody body = new UpdateReplicationPolicyRequestBody();
        TriggerSetting triggerSettingsTrigger = new TriggerSetting();
        triggerSettingsTrigger.withCron("");
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"))
            .withTriggerSettings(triggerSettingsTrigger);
        List<Filter> listbodyFilters = new ArrayList<>();
        listbodyFilters.add(
            new Filter()
                .withType("name")
                .withValue("**/**")
        );
        listbodyFilters.add(
            new Filter()
                .withType("tag")
                .withValue("**")
        );
        ReplicationRegistry destRegistrybody = new ReplicationRegistry();
        destRegistrybody.withId(6L);
        body.withEnabled(true);
        body.withOverride(true);
        body.withTrigger(triggerbody);
        body.withRepoScopeMode("regular");
        body.withFilters(listbodyFilters);
        body.withDestNamespace("");
        body.withDestRegistry(destRegistrybody);
        body.withDescription("test");
        body.withName("test");
        request.withBody(body);
        try {
```

 版权所有 © 华为云计算技术有限公司

```
            UpdateInstanceReplicationPolicyResponse response =
client.updateInstanceReplicationPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
        request.policy_id = {policy_id}
        triggerSettingsTrigger = TriggerSetting(
            cron=""
        )
        triggerbody = TriggerConfig(
            type="manual",
            trigger_settings=triggerSettingsTrigger
        )
        listFiltersbody = [
            Filter(
                type="name",
                value="**/**"
            ),
            Filter(
                type="tag",
                value="**"
            )
        ]
        destRegistrybody = ReplicationRegistry(
            id=6
        )
        request.body = UpdateReplicationPolicyRequestBody(
```

 版权所有 © 华为云计算技术有限公司

```
            enabled=True,
            override=True,
            trigger=triggerbody,
            repo_scope_mode="regular",
            filters=listFiltersbody,
            dest_namespace="",
            dest_registry=destRegistrybody,
            description="test",
            name="test"
        )
        response = client.update_instance_replication_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.PolicyId = int32({policy_id})
    cronTriggerSettings:= ""
    triggerSettingsTrigger := &model.TriggerSetting{
```

 版权所有 © 华为云计算技术有限公司

```
        Cron: &cronTriggerSettings,
    }
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
        TriggerSettings: triggerSettingsTrigger,
    }
    var listFiltersbody = []model.Filter{
        {
            Type: "name",
            Value: "**/**",
        },
        {
            Type: "tag",
            Value: "**",
        },
    }
    destRegistrybody := &model.ReplicationRegistry{
        Id: int64(6),
    }
    overrideUpdateReplicationPolicyRequestBody:= true
    destNamespaceUpdateReplicationPolicyRequestBody:= ""
    descriptionUpdateReplicationPolicyRequestBody:= "test"
    request.Body = &model.UpdateReplicationPolicyRequestBody{
        Enabled: true,
        Override: &overrideUpdateReplicationPolicyRequestBody,
        Trigger: triggerbody,
        RepoScopeMode: "regular",
        Filters: listFiltersbody,
        DestNamespace: &destNamespaceUpdateReplicationPolicyRequestBody,
        DestRegistry: destRegistrybody,
        Description: &descriptionUpdateReplicationPolicyRequestBody,
        Name: "test",
    }
    response, err := client.UpdateInstanceReplicationPolicy(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	成功更新策略信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.12.10 删除镜像同步策略 - DeleteInstanceReplicationPolicy

功能介绍

删除镜像同步策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:deleteReplicationPolicy	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}

表 4-749 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
policy_id	是	Integer	同步策略ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-750 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

表 4-751 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-752 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-753 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-754 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-755 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceReplicationPolicyRequest request = new DeleteInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withPolicyId({policy_id});
        try {
            DeleteInstanceReplicationPolicyResponse response = client.deleteInstanceReplicationPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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.
```

 版权所有 © 华为云计算技术有限公司

```
    # In this example, AK and SK are 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
        request.policy_id = {policy_id}
        response = client.delete_instance_replication_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.PolicyId = int32({policy_id})
    response, err := client.DeleteInstanceReplicationPolicy(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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.11 手动执行镜像同步策略 - ExecuteInstanceReplicationPolicy

功能介绍

手动执行同步策略

接口约束

无

调用方法

请参见**如何调用API**。

版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:executeReplicationPolicy	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/replication/executions

表 4-756 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-757 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-758 请求 Body 参数

参数	是否必选	参数类型	描述
policy_id	是	Integer	策略ID

版权所有 © 华为云计算技术有限公司

响应参数

状态码：202

表 4-759 响应 Body 参数

参数	参数类型	描述
id	Integer	执行ID

状态码：400

表 4-760 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-761 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-762 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-763 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-764 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/executions

{
  "policy_id" : 123
}
```

响应示例

状态码：202

执行策略被接受

```
{
  "id" : 34
}
```


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ExecuteInstanceReplicationPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ExecuteInstanceReplicationPolicyRequest request = new ExecuteInstanceReplicationPolicyRequest();
        request.withInstanceId("{instance_id}");
        CreateReplicationExecutionRequestBody body = new CreateReplicationExecutionRequestBody();
        body.withPolicyId(123);
        request.withBody(body);
        try {
            ExecuteInstanceReplicationPolicyResponse response = client.executeInstanceReplicationPolicy(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
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ExecuteInstanceReplicationPolicyRequest()
        request.instance_id = "{instance_id}"
        request.body = CreateReplicationExecutionRequestBody(
            policy_id=123
        )
        response = client.execute_instance_replication_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
```

 版权所有 © 华为云计算技术有限公司

```
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ExecuteInstanceReplicationPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.CreateReplicationExecutionRequestBody{
        PolicyId: int32(123),
    }
    response, err := client.ExecuteInstanceReplicationPolicy(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
202	执行策略被接受
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.12 获取镜像同步策略执行记录列表 - ListInstanceReplicationPolicyExecutions

功能介绍

获取镜像同步策略执行记录列表

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listReplicationPolicyExecutions	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/replication/executions

表 4-765 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-766 Query 参数

参数	是否必选	参数类型	描述
policy_id	否	Integer	同步策略ID
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100 **注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-767 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-768 响应 Body 参数

参数	参数类型	描述
executions	Array of **Execution** objects	执行记录列表
total	Integer	执行记录总数

表 4-769 Execution

参数	参数类型	描述
id	Integer	记录ID
policy_id	Integer	策略ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
status	String	状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
status_text	String	状态详情
total	Integer	总数
failed	Integer	失败数
succeed	Integer	成功数
in_progress	Integer	进行中的数量
stopped	Integer	停止数
trigger	String	触发类型
created_at	String	创建时间
updated_at	String	更新时间

状态码：400

表 4-770 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-771 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-772 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-773 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-774 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/executions
```

响应示例

状态码：200

查询镜像同步策略执行记录列表成功

```
{
  "total" : 1,
  "executions" : [ {
    "id" : 114204,
    "policy_id" : 12,
    "status" : "Failed",
    "status_text" : "failed to fetch artifacts",
    "total" : 0,
    "failed" : 0,
    "succeed" : 0,
    "in_progress" : 0,
    "stopped" : 0,
    "trigger" : "manual",
    "created_at" : "2021-02-01T03:49:58Z",
    "updated_at" : "2021-02-01T03:49:58Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceReplicationPolicyExecutionsSolution {

    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");
```


版权所有 © 华为云计算技术有限公司

```
        String projectId = "{project_id}";

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceReplicationPolicyExecutionsRequest request = new
ListInstanceReplicationPolicyExecutionsRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceReplicationPolicyExecutionsResponse response =
client.listInstanceReplicationPolicyExecutions(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

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

 版权所有 © 华为云计算技术有限公司

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceReplicationPolicyExecutionsRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceReplicationPolicyExecutions(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	查询镜像同步策略执行记录列表成功

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.13 停止镜像同步任务 - StopInstanceReplicationPolicyExecution

功能介绍

停止镜像同步任务

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:stopReplicationPolicyExecution	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

PUT /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}

表 4-775 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
execution_id	是	Integer	同步策略执行记录ID

请求参数

表 4-776 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

停止成功

状态码：400

表 4-777 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-778 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-779 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-780 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-781 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class StopInstanceReplicationPolicyExecutionSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        StopInstanceReplicationPolicyExecutionRequest request = new StopInstanceReplicationPolicyExecutionRequest();
        request.withInstanceId("{instance_id}");
        request.withExecutionId({execution_id});
        try {
            StopInstanceReplicationPolicyExecutionResponse response = client.stopInstanceReplicationPolicyExecution(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopInstanceReplicationPolicyExecutionRequest()
        request.instance_id = "{instance_id}"
        request.execution_id = {execution_id}
        response = client.stop_instance_replication_policy_execution(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.StopInstanceReplicationPolicyExecutionRequest{}
    request.InstanceId = "{instance_id}"
    request.ExecutionId = int32({execution_id})
    response, err := client.StopInstanceReplicationPolicyExecution(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	停止成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.12.14 获取镜像同步策略执行任务的镜像列表 - ListInstanceReplicationPolicyExecTasks

功能介绍

获取镜像同步策略执行任务的镜像列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listReplicationPolicyExecTasks	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks

表 4-782 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
execution_id	是	Integer	同步策略执行记录ID

表 4-783 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-784 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-785 响应 Body 参数

参数	参数类型	描述
tasks	Array of **TaskDetail** objects	同步策略执行记录任务列表
total	Integer	同步策略执行记录任务总数

版权所有 © 华为云计算技术有限公司

表 4-786 TaskDetail

参数	参数类型	描述
id	Integer	任务ID
execution_id	Integer	执行任务的ID
resource_type	String	资源类型 取值范围： ● artifact：镜像制品
src_resource	String	源资源
dst_resource	String	目标资源
operation	String	操作类型
job_id	String	harbor任务ID
status	String	状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
StatusRevision	Integer	状态修订
start_time	String	开始时间
end_time	String	结束时间

状态码：400

表 4-787 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-788 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-789 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-790 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-791 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 1,
  "tasks" : [ {
    "id" : 1927,
    "execution_id" : 83,
    "resource_type" : "artifact",
    "src_resource" : "{\"type\":\"artifact\",\"metadata\":{\"repository\":{\"name\":\"root/nats\"}}}",
    "dst_resource" : "{\"type\":\"artifact\",\"metadata\":{\"repository\":{\"name\":\"harbor0/nats\"}}}",
    "operation" : "copy",
    "job_id" : "3541b99f9e562586ffbf3445",
    "status" : "Succeed",
    "StatusRevision" : 1612598465,
    "start_time" : "2021-02-06T08:01:06.04266Z",
    "end_time" : "2021-02-06T08:34:46Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceReplicationPolicyExecTasksSolution {

    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}";
```


版权所有 © 华为云计算技术有限公司

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceReplicationPolicyExecTasksRequest request = new
ListInstanceReplicationPolicyExecTasksRequest();
        request.withInstanceId("{instance_id}");
        request.withExecutionId({execution_id});
        try {
            ListInstanceReplicationPolicyExecTasksResponse response =
client.listInstanceReplicationPolicyExecTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceReplicationPolicyExecTasksRequest()
        request.instance_id = "{instance_id}"
        request.execution_id = {execution_id}
        response = client.list_instance_replication_policy_exec_tasks(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceReplicationPolicyExecTasksRequest{}
    request.InstanceId = "{instance_id}"
    request.ExecutionId = int32({execution_id})
    response, err := client.ListInstanceReplicationPolicyExecTasks(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.12.15 获取镜像同步策略执行任务的镜像版本列表 - ListInstanceReplicationPolicyExecSubTasks

功能介绍

获取镜像同步策略执行任务的镜像版本列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listReplicationPolicyExecSubTasks	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks/{task_id}/subtasks

表 4-792 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
execution_id	是	Integer	同步策略执行记录ID
task_id	是	Integer	任务ID

表 4-793 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
status	否	String	状态 取值范围： • Initialized：初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止

请求参数

表 4-794 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-795 响应 Body 参数

参数	参数类型	描述
sub_tasks	Array of **SubtaskDetail** objects	镜像复制执行记录子任务列表
total	Integer	镜像复制执行记录子任务总数

表 4-796 SubtaskDetail

参数	参数类型	描述
id	Integer	子任务ID
job_id	String	内部任务ID
namespace	String	命名空间名

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
repository	String	制品仓库
tag	String	制品版本
digest	String	摘要
status	String	状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
status_text	String	状态信息
start_time	String	开始时间
end_time	String	结束时间

状态码：400

表 4-797 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-798 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-799 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-800 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-801 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks/{task_id}/subtasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 1,
  "sub_tasks" : [ {
    "id" : 2,
    "job_id" : "8",
    "namespace" : "libz",
    "repository" : "2d7c874a7407d6c40f016254",
    "tag" : "2d7c874a7407d6c40f016254",
    "digest" : "2d7c874a7407d6c40f016254",
    "status" : "Succeed",
    "status_text" : "",
    "start_time" : "2020-12-01T07:06:35Z",
    "end_time" : "2020-12-01T07:06:36Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceReplicationPolicyExecSubTasksSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceReplicationPolicyExecSubTasksRequest request = new
```

 版权所有 © 华为云计算技术有限公司

```
ListInstanceReplicationPolicyExecSubTasksRequest();
        request.withInstanceId("{instance_id}");
        request.withExecutionId({execution_id});
        request.withTaskId({task_id});
        try {
            ListInstanceReplicationPolicyExecSubTasksResponse response =
client.listInstanceReplicationPolicyExecSubTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceReplicationPolicyExecSubTasksRequest()
        request.instance_id = "{instance_id}"
        request.execution_id = {execution_id}
        request.task_id = {task_id}
        response = client.list_instance_replication_policy_exec_sub_tasks(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"
```

```
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceReplicationPolicyExecSubTasksRequest{}
    request.InstanceId = "{instance_id}"
    request.ExecutionId = int32({execution_id})
    request.TaskId = int32({task_id})
    response, err := client.ListInstanceReplicationPolicyExecSubTasks(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	查询成功
400	错误的请求
401	鉴权失败

 版权所有 © 华为云计算技术有限公司

状态码	描述
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13 镜像签名管理

4.13.1 获取签名策略列表 - ListInstanceSignPolicies

功能介绍

获取签名策略列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listSignPolicies	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/signature/policies

表 4-802 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-803 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-804 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-805 响应 Body 参数

参数	参数类型	描述
policies	Array of **SignPolicyDetail** objects	签名策略列表

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
total	Integer	签名策略总数

表 4-806 SignPolicyDetail

参数	参数类型	描述
id	Integer	签名策略ID
name	String	签名策略名称
description	String	签名策略描述
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名
trigger	**TriggerConfig** object	触发方式
creator	String	创建者
enabled	Boolean	是否开启
scope_rules	Array of **SignScopeRule** objects	作用范围规则
created_at	String	创建时间
updated_at	String	更新时间
signature_algorithm	String	加签算法，KMS的密钥算法EC_P256对应着ECDSA_SHA_256，EC_P384对应着ECDSA_SHA_384，SM2对应着SM2DSA_SM3
signature_key	String	签名算法key ID
signature_method	String	镜像签名方式

表 4-807 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
trigger_settings	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-808 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

表 4-809 SignScopeRule

参数	参数类型	描述
tag_selectors	Array of SignRuleSelector objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<SignRuleSelector>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-810 SignRuleSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
extras	String	预留字段，镜像签名策略中，对无版本的制品进行签名则传入{"untagged":true}

状态码：400

表 4-811 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-812 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-813 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-814 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-815 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

- 创建签名规则，范围包括无版本制品

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/signature/policies

{
  "name" : "aaa",
  "enabled" : true,
  "signature_method" : "KMS",
  "signature_algorithm" : "ECDSA_SHA_256",
  "signature_key" : "key-123456",
  "trigger" : {
    "type" : "manual"
  },
  "scope_rules" : [ {
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : "{\"untagged\":true}"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "{repo}"
      } ]
    },
    "repo_scope_mode" : "selection"
```


版权所有 © 华为云计算技术有限公司

```
  } ]
}
```

- 创建签名规则，范围不包括无版本制品

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/signature/policies

{
  "name" : "aaa",
  "enabled" : true,
  "signature_method" : "KMS",
  "signature_algorithm" : "ECDSA_SHA_256",
  "signature_key" : "key-123456",
  "trigger" : {
    "type" : "manual"
  },
  "scope_rules" : [ {
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : ""
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "{repo}"
      } ]
    },
    "repo_scope_mode" : "selection"
  } ]
}
```

响应示例

状态码：200

查询签名策略列表成功

```
{
  "total" : 2,
  "policies" : [ {
    "id" : 1,
    "name" : "test-abc",
    "description" : "",
    "created_at" : "2020-11-19T02:58:54.809Z",
    "updated_at" : "2020-11-19T02:58:54.809Z",
    "creator" : "admin",
    "enabled" : true,
    "scope_rules" : [ {
      "scope_selectors" : {
        "repository" : [ {
          "kind" : "doublestar",
          "decoration" : "repoMatches",
          "pattern" : "**"
        } ]
      },
      "tag_selectors" : [ {
        "kind" : "doublestar",
        "decoration" : "matches",
        "pattern" : "**"
      } ]
    } ],
    "namespace_id" : 1,
    "trigger" : {
      "type" : "manual"
    }
  } ]
}
```

 版权所有 © 华为云计算技术有限公司

SDK 代码示例

SDK代码示例如下。

Java

- 创建签名规则，范围包括无版本制品

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceSignPoliciesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceSignPoliciesRequest request = new ListInstanceSignPoliciesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceSignPoliciesResponse response = client.listInstanceSignPolicies(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());
        }
    }
}
```

- 创建签名规则，范围不包括无版本制品

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceSignPoliciesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceSignPoliciesRequest request = new ListInstanceSignPoliciesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceSignPoliciesResponse response = client.listInstanceSignPolicies(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

- 创建签名规则，范围包括无版本制品

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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)
```

 版权所有 © 华为云计算技术有限公司

```
    client = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

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

- 创建签名规则，范围不包括无版本制品

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceSignPoliciesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_sign_policies(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

- 创建签名规则，范围包括无版本制品

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

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

- 创建签名规则，范围不包括无版本制品

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

 版权所有 © 华为云计算技术有限公司

```
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceSignPoliciesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceSignPolicies(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	查询签名策略列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.13.2 创建签名策略 - CreateInstanceSignPolicy

功能介绍

创建签名策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:createSignPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies

表 4-816 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称

版权所有 © 华为云计算技术有限公司

请求参数

表 4-817 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-818 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	签名策略名称，由字母、汉字、数字、下划线（_）、中划线 (-) 组成，1-256个字符。
description	否	String	签名策略描述
enabled	是	Boolean	是否开启
signature_method	是	String	加签方式，可选KMS
signature_algorithm	是	String	加签算法，KMS的密钥算法EC_P256对应着ECDSA_SHA_256，EC_P384对应着ECDSA_SHA_384，SM2对应着SM2DSA_SM3
signature_key	是	String	加签Key
trigger	是	**TriggerConfig** object	触发方式
scope_rules	是	Array of **SignScopeRule** objects	作用范围规则

版权所有 © 华为云计算技术有限公司

表 4-819 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	**TriggerSetting** object	触发设置，只有type为scheduled的时候，才需要设置

表 4-820 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

表 4-821 SignScopeRule

参数	是否必选	参数类型	描述
tag_selectors	是	Array of **SignRuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**SignRuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-822 SignRuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段，镜像签名策略中，对无版本的制品进行签名则传入{"untagged":true}

响应参数

状态码：201

表 4-823 响应 Body 参数

参数	参数类型	描述
id	Integer	策略ID

状态码：400

表 4-824 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-825 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-826 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-827 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-828 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies

{
  "name" : "test11",
  "scope_rules" : [ {
    "repo_scope_mode" : "regular",
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : "{\"untagged\":true}"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    }
  } ],
  "enabled" : true,
  "trigger" : {
    "trigger_settings" : {
      "cron" : ""
    },
    "type" : "manual"
  },
  "signature_method" : "KMS",
  "signature_algorithm" : "ECDSA_SHA_384",
  "signature_key" : "668985d5-919d-4c51-9293-eb846c78cbf0"
}
```

响应示例

状态码：201

执行签名策略成功

```
{
  "id" : 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;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateInstanceSignPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceSignPolicyRequest request = new CreateInstanceSignPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        CreateSignaturePolicyRequestBody body = new CreateSignaturePolicyRequestBody();
        List<SignRuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new SignRuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<SignRuleSelector>> listScopeRulesScopeSelectors = new HashMap<>();
        listScopeRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<SignRuleSelector> listScopeRulesTagSelectors = new ArrayList<>();
        listScopeRulesTagSelectors.add(
            new SignRuleSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
                .withExtras("{"untagged":true}")
        );
        List<SignScopeRule> listbodyScopeRules = new ArrayList<>();
        listbodyScopeRules.add(
            new SignScopeRule()
                .withTagSelectors(listScopeRulesTagSelectors)
                .withScopeSelectors(listScopeRulesScopeSelectors)
                .withRepoScopeMode("regular")
        );
        TriggerSetting triggerSettingsTrigger = new TriggerSetting();
        triggerSettingsTrigger.withCron("");
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"))
            .withTriggerSettings(triggerSettingsTrigger);
        body.withScopeRules(listbodyScopeRules);
        body.withTrigger(triggerbody);
        body.withSignatureKey("668985d5-919d-4c51-9293-eb846c78cbf0");
```

 版权所有 © 华为云计算技术有限公司

```
body.withSignatureAlgorithm(CreateSignaturePolicyRequestBody.CreateSignaturePolicyRequestBody.fromVal
ue("ECDSA_SHA_384"));

body.withSignatureMethod(CreateSignaturePolicyRequestBody.CreateSignaturePolicyRequestBody.fromValu
e("KMS"));
        body.withEnabled(true);
        body.withName("test11");
        request.withBody(body);
        try {
            CreateInstanceSignPolicyResponse response = client.createInstanceSignPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateInstanceSignPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        listScopeSelectorsScopeSelectors = [
            SignRuleSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsScopeRules = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsScopeRules = [
            SignRuleSelector(
                kind="doublestar",
```

```
            decoration="matches",
            pattern="**",
            extras="{"untagged":true}"
        )
    ]
    listScopeRulesbody = [
        SignScopeRule(
            tag_selectors=listTagSelectorsScopeRules,
            scope_selectors=listScopeSelectorsScopeRules,
            repo_scope_mode="regular"
        )
    ]
    triggerSettingsTrigger = TriggerSetting(
        cron=""
    )
    triggerbody = TriggerConfig(
        type="manual",
        trigger_settings=triggerSettingsTrigger
    )
    request.body = CreateSignaturePolicyRequestBody(
        scope_rules=listScopeRulesbody,
        trigger=triggerbody,
        signature_key="668985d5-919d-4c51-9293-eb846c78cbf0",
        signature_algorithm="ECDSA_SHA_384",
        signature_method="KMS",
        enabled=True,
        name="test11"
    )
    response = client.create_instance_sign_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceSignPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    var listScopeSelectorsScopeSelectors = []model.SignRuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsScopeRules = map[string][](model.SignRuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    extrasTagSelectors:= "{"untagged":true}"
    var listTagSelectorsScopeRules = []model.SignRuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
            Extras: &extrasTagSelectors,
        },
    }
    var listScopeRulesbody = []model.SignScopeRule{
        {
            TagSelectors: listTagSelectorsScopeRules,
            ScopeSelectors: listScopeSelectorsScopeRules,
            RepoScopeMode: "regular",
        },
    }
    cronTriggerSettings:= ""
    triggerSettingsTrigger := &model.TriggerSetting{
        Cron: &cronTriggerSettings,
    }
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
        TriggerSettings: triggerSettingsTrigger,
    }
    request.Body = &model.CreateSignaturePolicyRequestBody{
        ScopeRules: listScopeRulesbody,
        Trigger: triggerbody,
        SignatureKey: "668985d5-919d-4c51-9293-eb846c78cbf0",
        SignatureAlgorithm:
model.GetCreateSignaturePolicyRequestBodySignatureAlgorithmEnum().ECDSA_SHA_384,
        SignatureMethod: model.GetCreateSignaturePolicyRequestBodySignatureMethodEnum().KMS,
        Enabled: true,
        Name: "test11",
    }
    response, err := client.CreateInstanceSignPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	执行签名策略成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.3 获取签名策略详情 - ShowInstanceSignPolicy

功能介绍

获取签名策略详情

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getSignPolicy	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}

表 4-829 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID

请求参数

表 4-830 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

版权所有 © 华为云计算技术有限公司

表 4-831 响应 Body 参数

参数	参数类型	描述
id	Integer	签名策略ID
name	String	签名策略名称
description	String	签名策略描述
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名
trigger	**TriggerConfig** object	触发方式
creator	String	创建者
enabled	Boolean	是否开启
scope_rules	Array of **SignScopeRule** objects	作用范围规则
created_at	String	创建时间
updated_at	String	更新时间
signature_algorithm	String	加签算法，KMS的密钥算法EC_P256对应着ECDSA_SHA_256，EC_P384对应着ECDSA_SHA_384，SM2对应着SM2DSA_SM3
signature_key	String	签名算法key ID
signature_method	String	镜像签名方式

表 4-832 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	**TriggerSetting** object	触发设置，只有type为scheduled的时候，才需要设置

版权所有 © 华为云计算技术有限公司

表 4-833 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

表 4-834 SignScopeRule

参数	参数类型	描述
tag_selectors	Array of **SignRuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<**SignRuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-835 SignRuleSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	String	预留字段，镜像签名策略中，对无版本的制品进行签名则传入{"untagged":true}

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-836 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-837 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-838 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-839 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-840 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : 1,
  "name" : "test-abc",
  "description" : "",
  "created_at" : "2020-11-19T02:58:54.809Z",
  "updated_at" : "2020-11-19T02:58:54.809Z",
  "creator" : "admin",
  "enabled" : true,
  "namespace_name" : "test-namespace",
  "signature_algorithm" : "ECDSA_SHA_384",
  "signature_key" : "668985d5-919d-4c51-9293-eb846c78cbf0",
  "signature_method" : "KMS",
  "scope_rules" : [ {
    "repo_scope_mode" : "regular",
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    },
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**"
    } ]
  } ],
```


版权所有 © 华为云计算技术有限公司

```
  "namespace_id" : 1,
  "trigger" : {
    "type" : "manual"
  }
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceSignPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceSignPolicyRequest request = new ShowInstanceSignPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ShowInstanceSignPolicyResponse response = client.showInstanceSignPolicy(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceSignPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.show_instance_sign_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
```

 版权所有 © 华为云计算技术有限公司

```
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceSignPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ShowInstanceSignPolicy(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.4 修改签名策略 - UpdateInstanceSignPolicy

功能介绍

修改签名策略

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateSignPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}

表 4-841 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-842 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-843 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	签名策略名称，由字母、汉字、数字、下划线（_）、中划线 (-) 组成，1-256个字符。
description	否	String	签名策略描述
enabled	是	Boolean	是否开启
signature_method	是	String	加签方式，可选KMS
signature_algorithm	是	String	加签算法，KMS的密钥算法EC_P256对应着ECDSA_SHA_256，EC_P384对应着ECDSA_SHA_384，SM2对应着SM2DSA_SM3
signature_key	是	String	加签Key
trigger	是	**TriggerConfig** object	触发方式
scope_rules	是	Array of **SignScopeRule** objects	作用范围规则

版权所有 © 华为云计算技术有限公司

表 4-844 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-845 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

表 4-846 SignScopeRule

参数	是否必选	参数类型	描述
tag_selectors	是	Array of SignRuleSelector objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<SignRuleSelector>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repository选择方式。可选regular、selection，前端显示需要，api调用时可选regular

表 4-847 SignRuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？： 匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段，镜像签名策略中，对无版本的制品进行签名则传入{"untagged":true}

响应参数

状态码：200

成功更新策略信息

状态码：400

表 4-848 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-849 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-850 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-851 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-852 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/
policies/{policy_id}

{
  "name" : "test11",
  "scope_rules" : [ {
    "repo_scope_mode" : "regular",
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : "{\"untagged\":true}"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    }
  } ],
  "enabled" : true,
  "trigger" : {
    "trigger_settings" : {
      "cron" : ""
    },
    "type" : "manual"
  },
  "signature_method" : "KMS",
  "signature_algorithm" : "ECDSA_SHA_384",
  "signature_key" : "668985d5-919d-4c51-9293-eb846c78cbf0"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateInstanceSignPolicySolution {

    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.
```

 版权所有 © 华为云计算技术有限公司

```
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateInstanceSignPolicyRequest request = new UpdateInstanceSignPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        UpdateSignaturePolicyRequestBody body = new UpdateSignaturePolicyRequestBody();
        List<SignRuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new SignRuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<SignRuleSelector>> listScopeRulesScopeSelectors = new HashMap<>();
        listScopeRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<SignRuleSelector> listScopeRulesTagSelectors = new ArrayList<>();
        listScopeRulesTagSelectors.add(
            new SignRuleSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
                .withExtras("{"untagged":true}")
        );
        List<SignScopeRule> listbodyScopeRules = new ArrayList<>();
        listbodyScopeRules.add(
            new SignScopeRule()
                .withTagSelectors(listScopeRulesTagSelectors)
                .withScopeSelectors(listScopeRulesScopeSelectors)
                .withRepoScopeMode("regular")
        );
        TriggerSetting triggerSettingsTrigger = new TriggerSetting();
        triggerSettingsTrigger.withCron("");
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"))
            .withTriggerSettings(triggerSettingsTrigger);
        body.withScopeRules(listbodyScopeRules);
        body.withTrigger(triggerbody);
        body.withSignatureKey("668985d5-919d-4c51-9293-eb846c78cbf0");

body.withSignatureAlgorithm(UpdateSignaturePolicyRequestBody.UpdateSignaturePolicyRequestBody.fromV
alue("ECDSA_SHA_384"));

body.withSignatureMethod(UpdateSignaturePolicyRequestBody.UpdateSignaturePolicyRequestBody.fromValu
e("KMS"));
        body.withEnabled(true);
        body.withName("test11");
        request.withBody(body);
        try {
            UpdateInstanceSignPolicyResponse response = client.updateInstanceSignPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceSignPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        listScopeSelectorsScopeSelectors = [
            SignRuleSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsScopeRules = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsScopeRules = [
            SignRuleSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**",
                extras="{"untagged":true}"
            )
        ]
        listScopeRulesbody = [
            SignScopeRule(
                tag_selectors=listTagSelectorsScopeRules,
                scope_selectors=listScopeSelectorsScopeRules,
                repo_scope_mode="regular"
            )
        ]
        triggerSettingsTrigger = TriggerSetting(
            cron=""
        )
```

 版权所有 © 华为云计算技术有限公司

```
        triggerbody = TriggerConfig(
            type="manual",
            trigger_settings=triggerSettingsTrigger
        )
        request.body = UpdateSignaturePolicyRequestBody(
            scope_rules=listScopeRulesbody,
            trigger=triggerbody,
            signature_key="668985d5-919d-4c51-9293-eb846c78cbf0",
            signature_algorithm="ECDSA_SHA_384",
            signature_method="KMS",
            enabled=True,
            name="test11"
        )
        response = client.update_instance_sign_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceSignPolicyRequest{}
    request.InstanceId = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    var listScopeSelectorsScopeSelectors = []model.SignRuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsScopeRules = map[string][](model.SignRuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    extrasTagSelectors:= "{"untagged":true}"
    var listTagSelectorsScopeRules = []model.SignRuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
            Extras: &extrasTagSelectors,
        },
    }
    var listScopeRulesbody = []model.SignScopeRule{
        {
            TagSelectors: listTagSelectorsScopeRules,
            ScopeSelectors: listScopeSelectorsScopeRules,
            RepoScopeMode: "regular",
        },
    }
    cronTriggerSettings:= ""
    triggerSettingsTrigger := &model.TriggerSetting{
        Cron: &cronTriggerSettings,
    }
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
        TriggerSettings: triggerSettingsTrigger,
    }
    request.Body = &model.UpdateSignaturePolicyRequestBody{
        ScopeRules: listScopeRulesbody,
        Trigger: triggerbody,
        SignatureKey: "668985d5-919d-4c51-9293-eb846c78cbf0",
        SignatureAlgorithm:
model.GetUpdateSignaturePolicyRequestBodySignatureAlgorithmEnum().ECDSA_SHA_384,
        SignatureMethod: model.GetUpdateSignaturePolicyRequestBodySignatureMethodEnum().KMS,
        Enabled: true,
        Name: "test11",
    }
    response, err := client.UpdateInstanceSignPolicy(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	成功更新策略信息
400	错误的请求

 版权所有 © 华为云计算技术有限公司

状态码	描述
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.5 删除签名策略 - DeleteInstanceSignPolicy

功能介绍

删除签名策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteSignPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}

表 4-853 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID

请求参数

表 4-854 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

表 4-855 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-856 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-857 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-858 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-859 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceSignPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceSignPolicyRequest request = new DeleteInstanceSignPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            DeleteInstanceSignPolicyResponse response = client.deleteInstanceSignPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceSignPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.delete_instance_sign_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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.
```

 版权所有 © 华为云计算技术有限公司

```
    // In this example, AK and SK are 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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceSignPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.DeleteInstanceSignPolicy(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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

 版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.13.6 手动执行签名策略 - ExecuteInstanceSignPolicy

功能介绍

手动执行签名策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:executeSignPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions

表 4-860 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID

请求参数

表 4-861 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：202

表 4-862 响应 Body 参数

参数	参数类型	描述
id	Integer	执行ID

状态码：400

表 4-863 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-864 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-865 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-866 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-867 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions
```

响应示例

状态码：202

执行策略成功

```
{
  "id" : 34
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ExecuteInstanceSignPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ExecuteInstanceSignPolicyRequest request = new ExecuteInstanceSignPolicyRequest();
        request.withInstanceId("{instance_id}");
```

 版权所有 © 华为云计算技术有限公司

```
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ExecuteInstanceSignPolicyResponse response = client.executeInstanceSignPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ExecuteInstanceSignPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.execute_instance_sign_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
```

 版权所有 © 华为云计算技术有限公司

```
)

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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ExecuteInstanceSignPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ExecuteInstanceSignPolicy(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
202	执行策略成功
400	错误的请求
401	鉴权失败
403	禁止访问

 版权所有 © 华为云计算技术有限公司

状态码	描述
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.7 获取签名执行记录列表 - ListInstanceSignPolicyExecutions

功能介绍

获取签名执行记录列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listSignPolicyExecutions	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions

版权所有 © 华为云计算技术有限公司

表 4-868 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID

表 4-869 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-870 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-871 响应 Body 参数

参数	参数类型	描述
executions	Array of **Execution** objects	执行记录列表

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
total	Integer	执行记录总数

表 4-872 Execution

参数	参数类型	描述
id	Integer	记录ID
policy_id	Integer	策略ID
status	String	状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
status_text	String	状态详情
total	Integer	总数
failed	Integer	失败数
succeed	Integer	成功数
in_progress	Integer	进行中的数量
stopped	Integer	停止数
trigger	String	触发类型
created_at	String	创建时间
updated_at	String	更新时间

状态码：400

表 4-873 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-874 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-875 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-876 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-877 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions
```

响应示例

状态码：200

查询签名执行记录列表成功

```
{
  "total" : 10,
  "executions" : [ {
    "id" : 114204,
    "policy_id" : 12,
    "status" : "Failed",
    "status_text" : "failed to fetch artifacts",
    "total" : 0,
    "failed" : 0,
    "succeed" : 0,
    "in_progress" : 0,
    "stopped" : 0,
    "trigger" : "manual",
    "created_at" : "2020-12-03T12:21:38.51851Z",
    "updated_at" : "0001-01-01T00:00:00Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;
```


版权所有 © 华为云计算技术有限公司

```
public class ListInstanceSignPolicyExecutionsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceSignPolicyExecutionsRequest request = new ListInstanceSignPolicyExecutionsRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ListInstanceSignPolicyExecutionsResponse response = client.listInstanceSignPolicyExecutions(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()
```

 版权所有 © 华为云计算技术有限公司

```
    try:
        request = ListInstanceSignPolicyExecutionsRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.list_instance_sign_policy_executions(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceSignPolicyExecutionsRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ListInstanceSignPolicyExecutions(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

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	查询签名执行记录列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.8 获取签名执行记录任务的镜像列表 - ListInstanceSignPolicyExecTasks

功能介绍

获取签名执行记录任务的镜像列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listSignPolicyExecTasks	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks

表 4-878 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	策略ID
execution_id	是	Integer	签名策略执行记录ID

表 4-879 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-880 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-881 响应 Body 参数

参数	参数类型	描述
tasks	Array of **SignatureExecutionTask** objects	签名策略执行记录任务列表
total	Integer	签名策略执行记录任务总数

表 4-882 SignatureExecutionTask

参数	参数类型	描述
id	Long	执行任务ID
execution_id	Long	执行ID
job_id	String	jobId
status	String	状态 取值范围： • Initialized：初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止
status_text	String	执行任务状态信息
namespace	String	命名空间

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
repository	String	镜像仓库名称
created_at	String	创建时间
updated_at	String	更新时间

状态码：400

表 4-883 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-884 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-885 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-886 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-887 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 2,
  "tasks" : [ {
    "id" : 2,
    "execution_id" : 8,
    "namespace" : "root",
    "repository" : "libz",
    "job_id" : "2d7c874a7407d6c40f016254",
    "status" : "Succeed",
    "status_text" : "",
    "created_at" : "2020-12-01T07:06:35Z",
    "updated_at" : "2020-12-01T07:06:36Z"
  } ]
}
```


版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceSignPolicyExecTasksSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceSignPolicyExecTasksRequest request = new ListInstanceSignPolicyExecTasksRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        request.withExecutionId({execution_id});
        try {
            ListInstanceSignPolicyExecTasksResponse response = client.listInstanceSignPolicyExecTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceSignPolicyExecTasksRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        request.execution_id = {execution_id}
        response = client.list_instance_sign_policy_exec_tasks(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
```

 版权所有 © 华为云计算技术有限公司

```
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceSignPolicyExecTasksRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    request.ExecutionId = int32({execution_id})
    response, err := client.ListInstanceSignPolicyExecTasks(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.13.9 获取签名策略执行任务的镜像版本列表 - ListInstanceSignatureExecutionSubtasks

功能介绍

获取签名策略执行任务的镜像版本列表

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listSignPolicyExecSubTasks	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks

表 4-888 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	签名策略ID
execution_id	是	Integer	签名策略执行记录ID
task_id	是	Integer	任务ID

版权所有 © 华为云计算技术有限公司

表 4-889 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
status	否	String	状态 取值范围： • Initialized：初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止

请求参数

表 4-890 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-891 响应 Body 参数

参数	参数类型	描述
sub_tasks	Array of **SignatureExecutionSubTask** objects	签名策略执行记录子任务列表

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
total	Integer	签名策略执行记录子任务总数

表 4-892 SignatureExecutionSubTask

参数	参数类型	描述
id	Long	子任务ID
job_id	String	子任务详情
namespace	String	命名空间
repository	String	镜像仓库
tags	String	镜像版本
digest	String	镜像层sha256
status	String	子任务状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
status_text	String	状态详情信息
created_at	String	创建时间
updated_at	String	更新时间

状态码：400

表 4-893 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-894 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-895 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-896 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-897 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 2,
  "sub_tasks" : [ {
    "id" : 694,
    "job_id" : "cad53dfc907c167edc563ac0",
    "namespace" : "retention",
    "repository" : "cuda-vector-add",
    "tags" : "with-tools",
    "digest" : "sha256:ad6600f44974db05ec31e84d765923bc34ccc654d62dd915ba208b961fe83425",
    "status" : "Succeed",
    "status_text" : ""
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceSignatureExecutionSubtasksSolution {

    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}";
```

 版权所有 © 华为云计算技术有限公司

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceSignatureExecutionSubtasksRequest request = new
ListInstanceSignatureExecutionSubtasksRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        request.withExecutionId({execution_id});
        request.withTaskId({task_id});
        try {
            ListInstanceSignatureExecutionSubtasksResponse response =
client.listInstanceSignatureExecutionSubtasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceSignatureExecutionSubtasksRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        request.execution_id = {execution_id}
        request.task_id = {task_id}
```

 版权所有 © 华为云计算技术有限公司

```
        response = client.list_instance_signature_execution_subtasks(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceSignatureExecutionSubtasksRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    request.ExecutionId = int32({execution_id})
    request.TaskId = int32({task_id})
    response, err := client.ListInstanceSignatureExecutionSubtasks(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14 老化策略管理

4.14.1 创建老化策略 - CreateInstanceRetentionPolicy

功能介绍

创建老化策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:createRetentionPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies

表 4-898 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称

请求参数

表 4-899 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-900 请求 Body 参数

参数	是否必选	参数类型	描述
algorithm	是	String	算法，目前只支持or
enabled	是	Boolean	是否启用或者关闭所有retentionRules

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
rules	是	Array of **RetentionRule** objects	匹配规则，配置repo范围、tag范围以及作用规则
trigger	是	**TriggerConfig** object	触发方式
name	是	String	策略名称，由字母、汉字、数字、下划线（_）、中划线 (-)组成，1-256个字符。

表 4-901 RetentionRule

参数	是否必选	参数类型	描述
priority	是	Integer	优先级，预留字段，默认为0
disabled	否	Boolean	是否关闭此条规则
action	是	String	预留字段，目前只支持retain
template	是	String	规则模板类型，值为：latestPulledN，latestPushedK，nDaysSinceLastPush，nDaysSinceLastPull
params	是	Map<String,Object>	保留方式的参数，配套template使用，可参考请求示例
tag_selectors	是	Array of **RetentionSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**RetentionSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repo选择模式，前端使用，可选regular、selection

表 4-902 RetentionSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
decoration	是	String	选择器匹配类型 制品仓库选择器可选repoMatches，repoMatches制品版本可选matches，excludes
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

表 4-903 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-904 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

响应参数

状态码：201

版权所有 © 华为云计算技术有限公司

表 4-905 响应 Body 参数

参数	参数类型	描述
id	Integer	命名空间ID

状态码：400

表 4-906 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-907 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-908 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：404

表 4-909 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-910 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies

{
  "algorithm" : "or",
  "enabled" : true,
  "rules" : [ {
    "priority" : 0,
    "disabled" : false,
    "action" : "retain",
    "template" : "latestPushedK",
    "params" : {
      "latestPushedK" : 10
    },
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**"
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    },
    "repo_scope_mode" : "regular"
  } ],
```


版权所有 © 华为云计算技术有限公司

```
  "trigger" : {
    "type" : "manual"
  },
  "name" : "aa"
}
```

响应示例

状态码：201

创建成功

```
{
  "id" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateInstanceRetentionPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateInstanceRetentionPolicyRequest request = new CreateInstanceRetentionPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        CreateRetentionPolicyRequestBody body = new CreateRetentionPolicyRequestBody();
        TriggerConfig triggerbody = new TriggerConfig();
        triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"));
        List<RetentionSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
```

 版权所有 © 华为云计算技术有限公司

```
            new RetentionSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<RetentionSelector>> listRulesScopeSelectors = new HashMap<>();
        listRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<RetentionSelector> listRulesTagSelectors = new ArrayList<>();
        listRulesTagSelectors.add(
            new RetentionSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
        );
        Map<String, Object> listRulesParams = new HashMap<>();
        listRulesParams.put("latestPushedK", "10");
        List<RetentionRule> listbodyRules = new ArrayList<>();
        listbodyRules.add(
            new RetentionRule()
                .withPriority(0)
                .withDisabled(false)
                .withAction("retain")
                .withTemplate("latestPushedK")
                .withParams(listRulesParams)
                .withTagSelectors(listRulesTagSelectors)
                .withScopeSelectors(listRulesScopeSelectors)
                .withRepoScopeMode("regular")
        );
        body.withName("aa");
        body.withTrigger(triggerbody);
        body.withRules(listbodyRules);
        body.withEnabled(true);
        body.withAlgorithm("or");
        request.withBody(body);
        try {
            CreateInstanceRetentionPolicyResponse response = client.createInstanceRetentionPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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"]
```

 版权所有 © 华为云计算技术有限公司

```
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateInstanceRetentionPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        triggerbody = TriggerConfig(
            type="manual"
        )
        listScopeSelectorsScopeSelectors = [
            RetentionSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsRules = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsRules = [
            RetentionSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**"
            )
        ]
        listParamsRules = {
            "latestPushedK": "10"
        }
        listRulesbody = [
            RetentionRule(
                priority=0,
                disabled=False,
                action="retain",
                template="latestPushedK",
                params=listParamsRules,
                tag_selectors=listTagSelectorsRules,
                scope_selectors=listScopeSelectorsRules,
                repo_scope_mode="regular"
            )
        ]
        request.body = CreateRetentionPolicyRequestBody(
            name="aa",
            trigger=triggerbody,
            rules=listRulesbody,
            enabled=True,
            algorithm="or"
        )
        response = client.create_instance_retention_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"
```

 版权所有 © 华为云计算技术有限公司

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateInstanceRetentionPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
    }
    var listScopeSelectorsScopeSelectors = []model.RetentionSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsRules = Map{
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsRules = []model.RetentionSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    var listParamsRules = Map{
        "latestPushedK": "10",
    }
    disabledRules:= false
    var listRulesbody = []model.RetentionRule{
        {
```

```
            Priority: int32(0),
            Disabled: &disabledRules,
            Action: "retain",
            Template: "latestPushedK",
            Params: listParamsRules,
            TagSelectors: listTagSelectorsRules,
            ScopeSelectors: listScopeSelectorsRules,
            RepoScopeMode: "regular",
        },
    }
    request.Body = &model.CreateRetentionPolicyRequestBody{
        Name: "aa",
        Trigger: triggerbody,
        Rules: listRulesbody,
        Enabled: true,
        Algorithm: "or",
    }
    response, err := client.CreateInstanceRetentionPolicy(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	创建成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.2 获取老化策略列表 - ListInstanceRetentionPolicies

功能介绍

获取老化策略列表

接口约束

无

 版权所有 © 华为云计算技术有限公司

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listRetentionPolicies	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/retention/policies

表 4-911 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-912 Query 参数

参数	是否必选	参数类型	描述
name	否	String	老化策略名称
namespace_id	否	Integer	所属命名空间ID
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-913 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-914 响应 Body 参数

参数	参数类型	描述
total	Integer	策略总数
retentions	Array of **ImageRetention** objects	老化策略详情

表 4-915 ImageRetention

参数	参数类型	描述
id	Integer	老化策略ID
algorithm	String	算法
rules	Array of **RetentionRuleResponseBody** objects	匹配规则
trigger	**TriggerConfig** object	触发方式

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
enabled	Boolean	是否开启策略
name	String	策略名称
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名

表 4-916 RetentionRuleResponseBody

参数	参数类型	描述
id	Integer	镜像老化策略ID
priority	Integer	预留字段，目前只支持0
disabled	Boolean	是否关闭此条规则
action	String	预留字段，目前只支持retain
template	String	规则模板类型，值为：latestPulledN，latestPushedK，nDaysSinceLastPush，nDaysSinceLastPull
params	Map<String,Object>	保留方式的参数，配套template使用，可参考请求示例
tag_selectors	Array of **RetentionSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<**RetentionSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	String	repo选择模式，前端使用，可选regular、selection

表 4-917 RetentionSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型 制品仓库选择器可选repoMatches，repoMatches制品版本可选matches，excludes

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	String	预留字段

表 4-918 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	**TriggerSetting** object	触发设置，只有type为scheduled的时候，才需要设置

表 4-919 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

状态码：400

表 4-920 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-921 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-922 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-923 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-924 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/retention/policies
```

响应示例

状态码：200

查询老化策略列表成功

```
{
  "total" : 3,
  "retentions" : [ {
    "id" : 34,
    "algorithm" : "or",
    "rules" : [ {
      "id" : 0,
      "priority" : 0,
      "disabled" : false,
      "action" : "retain",
      "template" : "latestPulledN",
      "repo_scope_mode" : "regular",
      "params" : {
        "latestPulledN" : 111
      },
      "tag_selectors" : [ {
        "kind" : "doublestar",
        "decoration" : "matches",
        "pattern" : "abc",
        "extras" : ""
      } ],
      "scope_selectors" : {
        "repository" : [ {
          "kind" : "doublestar",
          "decoration" : "repoMatches",
          "pattern" : "abc",
          "extras" : ""
        } ]
      }
    } ],
    "trigger" : {
      "type" : "scheduled",
      "trigger_settings" : {
        "cron" : "0 19 14 * * ?"
      }
    },
    "enabled" : true,
    "name" : "abc",
    "namespace_id" : 51,
    "namespace_name" : "abc"
  } ]
}
```

 版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRetentionPoliciesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRetentionPoliciesRequest request = new ListInstanceRetentionPoliciesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListInstanceRetentionPoliciesResponse response = client.listInstanceRetentionPolicies(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 import *
```

 版权所有 © 华为云计算技术有限公司

```
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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRetentionPoliciesRequest()
        request.instance_id = "{instance_id}"
        response = client.list_instance_retention_policies(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRetentionPoliciesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListInstanceRetentionPolicies(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	查询老化策略列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.3 获取老化策略详情 - ShowInstanceRetentionPolicy

功能介绍

获取老化策略详情

接口约束

无

调用方法

请参见**如何调用API**。

版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:getRetentionPolicy	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}

表 4-925 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID

请求参数

表 4-926 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：200

表 4-927 响应 Body 参数

参数	参数类型	描述
id	Integer	老化策略ID
algorithm	String	算法
rules	Array of **RetentionRuleResponseBody** objects	匹配规则
trigger	**TriggerConfig** object	触发方式
enabled	Boolean	是否开启策略
name	String	策略名称
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名

表 4-928 RetentionRuleResponseBody

参数	参数类型	描述
id	Integer	镜像老化策略ID
priority	Integer	预留字段，目前只支持0
disabled	Boolean	是否关闭此条规则
action	String	预留字段，目前只支持retain
template	String	规则模板类型，值为：latestPulledN，latestPushedK，nDaysSinceLastPush，nDaysSinceLastPull
params	Map<String,Object>	保留方式的参数，配套template使用，可参考请求示例
tag_selectors	Array of **RetentionSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<**RetentionSelector**>>	制品仓库选择器，目前只支持repository且长度为1

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
repo_scope_mode	String	repo选择模式，前端使用，可选regular、selection

表 4-929 RetentionSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型 制品仓库选择器可选repoMatches，repoMatches制品版本可选matches，excludes
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	String	预留字段

表 4-930 TriggerConfig

参数	参数类型	描述
type	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-931 TriggerSetting

参数	参数类型	描述
cron	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-932 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-933 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-934 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-935 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-936 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}
```

响应示例

状态码：200

查询成功

```
{
  "id" : 36,
  "algorithm" : "or",
  "rules" : [ {
    "id" : 0,
    "priority" : 0,
    "disabled" : false,
    "action" : "retain",
    "template" : "latestPushedK",
    "params" : {
      "latestPushedK" : 5
    },
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : ""
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**",
        "extras" : ""
      } ]
    },
    "repo_scope_mode" : "regular"
```


版权所有 © 华为云计算技术有限公司

```
 } ],
 "trigger" : {
   "type" : "scheduled",
   "trigger_settings" : {
     "cron" : "0 5 5 * * ?"
   }
 },
 "enabled" : true,
 "name" : "test-create",
 "namespace_id" : 95,
 "namespace_name" : "ccc"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowInstanceRetentionPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceRetentionPolicyRequest request = new ShowInstanceRetentionPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ShowInstanceRetentionPolicyResponse response = client.showInstanceRetentionPolicy(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());
```

 版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceRetentionPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.show_instance_retention_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
```

 版权所有 © 华为云计算技术有限公司

```
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowInstanceRetentionPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ShowInstanceRetentionPolicy(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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.14.4 修改老化策略 - UpdateInstanceRetentionPolicy

功能介绍

修改老化策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateRetentionPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}

表 4-937 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-938 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-939 请求 Body 参数

参数	是否必选	参数类型	描述
algorithm	是	String	算法，目前只支持or
enabled	是	Boolean	是否启用或者关闭所有retentionRules
rules	是	Array of **RetentionRule** objects	匹配规则，数组长度只允许为1，配置repo范围、tag范围以及作用规则
trigger	是	**TriggerConfig** object	触发方式
name	是	String	策略名称，由字母、汉字、数字、下划线（_）、中划线 (-)组成，1-256个字符。

表 4-940 RetentionRule

参数	是否必选	参数类型	描述
priority	是	Integer	优先级，预留字段，默认为0
disabled	否	Boolean	是否关闭此条规则
action	是	String	预留字段，目前只支持retain
template	是	String	规则模板类型，值为：latestPulledN，latestPushedK，nDaysSinceLastPush，nDaysSinceLastPull
params	是	Map<String,Object>	保留方式的参数，配套template使用，可参考请求示例

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
tag_selectors	是	Array of **RetentionSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**RetentionSelector**>>	制品仓库选择器，目前只支持repository且长度为1
repo_scope_mode	是	String	repo选择模式，前端使用，可选regular、selection

表 4-941 RetentionSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型 制品仓库选择器可选repoMatches，repoMatches制品版本可选matches，excludes
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

版权所有 © 华为云计算技术有限公司

表 4-942 TriggerConfig

参数	是否必选	参数类型	描述
type	是	String	触发类型，老化规则只支持manual(手动)、scheduled(定时+手动)；同步策略支持manual(手动)、scheduled(定时+手动)、event_based(事件触发+手动);镜像签名支持manual(手动)、event_based(事件触发+手动)
trigger_settings	否	TriggerSetting object	触发设置，只有type为scheduled的时候，才需要设置

表 4-943 TriggerSetting

参数	是否必选	参数类型	描述
cron	否	String	定时设置，cron格式描述 '* * * * * *'。cron 基于 UTC 时间。

响应参数

状态码：201

成功更新老化策略信息

状态码：400

表 4-944 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-945 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-946 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-947 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-948 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/
policies/{policy_id}

{
  "algorithm" : "or",
  "enabled" : true,
  "rules" : [ {
    "priority" : 0,
    "action" : "retain",
    "repo_scope_mode" : "regular",
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**"
      } ]
    },
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**"
    } ],
    "disabled" : false,
    "params" : {
      "latestPushedK" : 10
    },
    "template" : "latestPushedK"
  } ],
  "trigger" : {
    "trigger_settings" : {
      "cron" : ""
    },
    "type" : "manual"
  },
  "name" : "test11"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateInstanceRetentionPolicySolution {
```

 版权所有 © 华为云计算技术有限公司

```
  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);

      SwrClient client = SwrClient.newBuilder()
            .withCredential(auth)
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
      UpdateInstanceRetentionPolicyRequest request = new UpdateInstanceRetentionPolicyRequest();
      request.withInstanceId("{instance_id}");
      request.withNamespaceName("{namespace_name}");
      request.withPolicyId({policy_id});
      UpdateRetentionPolicyRequestBody body = new UpdateRetentionPolicyRequestBody();
      TriggerSetting triggerSettingsTrigger = new TriggerSetting();
      triggerSettingsTrigger.withCron("");
      TriggerConfig triggerbody = new TriggerConfig();
      triggerbody.withType(TriggerConfig.TypeEnum.fromValue("manual"))
         .withTriggerSettings(triggerSettingsTrigger);
      List<RetentionSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
      listScopeSelectorsScopeSelectors.add(
         new RetentionSelector()
            .withKind("doublestar")
            .withDecoration("repoMatches")
            .withPattern("**")
      );
      Map<String, List<RetentionSelector>> listRulesScopeSelectors = new HashMap<>();
      listRulesScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
      List<RetentionSelector> listRulesTagSelectors = new ArrayList<>();
      listRulesTagSelectors.add(
         new RetentionSelector()
            .withKind("doublestar")
            .withDecoration("matches")
            .withPattern("**")
      );
      Map<String, Object> listRulesParams = new HashMap<>();
      listRulesParams.put("latestPushedK", "10");
      List<RetentionRule> listbodyRules = new ArrayList<>();
      listbodyRules.add(
         new RetentionRule()
            .withPriority(0)
            .withDisabled(false)
            .withAction("retain")
            .withTemplate("latestPushedK")
            .withParams(listRulesParams)
            .withTagSelectors(listRulesTagSelectors)
            .withScopeSelectors(listRulesScopeSelectors)
            .withRepoScopeMode("regular")
      );
      body.withName("test11");
      body.withTrigger(triggerbody);
      body.withRules(listbodyRules);
      body.withEnabled(true);
      body.withAlgorithm("or");
      request.withBody(body);
      try {
         UpdateInstanceRetentionPolicyResponse response = client.updateInstanceRetentionPolicy(request);
         System.out.println(response.toString());
      } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateInstanceRetentionPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        triggerSettingsTrigger = TriggerSetting(
            cron=""
        )
        triggerbody = TriggerConfig(
            type="manual",
            trigger_settings=triggerSettingsTrigger
        )
        listScopeSelectorsScopeSelectors = [
            RetentionSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsRules = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsRules = [
            RetentionSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**"
            )
        ]
```

 版权所有 © 华为云计算技术有限公司

```
        listParamsRules = {
            "latestPushedK": "10"
        }
        listRulesbody = [
            RetentionRule(
                priority=0,
                disabled=False,
                action="retain",
                template="latestPushedK",
                params=listParamsRules,
                tag_selectors=listTagSelectorsRules,
                scope_selectors=listScopeSelectorsRules,
                repo_scope_mode="regular"
            )
        ]
        request.body = UpdateRetentionPolicyRequestBody(
            name="test11",
            trigger=triggerbody,
            rules=listRulesbody,
            enabled=True,
            algorithm="or"
        )
        response = client.update_instance_retention_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()
```

 版权所有 © 华为云计算技术有限公司

```
    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateInstanceRetentionPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    cronTriggerSettings:= ""
    triggerSettingsTrigger := &model.TriggerSetting{
        Cron: &cronTriggerSettings,
    }
    triggerbody := &model.TriggerConfig{
        Type: model.GetTriggerConfigTypeEnum().MANUAL,
        TriggerSettings: triggerSettingsTrigger,
    }
    var listScopeSelectorsScopeSelectors = []model.RetentionSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsRules = map[string][](model.RetentionSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsRules = []model.RetentionSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    var listParamsRules = map[string]interface{}{
        "latestPushedK": "10",
    }
    disabledRules:= false
    var listRulesbody = []model.RetentionRule{
        {
            Priority: int32(0),
            Disabled: &disabledRules,
            Action: "retain",
            Template: "latestPushedK",
            Params: listParamsRules,
            TagSelectors: listTagSelectorsRules,
            ScopeSelectors: listScopeSelectorsRules,
            RepoScopeMode: "regular",
        },
    }
    request.Body = &model.UpdateRetentionPolicyRequestBody{
        Name: "test11",
        Trigger: triggerbody,
        Rules: listRulesbody,
        Enabled: true,
        Algorithm: "or",
    }
    response, err := client.UpdateInstanceRetentionPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	成功更新老化策略信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.5 删除老化策略 - DeleteInstanceRetentionPolicy

功能介绍

删除老化策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteRetentionPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}

表 4-949 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID

请求参数

表 4-950 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：204

删除成功

状态码：400

版权所有 © 华为云计算技术有限公司

表 4-951 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-952 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-953 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-954 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-955 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteInstanceRetentionPolicySolution {

    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");
```


版权所有 © 华为云计算技术有限公司

```
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteInstanceRetentionPolicyRequest request = new DeleteInstanceRetentionPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            DeleteInstanceRetentionPolicyResponse response = client.deleteInstanceRetentionPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceRetentionPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.delete_instance_retention_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteInstanceRetentionPolicyRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.DeleteInstanceRetentionPolicy(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
204	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.6 执行老化策略 - ExecuteInstanceRetentionPolicy

功能介绍

执行老化策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:executeRetentionPolicy	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions

表 4-956 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID

请求参数

表 4-957 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-958 请求 Body 参数

参数	是否必选	参数类型	描述
dry_run	是	Boolean	是否模拟运行

响应参数

状态码：200

表 4-959 响应 Body 参数

参数	参数类型	描述
id	Integer	老化策略执行ID

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-960 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-961 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-962 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-963 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-964 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions

{
  "dry_run" : false
}
```

响应示例

状态码：200

执行老化策略成功

```
{
  "id" : 34
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
```


版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.swr.v2.model.*;


public class ExecuteInstanceRetentionPolicySolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ExecuteInstanceRetentionPolicyRequest request = new ExecuteInstanceRetentionPolicyRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        ExecuteRetentionPolicyRequestBody body = new ExecuteRetentionPolicyRequestBody();
        body.withDryRun(false);
        request.withBody(body);
        try {
            ExecuteInstanceRetentionPolicyResponse response = client.executeInstanceRetentionPolicy(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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)
```

 版权所有 © 华为云计算技术有限公司

```
    client = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ExecuteInstanceRetentionPolicyRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        request.body = ExecuteRetentionPolicyRequestBody(
            dry_run=False
        )
        response = client.execute_instance_retention_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ExecuteInstanceRetentionPolicyRequest{}
```

 版权所有 © 华为云计算技术有限公司

```
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    request.Body = &model.ExecuteRetentionPolicyRequestBody{
        DryRun: false,
    }
    response, err := client.ExecuteInstanceRetentionPolicy(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	执行老化策略成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.7 获取老化策略执行记录列表 - ListInstanceRetentionPolicyExecutions

功能介绍

获取老化策略执行记录列表

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listRetentionPolicyExecutions	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions

表 4-965 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID

表 4-966 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

请求参数

表 4-967 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-968 响应 Body 参数

参数	参数类型	描述
total	Integer	总数
executions	Array of **RetentionExecution** objects	老化策略执行列表

表 4-969 RetentionExecution

参数	参数类型	描述
id	Integer	老化策略执行记录ID
policy_id	Integer	老化策略ID
dry_run	Boolean	是否模拟运行
status	String	状态 取值范围： • InProgress：运行中 • Failed：错误 • Success：成功 • Stopped：停止
trigger	String	触发类型，scheduled manual
start_time	String	开始时间
end_time	String	结束时间

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-970 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-971 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-972 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-973 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-974 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions
```

响应示例

状态码：200

查询老化策略执行记录列表成功

```
{
  "total" : 1,
  "executions" : [ {
    "id" : 112,
    "policy_id" : 38,
    "start_time" : "2021-02-01T03:49:58Z",
    "end_time" : "2021-02-01T03:49:58Z",
    "status" : "Success",
    "trigger" : "manual",
    "dry_run" : false
  } ]
}
```

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;
```


版权所有 © 华为云计算技术有限公司

```
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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRetentionPolicyExecutionsSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRetentionPolicyExecutionsRequest request = new ListInstanceRetentionPolicyExecutionsRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        try {
            ListInstanceRetentionPolicyExecutionsResponse response = client.listInstanceRetentionPolicyExecutions(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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"]
```

 版权所有 © 华为云计算技术有限公司

```
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListInstanceRetentionPolicyExecutionsRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        response = client.list_instance_retention_policy_executions(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)
```

 版权所有 © 华为云计算技术有限公司

```
    request := &model.ListInstanceRetentionPolicyExecutionsRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    response, err := client.ListInstanceRetentionPolicyExecutions(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	查询老化策略执行记录列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.8 获取老化策略执行任务的镜像列表 - ListInstanceRetentionPolicyExecTasks

功能介绍

获取老化策略执行任务的镜像列表

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listRetentionPolicyExecTasks	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions/{execution_id}/tasks

表 4-975 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID
execution_id	是	Integer	老化策略执行记录ID

表 4-976 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
status	否	String	状态 取值范围： • Initialized：初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止

请求参数

表 4-977 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-978 响应 Body 参数

参数	参数类型	描述
tasks	Array of **Task** objects	老化策略执行记录任务列表
total	Integer	老化策略执行记录任务总数

表 4-979 Task

参数	参数类型	描述
id	Integer	老化策略执行记录任务ID

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
execution_id	Integer	老化策略执行记录ID
repository	String	制品仓库名称
job_id	String	任务名称
status	String	状态 取值范围： ● Pending：准备中 ● InProgress：运行中 ● Succeed：运行成功 ● Failed：运行失败
status_code	Integer	执行状态 取值范围： ● 0：准备中 ● 1：调度中 ● 2：运行中 ● 3：已结束
status_revision	Integer	状态修订
start_time	String	开始时间
end_time	String	结束时间
total	Integer	版本总数
retained	Integer	保留版本总数

状态码：400

表 4-980 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-981 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-982 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-983 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-984 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions/{execution_id}/tasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 1,
  "tasks" : [ {
    "id" : 2439,
    "execution_id" : 113,
    "repository" : "cuda-vector-add",
    "job_id" : "cad53dfc907c167edc563ac0",
    "status" : "Succeed",
    "status_code" : 3,
    "status_revision" : 1612151612,
    "start_time" : "2021-02-01T03:53:32Z",
    "end_time" : "2021-02-01T03:53:41Z",
    "total" : 1,
    "retained" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRetentionPolicyExecTasksSolution {

    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");
```


版权所有 © 华为云计算技术有限公司

```
        String projectId = "{project_id}";

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

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRetentionPolicyExecTasksRequest request = new
ListInstanceRetentionPolicyExecTasksRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        request.withExecutionId({execution_id});
        try {
            ListInstanceRetentionPolicyExecTasksResponse response =
client.listInstanceRetentionPolicyExecTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRetentionPolicyExecTasksRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        request.execution_id = {execution_id}
        response = client.list_instance_retention_policy_exec_tasks(request)
```

 版权所有 © 华为云计算技术有限公司

```
        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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRetentionPolicyExecTasksRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    request.ExecutionId = int32({execution_id})
    response, err := client.ListInstanceRetentionPolicyExecTasks(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.14.9 获取老化策略执行任务的镜像版本列表 - ListInstanceRetentionPolicyExecSubTasks

功能介绍

获取老化策略执行任务的镜像版本列表

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listRetentionPolicyExecSubTasks	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks

表 4-985 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
policy_id	是	Integer	老化策略ID
execution_id	是	Integer	老化策略执行记录ID
task_id	是	Integer	任务ID

表 4-986 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
status	否	String	状态 取值范围： • Initialized：初始化 • Pending：等待调度 • InProgress：运行中 • Failed：错误 • Succeed：成功 • Stopped：停止

请求参数

表 4-987 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-988 响应 Body 参数

参数	参数类型	描述
sub_tasks	Array of **Subtask** objects	老化策略执行记录子任务列表
total	Integer	老化策略执行记录子任务总数

表 4-989 Subtask

参数	参数类型	描述
id	Integer	子任务ID
job_id	String	内部任务ID
namespace	String	命名空间名

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
repository	String	制品仓库
tag	String	制品版本
digest	String	sha256值
action	String	老化动作， DEL
status	String	状态 取值范围： ● Initialized：初始化 ● Pending：等待调度 ● InProgress：运行中 ● Failed：错误 ● Succeed：成功 ● Stopped：停止
status_text	String	状态信息
op_time	String	开始时间

状态码：400

表 4-990 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-991 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-992 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-993 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-994 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/
policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks
```

响应示例

状态码：200

查询成功

```
{
  "total" : 2,
  "sub_tasks" : [ {
    "id" : 694,
    "job_id" : "cad53dfc907c167edc563ac0",
    "namespace" : "retention",
    "repository" : "cuda-vector-add",
    "tag" : "with-tools",
    "digest" : "sha256:ad6600f44974db05ec31e84d765923bc34ccc654d62dd915ba208b961fe83425",
    "status" : "Succeed",
    "status_text" : "",
    "op_time" : "2021-02-01T03:53:41.4285Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceRetentionPolicyExecSubTasksSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceRetentionPolicyExecSubTasksRequest request = new
ListInstanceRetentionPolicyExecSubTasksRequest();
```

 版权所有 © 华为云计算技术有限公司

```
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withPolicyId({policy_id});
        request.withExecutionId({execution_id});
        request.withTaskId({task_id});
        try {
            ListInstanceRetentionPolicyExecSubTasksResponse response =
client.listInstanceRetentionPolicyExecSubTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceRetentionPolicyExecSubTasksRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.policy_id = {policy_id}
        request.execution_id = {execution_id}
        request.task_id = {task_id}
        response = client.list_instance_retention_policy_exec_sub_tasks(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
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceRetentionPolicyExecSubTasksRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.PolicyId = int32({policy_id})
    request.ExecutionId = int32({execution_id})
    request.TaskId = int32({task_id})
    response, err := client.ListInstanceRetentionPolicyExecSubTasks(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	查询成功

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.15 镜像版本不可变

4.15.1 创建不可变 Tag 策略 - CreateImmutableRule

功能介绍

创建不可变Tag策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:createImmutableRule	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules

表 4-995 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称

请求参数

表 4-996 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-997 请求 Body 参数

参数	是否必选	参数类型	描述
priority	否	Integer	优先级，默认值为0
disabled	否	Boolean	不可变策略是否生效，默认值为false
action	否	String	预留字段，支持填immutable

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
template	否	String	预留字段，支持填immutable_template
tag_selectors	是	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1
scope_selectors	是	Map<String,Array<**RuleSelector**>>	制品仓库选择器，目前只支持repository且长度为1

表 4-998 RuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

响应参数

状态码：201

表 4-999 响应 Body 参数

参数	参数类型	描述
id	Integer	策略ID

版权所有 © 华为云计算技术有限公司

状态码：400

表 4-1000 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1001 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1002 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1003 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1004 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
immutabletagrules

{
  "priority" : 0,
  "disabled" : true,
  "action" : "immutable",
  "template" : "immutable_template",
  "tag_selectors" : [ {
    "kind" : "doublestar",
    "decoration" : "matches",
    "pattern" : "**"
  } ],
  "scope_selectors" : {
    "repository" : [ {
      "kind" : "doublestar",
      "decoration" : "repoMatches",
      "pattern" : "**"
    } ]
  }
}
```

响应示例

状态码：201

创建策略成功

```
{
  "id" : 34
}
```

 版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateImmutableRuleSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateImmutableRuleRequest request = new CreateImmutableRuleRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        CreateImmutableRuleBody body = new CreateImmutableRuleBody();
        List<RuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<RuleSelector>> listbodyScopeSelectors = new HashMap<>();
        listbodyScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<RuleSelector> listbodyTagSelectors = new ArrayList<>();
        listbodyTagSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
        );
        body.withScopeSelectors(listbodyScopeSelectors);
        body.withTagSelectors(listbodyTagSelectors);
        body.withTemplate("immutable_template");
        body.withAction("immutable");
        body.withDisabled(true);
        body.withPriority(0);
```

 版权所有 © 华为云计算技术有限公司

```
        request.withBody(body);
        try {
            CreateImmutableRuleResponse response = client.createImmutableRule(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateImmutableRuleRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        listScopeSelectorsScopeSelectors = [
            RuleSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsbody = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsbody = [
            RuleSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**"
            )
        ]
        request.body = CreateImmutableRuleBody(
            scope_selectors=listScopeSelectorsbody,
            tag_selectors=listTagSelectorsbody,
```

 版权所有 © 华为云计算技术有限公司

```
            template="immutable_template",
            action="immutable",
            disabled=True,
            priority=0
        )
        response = client.create_immutable_rule(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.CreateImmutableRuleRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    var listScopeSelectorsScopeSelectors = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
```

 版权所有 © 华为云计算技术有限公司

```
    var listScopeSelectorsbody = map[string][](model.RuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsbody = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    templateCreateImmutableRuleBody:= "immutable_template"
    actionCreateImmutableRuleBody:= "immutable"
    disabledCreateImmutableRuleBody:= true
    priorityCreateImmutableRuleBody:= int32(0)
    request.Body = &model.CreateImmutableRuleBody{
        ScopeSelectors: listScopeSelectorsbody,
        TagSelectors: listTagSelectorsbody,
        Template: &templateCreateImmutableRuleBody,
        Action: &actionCreateImmutableRuleBody,
        Disabled: &disabledCreateImmutableRuleBody,
        Priority: &priorityCreateImmutableRuleBody,
    }
    response, err := client.CreateImmutableRule(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	创建策略成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.15.2 获取不可变 Tag 策略列表 - ListImmutableRules

功能介绍

获取不可变Tag策略列表

 版权所有 © 华为云计算技术有限公司

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:listImmutableRules	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/immutabletagrules

表 4-1005 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-1006 Query 参数

参数	是否必选	参数类型	描述
namespace_id	否	Integer	所属命名空间ID
offset	否	Integer	起始索引，默认值为0。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
limit	否	Integer	返回条数，默认为10，最大值为100。**注意：offset和limit参数需要配套使用，offset必须为0或者为limit的倍数。**

请求参数

表 4-1007 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-1008 响应 Body 参数

参数	参数类型	描述
immutable_rules	Array of **ImmutableRule** objects	策略列表
total	Integer	策略总数

表 4-1009 ImmutableRule

参数	参数类型	描述
id	Integer	不可变策略ID
namespace_id	Integer	命名空间ID
namespace_name	String	命名空间名称
priority	Integer	优先级，保留字段
disabled	Boolean	不可变策略是否生效
action	String	预留字段，仅支持填immutable

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
template	String	预留字段，仅支持填immutable_template
tag_selectors	Array of RuleSelector objects	制品版本选择器，目前只支持长度为1
scope_selectors	Map<String,Array<RuleSelector>>	制品仓库选择器，目前只支持repository且长度为1

表 4-1010 RuleSelector

参数	参数类型	描述
kind	String	匹配类型，目前只支持doublestar
decoration	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？：匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	String	预留字段

状态码：400

表 4-1011 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-1012 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1013 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1014 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1015 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/immutabletagrules
```

响应示例

状态码：200

查询成功

```
{
  "total" : 1,
  "immutable_rules" : [ {
    "id" : 1,
    "namespace_id" : 1,
    "namespace_name" : "test",
    "priority" : 0,
    "disabled" : false,
    "action" : "immutable",
    "template" : "immutable_template",
    "tag_selectors" : [ {
      "kind" : "doublestar",
      "decoration" : "matches",
      "pattern" : "**",
      "extras" : ""
    } ],
    "scope_selectors" : {
      "repository" : [ {
        "kind" : "doublestar",
        "decoration" : "repoMatches",
        "pattern" : "**",
        "extras" : ""
      } ]
    }
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListImmutableRulesSolution {
```

 版权所有 © 华为云计算技术有限公司

```
    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListImmutableRulesRequest request = new ListImmutableRulesRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListImmutableRulesResponse response = client.listImmutableRules(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListImmutableRulesRequest()
        request.instance_id = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
        response = client.list_immutable_rules(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListImmutableRulesRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListImmutableRules(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.15.3 修改不可变 Tag 策略 - UpdateImmutableRule

功能介绍

修改不可变Tag策略

接口约束

无

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:updateImmutableRule	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

版权所有 © 华为云计算技术有限公司

URI

PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules/{immutable_rule_id}

表 4-1016 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
immutable_rule_id	是	Integer	不可变Tag策略ID

请求参数

表 4-1017 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-1018 请求 Body 参数

参数	是否必选	参数类型	描述
priority	否	Integer	优先级，默认值为0
disabled	否	Boolean	不可变策略是否生效，默认值为false
action	否	String	预留字段，支持填immutable
template	否	String	预留字段，支持填immutable_template
tag_selectors	是	Array of **RuleSelector** objects	制品版本选择器，目前只支持长度为1

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
scope_selectors	是	Map<String,Array<RuleSelector>>	制品仓库选择器，目前只支持repository且长度为1

表 4-1019 RuleSelector

参数	是否必选	参数类型	描述
kind	是	String	匹配类型，目前只支持doublestar
decoration	是	String	选择器匹配类型。tag_selectors场景支持matches；scope_selectors场景支持repoMatches
pattern	是	String	选择器匹配样式，最大长度512。支持正则表达式，正则表达式规则可填写如 nginx-* ，{repo1，repo2} 等，其中： ● ： 匹配不包含 '/' 的任何字段。 **：匹配包含 '/' 的任何字段。 ？： 匹配任何单个非 '/' 的字符。 {选项1,选项2,...}：同时匹配多个选项。
extras	否	String	预留字段

响应参数

状态码：200

成功更新不可变Tag策略信息

状态码：400

表 4-1020 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1021 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1022 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1023 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：500

表 4-1024 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
immutabletagrules/{immutable_rule_id}

{
  "priority" : 0,
  "action" : "immutable",
  "disabled" : false,
  "template" : "immutable_template",
  "scope_selectors" : {
    "repository" : [ {
      "kind" : "doublestar",
      "decoration" : "repoMatches",
      "pattern" : "**"
    } ]
  },
  "tag_selectors" : [ {
    "kind" : "doublestar",
    "decoration" : "matches",
    "pattern" : "**"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;

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


版权所有 © 华为云计算技术有限公司

```
import java.util.Map;
import java.util.HashMap;

public class UpdateImmutableRuleSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateImmutableRuleRequest request = new UpdateImmutableRuleRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withImmutableRuleId({immutable_rule_id});
        UpdateImmutableRuleBody body = new UpdateImmutableRuleBody();
        List<RuleSelector> listScopeSelectorsScopeSelectors = new ArrayList<>();
        listScopeSelectorsScopeSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("repoMatches")
                .withPattern("**")
        );
        Map<String, List<RuleSelector>> listbodyScopeSelectors = new HashMap<>();
        listbodyScopeSelectors.put("repository", listScopeSelectorsScopeSelectors);
        List<RuleSelector> listbodyTagSelectors = new ArrayList<>();
        listbodyTagSelectors.add(
            new RuleSelector()
                .withKind("doublestar")
                .withDecoration("matches")
                .withPattern("**")
        );
        body.withScopeSelectors(listbodyScopeSelectors);
        body.withTagSelectors(listbodyTagSelectors);
        body.withTemplate("immutable_template");
        body.withAction("immutable");
        body.withDisabled(false);
        body.withPriority(0);
        request.withBody(body);
        try {
            UpdateImmutableRuleResponse response = client.updateImmutableRule(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateImmutableRuleRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.immutable_rule_id = {immutable_rule_id}
        listScopeSelectorsScopeSelectors = [
            RuleSelector(
                kind="doublestar",
                decoration="repoMatches",
                pattern="**"
            )
        ]
        listScopeSelectorsbody = {
            "repository": listScopeSelectorsScopeSelectors
        }
        listTagSelectorsbody = [
            RuleSelector(
                kind="doublestar",
                decoration="matches",
                pattern="**"
            )
        ]
        request.body = UpdateImmutableRuleBody(
            scope_selectors=listScopeSelectorsbody,
            tag_selectors=listTagSelectorsbody,
            template="immutable_template",
            action="immutable",
            disabled=False,
            priority=0
        )
        response = client.update_immutable_rule(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
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateImmutableRuleRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.ImmutableRuleId = int32({immutable_rule_id})
    var listScopeSelectorsScopeSelectors = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "repoMatches",
            Pattern: "**",
        },
    }
    var listScopeSelectorsbody = map[string][](model.RuleSelector){
        "repository": listScopeSelectorsScopeSelectors,
    }
    var listTagSelectorsbody = []model.RuleSelector{
        {
            Kind: "doublestar",
            Decoration: "matches",
            Pattern: "**",
        },
    }
    templateUpdateImmutableRuleBody:= "immutable_template"
    actionUpdateImmutableRuleBody:= "immutable"
    disabledUpdateImmutableRuleBody:= false
    priorityUpdateImmutableRuleBody:= int32(0)
    request.Body = &model.UpdateImmutableRuleBody{
        ScopeSelectors: listScopeSelectorsbody,
```

 版权所有 © 华为云计算技术有限公司

```
        TagSelectors: listTagSelectorsbody,
        Template: &templateUpdateImmutableRuleBody,
        Action: &actionUpdateImmutableRuleBody,
        Disabled: &disabledUpdateImmutableRuleBody,
        Priority: &priorityUpdateImmutableRuleBody,
    }
    response, err := client.UpdateImmutableRule(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	成功更新不可变Tag策略信息
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.15.4 删除不可变 Tag 策略 - DeleteImmutableRule

功能介绍

删除不可变Tag策略

接口约束

无

调用方法

请参见**如何调用API**。

 版权所有 © 华为云计算技术有限公司

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:deleteImmutableRule	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules/{immutable_rule_id}

表 4-1025 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
immutable_rule_id	是	Integer	不可变Tag策略ID

请求参数

表 4-1026 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：204

删除成功

状态码：400

表 4-1027 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1028 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1029 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-1030 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1031 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
immutabletagrules/{immutable_rule_id}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class DeleteImmutableRuleSolution {

    public static void main(String[] args) {
```

 版权所有 © 华为云计算技术有限公司

```
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteImmutableRuleRequest request = new DeleteImmutableRuleRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withImmutableRuleId({immutable_rule_id});
        try {
            DeleteImmutableRuleResponse response = client.deleteImmutableRule(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteImmutableRuleRequest()
        request.instance_id = "{instance_id}"
```

 版权所有 © 华为云计算技术有限公司

```
        request.namespace_name = "{namespace_name}"
        request.immutable_rule_id = {immutable_rule_id}
        response = client.delete_immutable_rule(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.DeleteImmutableRuleRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.ImmutableRuleId = int32({immutable_rule_id})
    response, err := client.DeleteImmutableRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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	删除成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.16 制品扫描

4.16.1 手动启动制品扫描 - StartManualScanning

功能介绍

手动启动制品扫描

接口约束

此接口只在企业仓库实例版本25.7.20以上的版本才支持

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:repository:startManualScanning	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/scan

表 4-1032 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

请求参数

表 4-1033 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

版权所有 © 华为云计算技术有限公司

响应参数

状态码：202

启动制品扫描请求已接受

状态码：400

表 4-1034 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1035 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1036 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

版权所有 © 华为云计算技术有限公司

表 4-1037 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1038 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/
repositories/{repository_name}/artifacts/{reference}/scan
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class StartManualScanningSolution {

    public static void main(String[] args) {
```

 版权所有 © 华为云计算技术有限公司

```
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are 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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        StartManualScanningRequest request = new StartManualScanningRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        request.withReference("{reference}");
        try {
            StartManualScanningResponse response = client.startManualScanning(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartManualScanningRequest()
```

 版权所有 © 华为云计算技术有限公司

```
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        response = client.start_manual_scanning(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.StartManualScanningRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    response, err := client.StartManualScanning(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

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
202	启动制品扫描请求已接受
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.16.2 获取制品扫描的漏洞信息 - ListInstanceArtifactVulnerabilities

功能介绍

获取制品扫描的漏洞信息

接口约束

此接口只在企业仓库实例版本25.7.20以上的版本才支持

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，当前API调用无需身份策略权限。

版权所有 © 华为云计算技术有限公司

URI

GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/vulnerabilities

表 4-1039 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
namespace_name	是	String	命名空间名称
repository_name	是	String	制品仓库名称, 如果您的镜像名称repository_name参数中有/，请先把 / 替换成 %2F 再进行请求。注意：使用curl命令请求，需要把 / 替换成 %252F(即 %2F 的 % 被编码为 %25)再次请求。
reference	是	String	制品摘要

请求参数

表 4-1040 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-1041 响应 Body 参数

参数	参数类型	描述
{自定义key}	Map<String,**VulnerabilityReports**>	支持漏洞报告类型为application/vnd.security.vulnerability.report; version=1.1的漏洞报告

版权所有 © 华为云计算技术有限公司

表 4-1042 VulnerabilityReports

参数	参数类型	描述
generated_at	String	漏洞报告生成时间
severity	String	制品扫描报告的总体严重程度，None(无评分), Low(低危), Medium(中危), High(高危), Critical(严重), Security(安全)
scanner	**Scanner** object	扫描器的信息
vulnerabilities	Array of **vulnerability** objects	漏洞列表

表 4-1043 Scanner

参数	参数类型	描述
name	String	扫描器的名称
vendor	String	扫描器的提供商
version	String	扫描器的版本号

表 4-1044 vulnerability

参数	参数类型	描述
id	String	漏洞ID
package	String	含有漏洞的软件包名称
version	String	含有漏洞的软件包版本号
fix_version	String	漏洞修复的软件包版本号
severity	String	漏洞的严重程度，Low(低危), Medium(中危), High(高危), Critical(严重)
description	String	漏洞的描述信息
links	Array of strings	漏洞的相关链接信息
artifact_digests	Array of strings	哪些镜像层包含此漏洞
preferred_cvss	**preferred_cvss** object	基于CVSS3和CVSS2的漏洞项评分及攻击向量
cwe_ids	Array of strings	与此漏洞相关的CWE ID列表

版权所有 © 华为云计算技术有限公司

表 4-1045 preferred_cvss

参数	参数类型	描述
score_v3	Number	漏洞的CVSS-3评分
score_v2	Number	漏洞的CVSS-2评分
vector_v3	String	漏洞的CVSS-3攻击向量
vector_v2	String	漏洞的CVSS-2攻击向量

状态码：400

表 4-1046 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1047 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1048 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1049 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1050 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/vulnerabilities
```

响应示例

状态码：200

查询制品的漏洞信息成功

```
{
  "application/vnd.security.vulnerability.report; version=1.1" : {
    "generated_at" : "2025-09-12:06:44:31",
    "scanner" : {
      "name" : "HSS",
      "vendor" : "HSS",
      "version" : "v5"
```


版权所有 © 华为云计算技术有限公司

```
    },
    "severity" : "High",
    "vulnerabilities" : [ {
      "id" : "CVE-2020-2755",
      "package" : "openjdk-8-jre",
      "version" : "8u181-b13-2~deb9u1",
      "fix_version" : "",
      "severity" : "High",
      "description" : "Vulnerability in the Java SE, Java SE Embedded product of Oracle Java SE (component: Scripting). Supported versions that are affected are Java SE: 8u241, 11.0.6 and 14; Java SE Embedded: 8u241. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Java SE, Java SE Embedded. Successful attacks of this vulnerability can result in unauthorized ability to cause a partial denial of service (partial DOS) of Java SE, Java SE Embedded. Note: Applies to client and server deployment of Java. This vulnerability can be exploited through sandboxed Java Web Start applications and sandboxed Java applets. It can also be exploited by supplying data to APIs in the specified Component without using sandboxed Java Web Start applications or sandboxed Java applets, such as through a web service. CVSS 3.0 Base Score 3.7 (Availability impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A:L).",
      "links" : [ "https://security-tracker.debian.org/tracker/DSA-4668-1" ],
      "artifact_digests" : [ "sha256:a48d150ffd2faf9ea63217a7774a75cd3b4a252413810c2239d1ee257efc9e13" ],
      "preferred_cvss" : {
        "score_v3" : 3.700000047683716,
        "score_v2" : 3.7,
        "vector_v3" : "",
        "vector_v2" : ""
      },
      "cwe_ids" : [ "" ]
    } ]
  }
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListInstanceArtifactVulnerabilitiesSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
```

 版权所有 © 华为云计算技术有限公司

```
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ListInstanceArtifactVulnerabilitiesRequest request = new ListInstanceArtifactVulnerabilitiesRequest();
        request.withInstanceId("{instance_id}");
        request.withNamespaceName("{namespace_name}");
        request.withRepositoryName("{repository_name}");
        request.withReference("{reference}");
        try {
            ListInstanceArtifactVulnerabilitiesResponse response =
client.listInstanceArtifactVulnerabilities(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListInstanceArtifactVulnerabilitiesRequest()
        request.instance_id = "{instance_id}"
        request.namespace_name = "{namespace_name}"
        request.repository_name = "{repository_name}"
        request.reference = "{reference}"
        response = client.list_instance_artifact_vulnerabilities(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

 版权所有 © 华为云计算技术有限公司

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListInstanceArtifactVulnerabilitiesRequest{}
    request.InstanceId = "{instance_id}"
    request.NamespaceName = "{namespace_name}"
    request.RepositoryName = "{repository_name}"
    request.Reference = "{reference}"
    response, err := client.ListInstanceArtifactVulnerabilities(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。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	查询制品的漏洞信息成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.17 制品清理

4.17.1 查询制品清理的任务列表 - ListGcTasks

功能介绍

查询制品清理的任务列表

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:listGcTasks	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/system/gc

表 4-1051 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

表 4-1052 Query 参数

参数	是否必选	参数类型	描述
marker	否	String	分页查询时的查询标记，使用上一次接口调用返回的next_marker值，默认值从第一条数据查询。**注意：marker和limit参数需要配套使用。**
limit	否	Integer	条目数量，用于分页查询，默认值为10，最大值为100
status	否	String	任务状态，Success：已完成，Stopped：已停止，Running：清理中，Pending：排队中，Error：失败。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-1053 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-1054 响应 Body 参数

参数	参数类型	描述
page_info	**PageInfo** object	分页信息
tasks	Array of **GcTask** objects	gc任务列表详情

表 4-1055 PageInfo

参数	参数类型	描述
next_marker	String	下一次分页查询的起始标记。如果未返回该值，则表示数据已查询完毕

表 4-1056 GcTask

参数	参数类型	描述
id	Integer	gc任务的ID。
job_name	String	gc任务的名称。
job_kind	String	任务类型，MANUAL：手动执行，SCHEDULE：定时执行。
job_parameters	**JobParameters** object	gc任务的作业参数。
job_status	String	gc任务的状态，Success：已完成，Stopped：已停止，Running：清理中，Pending：排队中，Error：失败。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
creation_time	String	gc任务的创建时间。
update_time	String	gc任务的更新时间。

表 4-1057 JobParameters

参数	参数类型	描述
delete_untagged	Boolean	是否删除无tag的制品；默认为false，不删除无tag的制品。
workers	Integer	并行执行制品清理任务的工作者数量，最小值为1，最大值为5。
dry_run	Boolean	是否模拟执行任务；默认值为false，非模拟运行。
purged_blobs	Integer	清理镜像层的个数。
purged_manifests	Integer	清理镜像manifest的个数。
freed_space	Integer	释放的镜像存储空间大小(单位为字节)。

状态码：400

表 4-1058 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1059 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1060 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1061 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1062 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc
```

响应示例

状态码：200

查询制品清理的任务列表成功

```
{
  "page_info" : {
    "next_marker" : "eyJpZCI6MTIzNH0="
  },
  "tasks" : [ {
    "id" : 1001,
    "job_name" : "daily-gc-task",
    "job_kind" : "SCHEDULE",
    "job_parameters" : {
      "delete_untagged" : true,
      "dry_run" : true,
      "workers" : 3
    },
    "job_status" : "Success",
    "creation_time" : "2024-01-20T10:00:00Z",
    "update_time" : "2024-01-26T02:30:15Z"
  } ]
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ListGcTasksSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
```

 版权所有 © 华为云计算技术有限公司

```
                .build();
        ListGcTasksRequest request = new ListGcTasksRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListGcTasksResponse response = client.listGcTasks(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListGcTasksRequest()
        request.instance_id = "{instance_id}"
        response = client.list_gc_tasks(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ListGcTasksRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ListGcTasks(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	查询制品清理的任务列表成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

版权所有 © 华为云计算技术有限公司

错误码

请参见**错误码**。

4.17.2 查询制品清理的任务详情 - ShowGcTask

功能介绍

查询制品清理的任务详情

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getGcTask	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}

表 4-1063 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
gc_id	是	String	任务ID

请求参数

表 4-1064 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-1065 响应 Body 参数

参数	参数类型	描述
id	Integer	gc任务的ID。
job_name	String	gc任务的名称。
job_kind	String	任务类型，MANUAL：手动执行，SCHEDULE：定时执行。
job_parameters	JobParameters object	gc任务的作业参数。
job_status	String	gc任务的状态，Success：已完成，Stopped：已停止，Running：清理中，Pending：排队中，Error：失败。
creation_time	String	gc任务的创建时间。
update_time	String	gc任务的更新时间。

表 4-1066 JobParameters

参数	参数类型	描述
delete_untagged	Boolean	是否删除无tag的制品；默认为false，不删除无tag的制品。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
workers	Integer	并行执行制品清理任务的工作者数量，最小值为1，最大值为5。
dry_run	Boolean	是否模拟执行任务；默认值为false，非模拟运行。
purged_blobs	Integer	清理镜像层的个数。
purged_manifests	Integer	清理镜像manifest的个数。
freed_space	Integer	释放的镜像存储空间大小(单位为字节)。

状态码：400

表 4-1067 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1068 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1069 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1070 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1071 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}
```

响应示例

状态码：200

查询制品清理的任务详情成功

```
{
  "id" : 1001,
  "job_name" : "daily-gc-task",
  "job_kind" : "SCHEDULE",
  "job_parameters" : {
    "delete_untagged" : true,
```


版权所有 © 华为云计算技术有限公司

```
    "dry_run" : true,
    "workers" : 3
  },
  "job_status" : "Success",
  "creation_time" : "2024-01-20T10:00:00Z",
  "update_time" : "2024-01-26T02:30:15Z"
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowGcTaskSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowGcTaskRequest request = new ShowGcTaskRequest();
        request.withInstanceId("{instance_id}");
        request.withGcId({gc_id});
        try {
            ShowGcTaskResponse response = client.showGcTask(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());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowGcTaskRequest()
        request.instance_id = "{instance_id}"
        request.gc_id = {gc_id}
        response = client.show_gc_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
        return
    }

    hcClient, err := swr.SwrClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowGcTaskRequest{}
    request.InstanceId = "{instance_id}"
    request.GcId = int32({gc_id})
    response, err := client.ShowGcTask(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	查询制品清理的任务详情成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.17.3 停止制品清理任务 - StopGcTask

功能介绍

停止制品清理任务

 版权所有 © 华为云计算技术有限公司

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:stopGcTask	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}

表 4-1072 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID
gc_id	是	String	任务ID

版权所有 © 华为云计算技术有限公司

请求参数

表 4-1073 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

停止制品清理任务成功

状态码：400

表 4-1074 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1075 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-1076 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1077 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1078 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

版权所有 © 华为云计算技术有限公司

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class StopGcTaskSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        StopGcTaskRequest request = new StopGcTaskRequest();
        request.withInstanceId("{instance_id}");
        request.withGcId({gc_id});
        try {
            StopGcTaskResponse response = client.stopGcTask(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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.
```

 版权所有 © 华为云计算技术有限公司

```
    # In this example, AK and SK are 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopGcTaskRequest()
        request.instance_id = "{instance_id}"
        request.gc_id = {gc_id}
        response = client.stop_gc_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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }
```

 版权所有 © 华为云计算技术有限公司

```
    client := swr.NewSwrClient(hcClient)

    request := &model.StopGcTaskRequest{}
    request.InstanceId = "{instance_id}"
    request.GcId = int32({gc_id})
    response, err := client.StopGcTask(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	停止制品清理任务成功
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.17.4 获取制品清理的计划信息 - ShowGcSchedule

功能介绍

获取制品清理的计划信息

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

 版权所有 © 华为云计算技术有限公司

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:getGcSchedule	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

GET /v2/{project_id}/instances/{instance_id}/system/gc/schedule

表 4-1079 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-1080 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 4-1081 响应 Body 参数

参数	参数类型	描述
schedule	ScheduleDetails object	gc任务的调度信息。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
parameters	GcParameters object	gc任务的作业参数。

表 4-1082 ScheduleDetails

参数	参数类型	描述
type	String	计划类型。有效值为'None'（无）和'Custom'（自定义）。 'Custom' 表示按照指定的 cron 计划触发，而 'None' 则表示取消定时计划。
cron	String	cron表达式，一种基于时间的任务调度器，type设置为Custom时，需要配置此值。
next_scheduled_time	String	下次执行任务的时间。

表 4-1083 GcParameters

参数	参数类型	描述
delete_untagged	Boolean	是否删除无tag的制品；默认为false，不删除无tag的制品。
workers	Integer	并行执行制品清理任务的工作者数量，最小值为1，最大值为5。
dry_run	Boolean	是否模拟执行任务；默认值为false，非模拟运行。

状态码：400

表 4-1084 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

版权所有 © 华为云计算技术有限公司

状态码：401

表 4-1085 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1086 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1087 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1088 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc/schedule
```

响应示例

状态码：200

获取制品清理的计划信息成功。

```
{
  "parameters" : {
    "delete_untagged" : true,
    "workers" : 3
  },
  "schedule" : {
    "type" : "Custom",
    "cron" : "0 2 * * *",
    "next_scheduled_time" : "2024-01-27T02:00:00Z"
  }
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ShowGcScheduleSolution {

    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)
```

 版权所有 © 华为云计算技术有限公司

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

    SwrClient client = SwrClient.newBuilder()
            .withCredential(auth)
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
    ShowGcScheduleRequest request = new ShowGcScheduleRequest();
    request.withInstanceId("{instance_id}");
    try {
        ShowGcScheduleResponse response = client.showGcSchedule(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowGcScheduleRequest()
        request.instance_id = "{instance_id}"
        response = client.show_gc_schedule(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 (
```

 版权所有 © 华为云计算技术有限公司

```
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ShowGcScheduleRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ShowGcSchedule(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	获取制品清理的计划信息成功。
400	错误的请求
401	鉴权失败

 版权所有 © 华为云计算技术有限公司

状态码	描述
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.17.5 执行制品清理计划 - ExecuteGcSchedule

功能介绍

执行制品清理计划

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:executeGcSchedule	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

POST /v2/{project_id}/instances/{instance_id}/system/gc/schedule

版权所有 © 华为云计算技术有限公司

表 4-1089 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-1090 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-1091 请求 Body 参数

参数	是否必选	参数类型	描述
schedule	是	**ExecuteScheduleObj** object	gc任务的调度信息。
parameters	是	**GcParameters** object	gc任务的作业参数。

表 4-1092 ExecuteScheduleObj

参数	是否必选	参数类型	描述
type	是	String	计划类型。有效值为 'Manual'（手动）。 'Manual' 表示立即触发。

表 4-1093 GcParameters

参数	是否必选	参数类型	描述
delete_untagged	否	Boolean	是否删除无tag的制品；默认为false，不删除无tag的制品。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
workers	是	Integer	并行执行制品清理任务的工作者数量，最小值为1，最大值为5。
dry_run	否	Boolean	是否模拟执行任务；默认值为false，非模拟运行。

响应参数

状态码：202

表 4-1094 响应 Body 参数

参数	参数类型	描述
id	Integer	制品清理的任务ID

状态码：400

表 4-1095 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

表 4-1096 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

版权所有 © 华为云计算技术有限公司

表 4-1097 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1098 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1099 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc/schedule

{
  "parameters" : {
    "delete_untagged" : false,
    "workers" : 1,
    "dry_run" : false
  },
  "schedule" : {
    "type" : "Manual"
```


版权所有 © 华为云计算技术有限公司

```
  }
}
```

响应示例

状态码：202

执行制品清理计划成功。

```
{
  "id" : 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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class ExecuteGcScheduleSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
                .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
                .build();
        ExecuteGcScheduleRequest request = new ExecuteGcScheduleRequest();
        request.withInstanceId("{instance_id}");
        ExecuteScheduleRequestBody body = new ExecuteScheduleRequestBody();
        GcParameters parametersbody = new GcParameters();
        parametersbody.withDeleteUntagged(false)
            .withWorkers(1)
            .withDryRun(false);
        ExecuteScheduleObj schedulebody = new ExecuteScheduleObj();
        schedulebody.withType(ExecuteScheduleObj.TypeEnum.fromValue("Manual"));
        body.withParameters(parametersbody);
        body.withSchedule(schedulebody);
        request.withBody(body);
        try {
            ExecuteGcScheduleResponse response = client.executeGcSchedule(request);
            System.out.println(response.toString());
```

 版权所有 © 华为云计算技术有限公司

```
        } 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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ExecuteGcScheduleRequest()
        request.instance_id = "{instance_id}"
        parametersbody = GcParameters(
            delete_untagged=False,
            workers=1,
            dry_run=False
        )
        schedulebody = ExecuteScheduleObj(
            type="Manual"
        )
        request.body = ExecuteScheduleRequestBody(
            parameters=parametersbody,
            schedule=schedulebody
        )
        response = client.execute_gc_schedule(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 (
```

 版权所有 © 华为云计算技术有限公司

```
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.ExecuteGcScheduleRequest{}
    request.InstanceId = "{instance_id}"
    deleteUntaggedParameters:= false
    dryRunParameters:= false
    parametersbody := &model.GcParameters{
        DeleteUntagged: &deleteUntaggedParameters,
        Workers: int32(1),
        DryRun: &dryRunParameters,
    }
    schedulebody := &model.ExecuteScheduleObj{
        Type: model.GetExecuteScheduleObjTypeEnum().MANUAL,
    }
    request.Body = &model.ExecuteScheduleRequestBody{
        Parameters: parametersbody,
        Schedule: schedulebody,
    }
    response, err := client.ExecuteGcSchedule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

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
202	执行制品清理计划成功。
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

4.17.6 配置制品清理计划 - UpdateGcSchedule

功能介绍

配置制品清理计划

接口约束

此接口当前是租户白名单开放，请提交工单申请开通访问。

调用方法

请参见**如何调用API**。

授权信息

账号具备所有API的调用权限，如果使用账号下的IAM用户调用当前API，该IAM用户需具备调用API所需的权限。

- 如果使用角色与策略授权，具体权限要求请参见**权限和授权项**。
- 如果使用身份策略授权，需具备如下身份策略权限。

版权所有 © 华为云计算技术有限公司

授权项	访问级别	资源类型（*为必须）	条件键	别名	依赖的授权项
swr:instance:updateGcSchedule	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-	-

URI

PUT /v2/{project_id}/instances/{instance_id}/system/gc/schedule

表 4-1100 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID
instance_id	是	String	企业仓库实例ID

请求参数

表 4-1101 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-1102 请求 Body 参数

参数	是否必选	参数类型	描述
schedule	是	ScheduleObj object	gc任务的调度信息。
parameters	否	GcParameters object	gc任务的作业参数，当schedule的type为Custom时，需要传入作业参数。

版权所有 © 华为云计算技术有限公司

表 4-1103 ScheduleObj

参数	是否必选	参数类型	描述
type	是	String	计划类型。有效值为'None'（无）和 'Custom'（自定义）。 'Custom' 表示按照指定的 cron 计划触发，而 'None' 则表示取消定时计划。
cron	否	String	cron表达式，一种基于时间的任务调度器，type设置为Custom时，需要配置此值。

表 4-1104 GcParameters

参数	是否必选	参数类型	描述
delete_untagged	否	Boolean	是否删除无tag的制品；默认为false，不删除无tag的制品。
workers	是	Integer	并行执行制品清理任务的工作者数量，最小值为1，最大值为5。
dry_run	否	Boolean	是否模拟执行任务；默认值为false，非模拟运行。

响应参数

状态码：200

配置制品清理计划成功。

状态码：400

表 4-1105 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：401

版权所有 © 华为云计算技术有限公司

表 4-1106 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：403

表 4-1107 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：404

表 4-1108 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

状态码：500

表 4-1109 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
encoded_authorization_message	String	加密后的详细拒绝原因，用户可以自行调用STS服务的decode-authorization-message接口进行解密。

请求示例

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/system/gc/schedule

{
  "parameters" : {
    "delete_untagged" : false,
    "workers" : 1
  },
  "schedule" : {
    "type" : "Custom",
    "cron" : "0 0 0 * * *"
  }
}
```

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.swr.v2.region.SwrRegion;
import com.huaweicloud.sdk.swr.v2.*;
import com.huaweicloud.sdk.swr.v2.model.*;


public class UpdateGcScheduleSolution {

    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);

        SwrClient client = SwrClient.newBuilder()
                .withCredential(auth)
```

 版权所有 © 华为云计算技术有限公司

```
            .withRegion(SwrRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateGcScheduleRequest request = new UpdateGcScheduleRequest();
        request.withInstanceId("{instance_id}");
        UpdateScheduleRequestBody body = new UpdateScheduleRequestBody();
        GcParameters parametersbody = new GcParameters();
        parametersbody.withDeleteUntagged(false)
            .withWorkers(1);
        ScheduleObj schedulebody = new ScheduleObj();
        schedulebody.withType(ScheduleObj.TypeEnum.fromValue("Custom"))
            .withCron("0 0 0 * * *");
        body.withParameters(parametersbody);
        body.withSchedule(schedulebody);
        request.withBody(body);
        try {
            UpdateGcScheduleResponse response = client.updateGcSchedule(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 huaweicloudsdkswr.v2.region.swr_region import SwrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkswr.v2 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 = SwrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SwrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateGcScheduleRequest()
        request.instance_id = "{instance_id}"
        parametersbody = GcParameters(
            delete_untagged=False,
            workers=1
        )
        schedulebody = ScheduleObj(
            type="Custom",
            cron="0 0 0 * * *"
        )
```

 版权所有 © 华为云计算技术有限公司

```
        request.body = UpdateScheduleRequestBody(
            parameters=parametersbody,
            schedule=schedulebody
        )
        response = client.update_gc_schedule(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"
    swr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/swr/v2/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, err := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        SafeBuild()

    if err != nil {
        fmt.Println(err)
        return
    }

    hcClient, err := swr.SwrClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         SafeBuild()


    if err != nil {
        fmt.Println(err)
        return
    }

    client := swr.NewSwrClient(hcClient)

    request := &model.UpdateGcScheduleRequest{}
    request.InstanceId = "{instance_id}"
    deleteUntaggedParameters:= false
    parametersbody := &model.GcParameters{
        DeleteUntagged: &deleteUntaggedParameters,
        Workers: int32(1),
    }
    cronSchedule:= "0 0 0 * * *"
    schedulebody := &model.ScheduleObj{
        Type: model.GetScheduleObjTypeEnum().CUSTOM,
        Cron: &cronSchedule,
```

 版权所有 © 华为云计算技术有限公司

```
    }
    request.Body = &model.UpdateScheduleRequestBody{
        Parameters: parametersbody,
        Schedule: schedulebody,
    }
    response, err := client.UpdateGcSchedule(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	配置制品清理计划成功。
400	错误的请求
401	鉴权失败
403	禁止访问
404	未找到资源
500	内部错误

错误码

请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

5 权限和授权项

5.1 权限和授权项说明

如果您需要对您所拥有的SWR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），如果华为账号所具备的权限功能已经能满足您的要求，您可以跳过本章节，不影响您使用SWR服务的其它功能。

通过IAM，您可以通过授权控制主体（IAM用户、用户组、IAM委托或信任委托）对华为云资源的访问范围。目前IAM支持两类授权，一类是角色与策略授权，另一类为身份策略授权。

两者有如下的区别和关系：

表 5-1 两类授权的区别

名称	核心关系	涉及的权限	授权方式	适用场景
角色与策略授权	用户-权限-授权范围	• 系统角色 • 系统策略 • 自定义策略	为主体授予角色或策略	核心关系为“用户-权限-授权范围”，每个用户根据所需权限和所需授权范围进行授权，无法直接给用户授权，需要维护更多的用户组，且支持的条件键较少，难以满足细粒度精确权限控制需求，更适用于对细粒度权限管控要求较低的中小企业用户。

版权所有 © 华为云计算技术有限公司

名称	核心关系	涉及的权限	授权方式	适用场景
身份策略授权	用户-策略	• 系统身份策略 • 自定义身份策略	• 为主体授予身份策略 • 身份策略附加至主体	核心关系为“用户-策略”，管理员可根据业务需求定制不同的访问控制策略，能够做到更细粒度更灵活的权限控制，新增资源时，对比角色与策略授权，基于身份策略的授权模型可以更快速地直接给用户授权，灵活性更强，更方便，但相对应的，整体权限管控模型构建更加复杂，对相关人员专业能力要求更高，因此更适用于中大型企业。

例如：如果需要对IAM用户授予可以使用华北-北京四区域的SWR和华南-广州区域的SWR的权限，基于角色与策略授权的场景中，管理员需要创建两个自定义策略，并且为IAM用户同时授予这两个自定义策略才可以实现权限控制。在基于身份策略授权的场景中，管理员仅需要创建一个自定义身份策略，在策略中通过条件键“g:RequestedRegion”的配置即可达到策略对于授权区域的控制。将身份策略附加主体或为主体授予该身份策略即可获得相应权限，权限配置方式更细粒度更灵活。

两种授权场景下的策略/身份策略、授权项等并不互通，推荐使用身份策略进行授权。

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。

5.2 角色与策略授权参考（旧版 IAM）

本章节介绍SWR策略授权场景下支持的策略授权项。

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，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。
-

版权所有 © 华为云计算技术有限公司

5.3 身份策略授权参考（新版 IAM）

云服务在IAM预置了常用的权限，称为系统身份策略。如果IAM系统身份策略无法满足授权要求，管理员可以根据各服务支持的授权项，创建IAM自定义身份策略来进行精细的访问控制，IAM自定义身份策略是对系统身份策略的扩展和补充。

除IAM服务外，**Organizations**服务中的**服务控制策略**（Service Control Policy，以下简称SCP）也可以使用这些授权项元素设置访问控制策略。

SCP不直接进行授权，只划定权限边界。将SCP绑定到组织单元或者成员账号时，并没有直接对组织单元或成员账号授予操作权限，而是规定了成员账号或组织单元包含的成员账号的授权范围。IAM身份策略授予权限的有效性受SCP限制，只有在SCP允许范围内的权限才能生效。

IAM服务与Organizations服务在使用这些元素进行访问控制时，存在着一些区别，详情请参见：**IAM服务与Organizations服务权限访问控制的区别**。

本章节介绍IAM服务身份策略授权场景中自定义身份策略和组织服务中SCP使用的元素，这些元素包含了操作（Action）、资源（Resource）和条件（Condition）。

- 如何使用这些元素编辑IAM自定义身份策略，请参考**创建自定义身份策略**。
- 如何使用这些元素编辑SCP自定义策略，请参考**创建SCP**。

操作（Action）

操作（Action）即为身份策略中支持的授权项。

- “访问级别”列描述如何对操作进行分类（List、Read和Write等）。此分类可帮助您了解在身份策略中相应操作对应的访问级别。
- “资源类型”列指每个操作是否支持资源级权限。
  - 资源类型支持通配符号*表示所有。如果此列没有值（-），则必须在身份策略语句的Resource元素中指定所有资源类型（“*”）。
  - 如果该列包含资源类型，则必须在具有该操作的语句中指定该资源的URN。
  - 资源类型列中必需资源在表中用星号（*）标识，表示使用此操作必须指定该资源类型。

  关于SWR定义的资源类型的详细信息请参见**资源类型（Resource）**。
- “条件键”列包括了可以在身份策略语句的Condition元素中支持指定的键值。
  - 如果该授权项资源类型列存在值，则表示条件键仅对列举的资源类型生效。
  - 如果该授权项资源类型列没有值（-），则表示条件键对整个授权项生效。
  - 如果此列条件键没有值（-），表示此操作不支持指定条件键。

  关于SWR定义的条件键的详细信息请参见**条件（Condition）**。
- “别名”列包括了可以在身份策略中配置的策略授权项。通过这些授权项，可以控制支持策略授权的API访问。详细信息请参见**身份策略兼容性说明**。

您可以在身份策略语句的Action元素中指定以下SWR的相关操作。

版权所有 © 华为云计算技术有限公司

表 5-2 SWR 支持的授权项

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:createImmutableRule	授予创建企业仓库镜像不可变规则的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteImmutableRule	授予删除企业仓库镜像不可变规则的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listImmutableRules	授予企业仓库获取镜像不可变规则列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateImmutableRule	授予企业仓库修改镜像不可变规则的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listArtifacts	授予查询制品列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getArtifact	授予查询制品详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:deleteArtifact	授予删除制品的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listAccessories	授予查询制品附件列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getArtifactAddition	授予查询制品附加信息的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getConfigurations	授予查询企业版实例配置的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateConfigurations	授予更新企业版实例配置的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listResourceInstances	授予查询资源实例列表的权限。	List	instance *	-	-
			-	• g:TagKeys • g:RequestTag/<tag-key>	
swr:instance:getResourceInstancesCount	授予查询资源实例数量的权限。	Read	instance *	-	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
			-	• g:TagKeys • g:RequestTag/<tag-key>	
swr:instance:createResourceTags	授予批量创建资源标签的权限。	Tagging	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
			-	• g:RequestTag/<tag-key> • g:TagKeys	
swr:instance:deleteResourceTags	授予批量删除资源标签的权限。	Tagging	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
			-	• g:RequestTag/<tag-key> • g:TagKeys	
swr:instance:getProjectTags	授予查询项目标签的权限。	Read	-	-	-
swr:instance:getResourceTags	授予查询资源标签的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:create	授予创建企业版实例的权限。	Write	instance *	-	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
			-	• g:EnterpriseProjectId • g:RequestTag/<tag-key> • g:TagKeys • swr:VpcId • swr:SubnetId • swr:EnableObsEncrypt	
swr:instance:list	授予查询企业版实例列表信息的权限。	List	instance *	-	-
swr:instance:get	授予查询企业版实例信息的权限。	Read	instance *	• g:EnterpriseProjectId • g:ResourceTag/<tag-key>	-
swr:instance:delete	授予删除企业版实例的权限。	Write	instance *	• g:EnterpriseProjectId • g:ResourceTag/<tag-key>	-
swr:instance:getAuditLogs	授予查询企业版实例审计日志的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:getStatistics	授予查询企业版实例统计信息的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listJobs	授予查询任务列表的权限。	List	instance *	-	-
swr:instance:getJobs	授予查询任务详情的权限。	Read	instance *	-	-
swr:instance:deleteJob	授予删除任务的权限。	Write	instance *	-	-
swr:repository:createNamespace	授予创建命名空间(组织)的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
			-	swr:EnablePublicNameSpace	
swr:repository:listNamespaces	授予查询命名空间(组织)列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getNamespace	授予查询命名空间(组织)详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateNamespace	授予修改命名空间(组织)的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
			-	swr:EnablePublicNameSpace	
swr:repository:deleteNamespace	授予删除命名空间(组织)的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listRepositories	授予获取制品仓库列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getRepository	授予获取制品仓库详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateRepository	授予修改制品仓库配置的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteRepository	授予删除制品仓库的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listAllInstanceRepositories	授予获取所有仓库实例的制品仓库列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:listTags	授予查询制品版本列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getTag	授予查询制品版本详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteTag	授予删除制品版本的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getTagAddition	授予查询制品版本附加信息的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:createRetentionPolicy	授予创建版本清理策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listRetentionPolicies	授予查询版本清理策略列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:getRetentionPolicy	授予查询版本清理策略详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateRetentionPolicy	授予修改版本清理策略配置的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteRetentionPolicy	授予删除版本清理策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:executeRetentionPolicy	授予应用版本清理策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listRetentionPolicyExecutions	授予获取版本清理执行记录列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listRetentionPolicyExecTasks	授予获取版本清理任务列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:listRetentionPolicyExecSubTasks	授予获取版本清理子任务列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:createWebhook	授予创建触发器的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listWebhooks	授予查询触发器列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getWebhook	授予查询触发器详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateWebhook	授予修改触发器配置的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteWebhook	授予删除触发器的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:listWebhookJobs	授予获取触发器执行记录列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:createRegistry	授予创建目标仓库的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listRegistries	授予获取目标仓库列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getRegistry	授予获取目标仓库详情的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateRegistry	授予修改目标仓库配置的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:deleteRegistry	授予删除目标仓库的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:createReplicationPolicy	授予创建复制策略的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId • swr:TargetRegion • swr:SourceRegion • swr:TargetUrls • swr:SourceUrls	-
swr:instance:listReplicationPolicies	授予查询复制策略列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getReplicationPolicy	授予查询复制策略详情的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateReplicationPolicy	授予修改复制策略的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId • swr:TargetRegion • swr:SourceRegion • swr:TargetUrls • swr:SourceUrls	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:deleteReplicationPolicy	授予删除复制策略的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:executeReplicationPolicy	授予应用复制策略的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:stopReplicationPolicyExecution	授予停止复制任务的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listReplicationPolicyExecutions	授予查询复制记录列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listReplicationPolicyExecTasks	授予查询复制任务列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listReplicationPolicyExecSubTasks	授予查询复制子任务列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:createSignPolicy	授予创建签名策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listSignPolicies	授予查询签名策略列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getSignPolicy	授予查询签名策略详情的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:updateSignPolicy	授予修改签名策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:deleteSignPolicy	授予删除签名策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:executeSignPolicy	授予执行签名策略的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:listSignPolicyExecutions	授予查询签名执行记录列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listSignPolicyExecTasks	授予查询签名任务列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listSignPolicyExecSubTasks	授予查询签名策略执行记录子任务列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateEndpointPolicy	授予更新公网访问白名单配置的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateEndpointPolicyStatus	授予更新公网访问白名单配置状态的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getEndpointPolicy	授予查询公网访问白名单配置的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:createInternalEndpoint	授予创建内网访问的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
			-	• swr:VpcId • swr:SubnetId	
swr:instance:getInternalEndpoint	授予获取内网访问的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:deleteInternalEndpoint	授予删除内网访问的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listInternalEndpoints	授予查询内网访问列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:uploadArtifact	授予上传制品的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:downloadArtifact	授予下载制品的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId • swr:RepositoryIsPublic	-
swr:instance:createTempCredential	授予创建临时访问凭证的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:createLTCredential	授予创建长期访问凭证的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateLTCredential	授予启用/停用长期访问凭证的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listLTCredentials	授予查询长期访问凭证列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:deleteLTCredential	授予删除长期访问凭证的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:addDomainName	授予增加域名的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:deleteDomainName	授予删除域名的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:updateDomainName	授予更新域名的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listDomainNames	授予查询域名列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:listCatalog	授予查询Catalog列表的权限。	List	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listAllArtifacts	授予查询仓库实例的所有制品版本列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:repository:startManualScanning	授予手动启动制品扫描的权限。	Write	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:repository:getArtifactVulnerabilities	授予查询制品漏洞信息的权限。	Read	repository *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:listGcTasks	授予查询制品清理任务列表的权限。	List	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getGcTask	授予查询制品清理任务详情的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:stopGcTask	授予停止制品清理任务的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:getGcSchedule	授予查询制品清理计划的权限。	Read	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

版权所有 © 华为云计算技术有限公司

授权项	描述	访问级别	资源类型（*为必须）	条件键	别名
swr:instance:updateGcSchedule	授予配置制品清理计划的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-
swr:instance:executeGcSchedule	授予执行制品清理计划的权限。	Write	instance *	• g:ResourceTag/<tag-key> • g:EnterpriseProjectId	-

SWR的API通常对应着一个或多个授权项。表5-3展示了API与授权项的关系，以及该API需要依赖的授权项。

表 5-3 API 与授权项的关系

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts	swr:repository:listArtifacts	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}	swr:repository:getArtifact	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}	swr:repository:deleteArtifact	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/accessories	swr:repository:listAccessories	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/additions/{addition}	swr:repository:getArtifactAddition	-
GET /v2/{project_id}/instances/{instance_id}/configurations	swr:instance:getConfigurations	-
PUT /v2/{project_id}/instances/{instance_id}/configurations	swr:instance:updateConfigurations	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
POST /v2/{project_id}/{resource_type}/resource-instances/filter	swr:instance:listResourceInstances	-
POST /v2/{project_id}/{resource_type}/resource-instances/count	swr:instance:getResourceInstancesCount	-
POST /v2/{project_id}/{resource_type}/{resource_id}/tags/create	swr:instance:createResourceTags	-
DELETE /v2/{project_id}/{resource_type}/{resource_id}/tags/delete	swr:instance:deleteResourceTags	-
GET /v2/{project_id}/instances/tags	swr:instance:getProjectTags	-
GET /v2/{project_id}/{resource_type}/{resource_id}/tags	swr:instance:getResourceTags	-
POST /v2/{project_id}/instances	swr:instance:create	-
GET /v2/{project_id}/instances	swr:instance:list	-
GET /v2/{project_id}/instances/{instance_id}	swr:instance:get	-
DELETE /v2/{project_id}/instances/{instance_id}	swr:instance:delete	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/audit-logs	swr:instance:getAuditLogs	-
GET /v2/{project_id}/instances/{instance_id}/statistics	swr:instance:getStatistics	-
GET /v2/{project_id}/jobs	swr:instance:listJobs	-
-	swr:instance:getJobs	-
DELETE /v2/{project_id}/jobs/{job_id}	swr:instance:deleteJob	-
POST /v2/{project_id}/instances/{instance_id}/namespaces	swr:repository:createNamespace	-
GET /v2/{project_id}/instances/{instance_id}/namespaces	swr:repository:listNamespaces	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}	swr:repository:getNamespace	-
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}	swr:repository:updateNamespace	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}	swr:repository:deleteNamespace	-
GET /v2/{project_id}/instances/{instance_id}/repositories	swr:repository:listRepositories	-
GET /v2/{project_id}/repositories	swr:repository:listAllInstanceRepositories	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}	swr:repository:getRepository	-
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}	swr:repository:updateRepository	-
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}	swr:repository:deleteRepository	-
GET /v2/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/tags	swr:repository:listTags	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/additions/{addition}	swr:repository:getTagAddition	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies	swr:repository:createRetentionPolicy	-
GET /v2/{project_id}/instances/{instance_id}/retention/policies	swr:repository:listRetentionPolicies	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}	swr:repository:getRetentionPolicy	-
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}	swr:repository:updateRetentionPolicy	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}	swr:repository:deleteRetentionPolicy	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions	swr:repository:executeRetentionPolicy	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions	swr:repository:listRetentionPolicyExecutions	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions/{execution_id}/tasks	swr:repository:listRetentionPolicyExecTasks	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/retention/policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks	swr:repository:listRetentionPolicyExecSubTasks	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies	swr:repository:createWebhook	-
GET /v2/{project_id}/instances/{instance_id}/webhook/policies	swr:repository:listWebhooks	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}	swr:repository:getWebhook	-
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}	swr:repository:updateWebhook	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}	swr:repository:deleteWebhook	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/webhook/policies/{policy_id}/jobs	swr:repository:listWebhookJobs	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules	swr:repository:createImmutableRule	-
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules/{immutable_rule_id}	swr:repository:deleteImmutableRule	-
GET /v2/{project_id}/instances/{instance_id}/immutabletagrules	swr:repository:listImmutableRules	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/immutabletagrules/{immutable_rule_id}	swr:repository:updateImmutableRule	-
POST /v2/{project_id}/instances/{instance_id}/registries	swr:instance:createRegistry	-
GET /v2/{project_id}/instances/{instance_id}/registries	swr:instance:listRegistries	-
GET /v2/{project_id}/instances/{instance_id}/registries/{registry_id}	swr:instance:getRegistry	-
PUT /v2/{project_id}/instances/{instance_id}/registries/{registry_id}	swr:instance:updateRegistry	-
DELETE /v2/{project_id}/instances/{instance_id}/registries/{registry_id}	swr:instance:deleteRegistry	-
POST /v2/{project_id}/instances/{instance_id}/replication/policies	swr:instance:createReplicationPolicy	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/replication/policies	swr:instance:listReplicationPolicies	-
GET /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}	swr:instance:getReplicationPolicy	-
PUT /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}	swr:instance:updateReplicationPolicy	-
DELETE /v2/{project_id}/instances/{instance_id}/replication/policies/{policy_id}	swr:instance:deleteReplicationPolicy	-
POST /v2/{project_id}/instances/{instance_id}/replication/executions	swr:instance:executeReplicationPolicy	-
PUT /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}	swr:instance:stopReplicationPolicyExecution	-
GET /v2/{project_id}/instances/{instance_id}/replication/executions	swr:instance:listReplicationPolicyExecutions	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks	swr:instance:listReplicationPolicyExecTasks	-
GET /v2/{project_id}/instances/{instance_id}/replication/executions/{execution_id}/tasks/{task_id}/subtasks	swr:instance:listReplicationPolicyExecSubTasks	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies	swr:repository:createSignPolicy	-
GET /v2/{project_id}/instances/{instance_id}/signature/policies	swr:repository:listSignPolicies	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}	swr:repository:getSignPolicy	-
PUT /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}	swr:repository:updateSignPolicy	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}	swr:repository:deleteSignPolicy	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions	swr:repository:executeSignPolicy	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions	swr:repository:listSignPolicyExecutions	-
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks	swr:repository:listSignPolicyExecTasks	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/signature/policies/{policy_id}/executions/{execution_id}/tasks/{task_id}/subtasks	swr:repository:listSignPolicyExecSubTasks	-
PUT /v2/{project_id}/instances/{instance_id}/endpoint-policy	swr:instance:updateEndpointPolicy	-
POST /v2/{project_id}/instances/{instance_id}/endpoint-policy	swr:instance:updateEndpointPolicyStatus	-
GET /v2/{project_id}/instances/{instance_id}/endpoint-policy	swr:instance:getEndpointPolicy	-
POST /v2/{project_id}/instances/{instance_id}/internal-endpoints	swr:instance:createInternalEndpoint	vpcep:endpoints:create
GET /v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}	swr:instance:getInternalEndpoint	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
DELETE /v2/{project_id}/instances/{instance_id}/internal-endpoints/{internal_endpoints_id}	swr:instance:deleteInternalEndpoint	vpcep:endpoints:delete
GET /v2/{project_id}/instances/{instance_id}/internal-endpoints	swr:instance:listInternalEndpoints	-
-	swr:repository:uploadArtifact	-
-	swr:repository:downloadArtifact	-
POST /v2/{project_id}/instances/{instance_id}/temp-credential	swr:instance:createTempCredential	-
POST /v2/{project_id}/instances/{instance_id}/long-term-credential	swr:instance:createLTCredential	-
PUT /v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}	swr:instance:updateLTCredential	-
GET /v2/{project_id}/instances/{instance_id}/long-term-credentials	swr:instance:listLTCredentials	-
DELETE /v2/{project_id}/instances/{instance_id}/long-term-credentials/{credential_id}	swr:instance:deleteLTCredential	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
POST /v2/{project_id}/instances/{instance_id}/domainname	swr:instance:addDomainName	scm:cert:export
DELETE /v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}	swr:instance:deleteDomainName	-
PUT /v2/{project_id}/instances/{instance_id}/domainname/{domainname_id}	swr:instance:updateDomainName	-
GET /v2/{project_id}/instances/{instance_id}/domainname	swr:instance:listDomainNames	-
-	swr:repository:listCatalog	-
GET /v2/{project_id}/instances/{instance_id}/artifacts	swr:instance:listAllArtifacts	-
POST /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/scan	swr:repository:startManualScanning	-

版权所有 © 华为云计算技术有限公司

API	对应的授权项	依赖的授权项
GET /v2/{project_id}/instances/{instance_id}/namespaces/{namespace_name}/repositories/{repository_name}/artifacts/{reference}/vulnerabilities	swr:repository:listArtifactVulnerabilities	-
GET /v2/{project_id}/instances/{instance_id}/system/gc	swr:instance:listGcTasks	-
GET /v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}	swr:instance:getGcTask	-
PUT /v2/{project_id}/instances/{instance_id}/system/gc/{gc_id}	swr:instance:stopGcTask	-
GET /v2/{project_id}/instances/{instance_id}/system/gc/schedule	swr:instance:getGcSchedule	-
PUT /v2/{project_id}/instances/{instance_id}/system/gc/schedule	swr:instance:updateGcSchedule	-
POST /v2/{project_id}/instances/{instance_id}/system/gc/schedule	swr:instance:executeGcSchedule	-

版权所有 © 华为云计算技术有限公司

资源类型（Resource）

资源类型（Resource）表示身份策略所作用的资源。如表5-4中的某些操作指定了可以在该操作指定的资源类型，则必须在具有该操作的身份策略语句中指定该资源的URN，身份策略仅作用于此资源；如未指定，Resource默认为“*”，则身份策略将应用到所有资源。您也可以在身份策略中设置条件，从而指定资源类型。

SWR定义了以下可以在自定义身份策略的Resource元素中使用的资源类型。

表 5-4 SWR 支持的资源类型

资源类型	URN
repository	swr:<region>:<account-id>:repository:<instance-name>/<repository-path>
instance	swr:<region>:<account-id>:instance:<instance-name>

条件（Condition）

条件键概述

条件（Condition）是身份策略生效的特定条件，包括条件键和运算符。

- 条件键表示身份策略语句的Condition元素中的键值。根据适用范围，分为全局级条件键和服务级条件键。
  - 全局级条件键（前缀为g:）适用于所有操作，在鉴权过程中，云服务不需要提供用户身份信息，系统将自动获取并鉴权。详情请参见：全局条件键。
  - 服务级条件键（前缀通常为服务缩写，如swr:）仅适用于对应服务的操作，详情请参见表5-5。
  - 单值/多值表示API调用时请求中与条件关联的值数。单值条件键在API调用时的请求中最多包含一个值，多值条件键在API调用时请求可以包含多个值。例如：g:SourceVpce是单值条件键，表示仅允许通过某个VPC终端节点发起请求访问某资源，一个请求最多包含一个VPC终端节点ID值。g:TagKeys是多值条件键，表示请求中携带的所有标签的key组成的列表，当用户在调用API请求时传入标签可以传入多个值。
- 运算符与条件键、条件值一起构成完整的条件判断语句，当请求信息满足该条件时，身份策略才能生效。支持的运算符请参见：运算符。

SWR支持的服务级条件键

SWR定义了以下可以在自定义身份策略的Condition元素中使用的条件键，您可以使用这些条件键进一步细化身份策略语句应用的条件。

表 5-5 SWR 支持的服务级条件键

服务级条件键	类型	单值/多值	说明
swr:VpcId	string	单值	按照用户VPC ID进行权限控制。

版权所有 © 华为云计算技术有限公司

服务级条件键	类型	单值/多值	说明
swr:SubnetId	string	单值	按照用户Subnet ID进行权限控制。
swr:EnablePublicNameSpace	boolean	单值	限制企业仓库是否允许创公开组织。
swr:EnableObsEncrypt	boolean	单值	限制企业版实例是否必须使用加密桶。
swr:TargetRegion	string	单值	根据目标区域进行权限控制。
swr:SourceRegion	string	单值	根据源区域进行权限控制。
swr:TargetUrls	string	多值	根据目标URL地址进行权限控制。
swr:SourceUrls	string	多值	根据源URL地址进行权限控制。
swr:RepositoryIsPublic	boolean	单值	根据镜像是否为公开镜像进行权限控制。

版权所有 © 华为云计算技术有限公司

6 附录

6.1 状态码

- 正常

返回值	说明
200 OK	GET和PUT操作正常返回。
201 Created	POST操作正常返回。
202 Accepted	请求已被接受。
204 No Content	DELETE操作正常返回。

- 异常

返回值	说明
400 Bad Request	服务器未能处理请求。
401 Unauthorized	被请求的页面需要用户名和密码。
403 Forbidden	对被请求页面的访问被禁止。
404 Not Found	服务器无法找到被请求的页面。
405 Method Not Allowed	请求中指定的方法不被允许。
406 Not Acceptable	服务器生成的响应无法被客户端所接受。
407 Proxy Authentication Required	用户必须首先使用代理服务器进行验证，这样请求才会被处理。
408 Request Timeout	请求超出了服务器的等待时间。
409 Conflict	由于冲突，请求无法被完成。
500 Internal Server Error	请求未完成。服务异常。

版权所有 © 华为云计算技术有限公司

返回值	说明
501 Not Implemented	请求未完成。服务器不支持所请求的功能。
502 Bad Gateway	请求未完成。服务器从上游服务器收到一个无效的响应。
503 Service Unavailable	请求未完成。系统暂时异常。
504 Gateway Timeout	网关超时。

6.2 错误码

调用接口出错后，将不会返回结果数据。调用方可根据每个接口对应的错误码来定位错误原因。当调用出错时，HTTP请求返回一个 4xx或5xx的HTTP状态码。返回的消息体中是具体的错误代码及错误信息。在调用方找不到错误原因时，可以联系华为云客服，并提供错误码，以便我们尽快帮您解决问题。

错误响应 Body 体格式说明

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。

```
{
    "errors": [{
        "errorCode": "SVCSTG.SWR.4030170",
        "errorMessage": "Insufficient permissions",
        "detail": "",
        "encoded_authorization_message":
"Egpjbi1ub3J0aC03GiAwO**********************************************YJns96GCCcWEws54Kds1Gj/wy3NzP/
NiSjP14=.BORs/KvbnoEeTkZCrRO1gw=="
    }],
    "error_code": "SVCSTG.SWR.4030170",
    "error_msg": "Insufficient permissions",
    "details": [{
        "error_code": "SVCSTG.SWR.4030170",
        "error_msg": ""
    }],
    "encoded_authorization_message":
"Egpjbi1ub3J0aC03GiAwO**********************************************YJns96GCCcWEws54Kds1Gj/wy3NzP/
NiSjP14=.BORs/KvbnoEeTkZCrRO1gw=="
}
```

其中，该body同时存在新旧两种格式的错误信息。

- 旧格式：errorCode表示错误码，errorMessage表示错误描述信息，detail表示更多的错误详情，encoded_authorization_message表示包含有关未经授权请求的信息。
- 新格式（推荐）：error_code表示错误码，error_msg表示错误描述信息，details表示更多的错误详情，encoded_authorization_message表示包含有关未经授权请求的信息，编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码。

版权所有 © 华为云计算技术有限公司

⚠ 注意

- encoded_authorization_message字段只在鉴权失败时才会携带。
- 推荐使用新格式的错误信息，旧格式的错误信息不建议使用。

错误码说明

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	描述	处理措施
401	SVCSTG.SWR.4010000	鉴权失败，可能因为Token过期。	请重新获取Token后重试。
500	SVCSTG.SWR.5000001	服务器内部错误：获取应用描述失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000002	服务器内部错误：获取镜像信息失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000003	服务器内部错误：获取镜像列表失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040004	软件包不存在。	请检查软件包是否存在。
500	SVCSTG.SWR.5000005	服务器内部错误：删除失败。	请参考返回的message，或联系技术支持。
409	SVCSTG.SWR.4090006	镜像已存在。	请检查镜像是否已存在。
500	SVCSTG.SWR.5000007	服务器内部错误：获取失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000008	服务器内部错误：获取列表失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040009	镜像列表未找到。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040010	镜像不存在。	请检查镜像是否存在。
500	SVCSTG.SWR.5000011	服务器内部错误：更新失败。	请参考返回的message，或联系技术支持。
406	SVCSTG.SWR.4060074	组织不能被删除，还含有镜像。	请先删除组织内的镜像再删除组织。
400	SVCSTG.SWR.4000013	无效的摘要。	请修正摘要。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4000014	API请求参数不合法。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
400	SVCSTG.SWR.4000015	无效的文件包名称。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
400	SVCSTG.SWR.4000016	无效的版本号。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
500	SVCSTG.SWR.5000017	服务器内部错误：应用签名功能未开启。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000018	服务器内部错误：读取公共密钥失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000019	服务器内部错误：不能找到本地已储存文件。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000020	服务器内部错误：会话锁获取失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000021	服务器内部错误：同步文件失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000022	服务器内部错误：同步元文件失败。	请参考返回的message，或联系技术支持。
406	SVCSTG.SWR.4060012	镜像仓库不能被删除，还含有镜像。	请先删除仓库里的镜像再删除仓库。
409	SVCSTG.SWR.4090023	系统资源繁忙。	系统资源繁忙，请稍后重试。
500	SVCSTG.SWR.5000024	服务器内部错误：无法生成sha值。	请参考返回的message，或联系技术支持。
413	SVCSTG.SWR.4130025	上传的文件太大。	请减小上传的文件。
500	SVCSTG.SWR.5000027	服务器内部错误：无法解开打包文件。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000028	错误的镜像格式。	请修正镜像格式。
400	SVCSTG.SWR.4000029	错误的镜像格式：镜像名不一致。	请修正镜像名称。
500	SVCSTG.SWR.5000030	服务器内部错误：无法定位文件指针。	请参考返回的message，或联系技术支持。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4000031	无效的ak、sk。	请检查ak、sk是否有效。
500	SVCSTG.SWR.5000032	服务器内部错误：json转换失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000033	服务器内部错误：应用文件保存失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000034	服务器内部错误：应用删除失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000035	服务器内部错误：应用保存到数据库失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000036	服务器内部错误：更新应用数据库失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000037	服务器内部错误：创建签名失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000038	服务器内部错误：删除签名失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000039	无效的sessionID。	请检查sessionID是否有效。
500	SVCSTG.SWR.5000040	无法更新状态。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000041	服务器内部错误：读取session失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000042	服务器内部错误：回收session失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040043	session不存在。	请检查session是否存在。
500	SVCSTG.SWR.5000044	服务器内部错误：生成session失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000045	服务器内部错误：绑定session失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000046	服务器内部错误：创建失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000047	此版本不支持。	此版本暂不支持。
404	SVCSTG.SWR.4040048	镜像版本不存在。	请检查镜像版本是否存在。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
500	SVCSTG.SWR.5000049	服务器内部错误：镜像删除失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000050	服务器内部错误：摘要生成失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000051	服务器内部错误：无法获取文件sha列表。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000052	服务器内部错误：无法减少镜像引用。	请参考返回的message，或联系技术支持。
406	SVCSTG.SWR.4060053	无法删除，请先删除软件包。	请先删除软件包。
500	SVCSTG.SWR.5000054	服务器内部错误：保存失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000055	缺少搜索输入。	请补充搜索输入。
400	SVCSTG.SWR.4000056	页面大小信息无效。	请修正页面大小信息。
500	SVCSTG.SWR.5000057	服务器内部错误：获取组织失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040058	组织不存在。	请检查组织是否存在。
500	SVCSTG.SWR.5000059	服务器内部错误：创建组织失败。	请参考返回的message，或联系技术支持。
409	SVCSTG.SWR.4090060	组织已存在。	请检查组织是否已存在。
500	SVCSTG.SWR.5000061	获取软件实例IP失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000062	获取软件本地IP失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000063	获取tag表失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000064	无效的镜像，镜像描述文件中含有空的“RepoTags”信息。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
400	SVCSTG.SWR.4000065	无效的镜像，镜像描述文件里的层文件缺失。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4000066	无效的镜像，镜像描述文件里的配置文件缺失。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
400	SVCSTG.SWR.4000067	您上传的镜像包格式不正确，镜像包中缺少描述文件。	请检查镜像包格式是否正确。
400	SVCSTG.SWR.4000068	无效的镜像，镜像描述文件无法解析。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
400	SVCSTG.SWR.4000069	无效的镜像，镜像格式错误或不支持。	请参考返回的message和SWR接口文档修改请求体，或联系技术支持。
507	SVCSTG.SWR.5070070	您的OBS服务未启用或有错误，请先检查您的OBS服务。	请先检查您的OBS服务是否正常可用。
500	SVCSTG.SWR.5000071	服务器内部错误：无法获取用户列表。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000073	服务器内部错误：恢复失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000075	服务器内部错误：无法更新组织。	请参考返回的message，或联系技术支持。
409	SVCSTG.SWR.4090076	组织的权限已存在。	请检查组织的权限是否已存在。
500	SVCSTG.SWR.5000077	服务器内部错误：获取组织权限失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040078	组织权限不存在。	请检查组织的权限是否存在。
500	SVCSTG.SWR.5000079	服务器内部错误：更新组织权限失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000080	服务器内部错误：创建组织权限失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000081	服务器内部错误：删除组织权限失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000082	服务器内部错误：解析json文件失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040083	没有找到请求url。	请检查请求url是否正确。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
500	SVCSTG.SWR.5000084	服务器内部错误：获取服务失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000085	服务器内部错误：创建tag失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000086	服务器内部错误：更新tag失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000087	服务器内部错误：获取镜像layer失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000088	服务器内部错误：uds中删除镜像层失败。	请参考返回的message，或联系技术支持。
409	SVCSTG.SWR.4090089	权限已经存在。	请检查权限是否已存在。
500	SVCSTG.SWR.5000090	服务器内部错误：获取权限失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040091	权限不存在。	检查权限是否存在。
500	SVCSTG.SWR.5000092	服务器内部错误：更新权限失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5000093	服务器内部错误：创建权限失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4000097	已超出可创建组织数量限额。	请提交工单申请增加配额。
400	SVCSTG.SWR.4000098	镜像数目超过上限。	镜像数目不能超过上限，请修改镜像数目后重新请求。
400	SVCSTG.SWR.4000099	镜像版本数目超过上限。	镜像版本数目不能超过上限，请修改镜像版本数目后重新请求。
500	SVCSTG.SWR.5000112	服务器内部错误：获取镜像资源失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040113	镜像资源不存在。	请检查镜像资源是否存在。
500	SVCSTG.SWR.5000114	服务器内部错误：获取镜像版本失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4040115	镜像版本不存在。	请检查镜像版本是否存在。
400	SVCSTG.SWR.4001000	Content-Type必须为multipart/form-data类型。	请检查Content-Type是否为multipart/form-data类型。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
413	SVCSTG.SWR.4131001	请求体太大。	请求体过大，请修改请求体重新请求。
500	SVCSTG.SWR.5001002	服务器内部错误：“External”地址未开启。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001003	获取表单文件失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001004	参数“type”应该为“app”。	请修改参数“type”。
400	SVCSTG.SWR.4001005	解析表单失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001006	元文件格式化无效。	请修改元文件格式化。
400	SVCSTG.SWR.4001008	组织或名称无效。	请修改组织或名称。
400	SVCSTG.SWR.4001009	参数“is_public”应该为“true”或“false”。	请修改参数“is_public”。
400	SVCSTG.SWR.4001010	参数“type”应该为“app”或“image”。	请修改参数“type”。
400	SVCSTG.SWR.4001011	参数“type”应该为“app”、“image”或为空。	请修改参数“type”。
413	SVCSTG.SWR.4131012	该描述内容长度超过限制。	请缩短描述内容。
400	SVCSTG.SWR.4001013	“ak”或“sk”不能为空。	请填写“ak”或“sk”。
400	SVCSTG.SWR.4001014	必选参数未填。	请填写全部的必填参数。
400	SVCSTG.SWR.4001015	表单文件缺失。	请检查表单文件是否缺失。
400	SVCSTG.SWR.4001016	必选参数未填，至少要包含“is_public”和“manifests”两个参数。	请检查“is_public”和“manifests”两个参数是否都填写了。
400	SVCSTG.SWR.4001018	请求体类型不正确。	请修改请求体类型。
400	SVCSTG.SWR.4001019	解析表单文件失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001020	服务器内部错误：读取表单文件失败。	请参考返回的message，或联系技术支持。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4001021	版本号不能为空。	请检查版本号是否为空。
400	SVCSTG.SWR.4001022	上传文件类型错误。	请修改上传文件类型。
400	SVCSTG.SWR.4001023	不支持的文件类型。	请修改文件类型。
400	SVCSTG.SWR.4001024	查询参数失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001025	查询参数不合法。	请参照接口api文档指导修改查询参数。
500	SVCSTG.SWR.5001026	manifest获取失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001027	“status”只能为“done”或“error”。	请按照返回的message修改“status”。
400	SVCSTG.SWR.4001028	组织名称不合法。	请修改组织名称。
400	SVCSTG.SWR.4001029	镜像名称不合法。	请修改镜像名称。
400	SVCSTG.SWR.4001030	“package”不合法。	请修改“package”。
403	SVCSTG.SWR.4031032	服务器运行时的资源，请求太多。	请参考返回的message，或联系技术支持。
403	SVCSTG.SWR.4031033	请求不可用。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001035	过滤无效。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001036	参数丢失。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001037	注册中心类型不支持。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001038	不可行的注册中心。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001039	搜索失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001041	“domain”参数无效。	请修改“domain”参数。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
403	SVCSTG.SWR.4031042	请求数目太多。	请参考返回的message，或联系技术支持。
403	SVCSTG.SWR.4031043	公证人jwt令牌请求无效。	请参考返回的message，或联系技术支持。
401	SVCSTG.SWR.4011044	镜像已被其他用户创建，您无权上传该镜像。	请检查您是否有对应的权限。
400	SVCSTG.SWR.4001045	缺少“SWRServer”参数。	请检查是否缺少“SWRServer”参数。
500	SVCSTG.SWR.5001046	服务器内部错误：检查组织失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001047	服务器内部错误：保存组织失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4001048	获取token失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001049	服务器内部错误：删除组织失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001050	服务器内部错误：获取镜像层失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4041051	镜像层不存在。	请检查镜像层是否存在。
500	SVCSTG.SWR.5001052	服务器内部错误：镜像层不合法。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001053	服务器内部错误：镜像层上传不合法。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001054	服务器内部错误：创建镜像路径失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001055	服务器内部错误：保存镜像层失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4041056	镜像tag不存在。	请检查镜像tag是否存在。
500	SVCSTG.SWR.5001056	服务器内部错误：获取tag列表失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4041057	tag列表不存在。	请检查tag列表是否存在。
500	SVCSTG.SWR.5001058	服务器内部错误：保存镜像失败。	请参考返回的message，或联系技术支持。

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
500	SVCSTG.SWR.5001059	服务器内部错误：获取镜像失败。	请参考返回的message，或联系技术支持。
404	SVCSTG.SWR.4041060	manifest不存在。	请检查manifest是否存在。
404	SVCSTG.SWR.4041061	组织不存在。	请检查组织是否存在。
500	SVCSTG.SWR.5001062	服务器内部错误：更新组织失败。	请参考返回的message，或联系技术支持。
500	SVCSTG.SWR.5001063	服务器内部错误：删除镜像层失败。	请参考返回的message，或联系技术支持。
401	SVCSTG.SWR.4011064	跨账号错误（例如：使用A账号的token去访问或操作B账号下的资源），没有权限。	请检查您是否有对应的权限。
400	SVCSTG.SWR.4001065	组织权限不合法。	请修改组织权限。
400	SVCSTG.SWR.4001066	权限不合法。	请修改权限。
400	SVCSTG.SWR.4001067	用户名不合法。	请修改用户名。
500	SVCSTG.SWR.5001076	获取服务token失败。	请参考返回的message，或联系技术支持。
401	SVCSTG.S2I.4019000	鉴权错误：没有发现任何用户token或用户AK/SK信息。	请检查您是否有对应的权限。
403	SVCSTG.SWR.4039001	您无法申请或操作不属于您的资源。	请检查您是否有对应的权限。
403	SVCSTG.SWR.4039002	您没有权限执行本次操作。	请检查您是否有对应的权限。
400	SVCSTG.SWR.4009010	无效的组织。	请修正组织。
400	SVCSTG.SWR.4009011	无效的镜像名称。	请修正镜像名称。
400	SVCSTG.SWR.4009012	无效的代码源类型。	请修正代码源类型。
400	SVCSTG.SWR.4009013	无效的ID。	请修正ID.

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4009014	无效的过滤条件。	请修正过滤条件。
400	SVCSTG.SWR.4009015	解析构建任务失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4009016	无效的文件路径。	请修正文件路径。
400	SVCSTG.SWR.4009017	无效的代码源所有者。	请修正代码源所有者。
400	SVCSTG.SWR.4009018	无效的代码源仓库。	请修正代码源仓库。
400	SVCSTG.SWR.4009019	无效的代码源仓库ID。	请修正代码源仓库ID。
400	SVCSTG.SWR.4009020	无效的代码源分支。	请修正代码源分支。
400	SVCSTG.SWR.4009021	无效的镜像标签。	请修正镜像标签。
400	SVCSTG.SWR.4009022	无效的代码源用户名。	请修正代码源用户名。
400	SVCSTG.SWR.4009023	无效的commit id。	请修正commit id。
400	SVCSTG.SWR.4006030	执行本次构建失败。	请参考返回的message，或联系技术支持。
400	SVCSTG.SWR.4006031	本次构建记录已存在。	请检查本次构建记录是否已存在。
404	SVCSTG.SWR.4046032	本次构建记录不存在。	请检查本次构建记录是否不存在。
500	SVCSTG.SWR.5006033	从数据库中获取本次构建记录失败。	请参考返回的message，或联系技术支持。
403	SVCSTG.SWR.4037017	您没有编辑/管理任何组织的权限。	请先创建组织或获取组织的编辑/管理权限。
403	SVCSTG.SWR.4037018	您没有权限编辑/管理镜像仓库中的任何镜像。	请先创建镜像或获取可编辑的/可管理的镜像。
500	SVCSTG.SWR.5007019	创建镜像失败，镜像个数已达到上限。	请参考返回的message，或联系技术支持。
403	SVCSTG.SWR.4037020	您暂无权限操作该镜像仓库资源。	请提升您对应的组织权限

版权所有 © 华为云计算技术有限公司

状态码	错误码	描述	处理措施
400	SVCSTG.SWR.4001091	参数permission错误。	请修正参数permission。

6.3 获取项目 ID

操作场景

在调用接口的时候，部分URL中需要填入项目ID，所以需要获取到项目ID。有如下两种获取方式：

- **调用API获取项目ID**
- **从控制台获取项目ID**

调用 API 获取项目 ID

项目ID可以通过调用**查询指定条件下的项目列表**API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects”，其中{Endpoint}为IAM的终端节点，可以从**地区和终端节点**获取。接口的认证鉴权请参见**认证鉴权**。

响应示例如下，其中projects下的“id”即为项目ID。

```
{
    "projects": [
        {
            "domain_id": "65382**********14e684b",
            "is_domain": false,
            "parent_id": "65382**********14e684b",
            "name": "project_name",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```

从控制台获取项目 ID

从控制台获取项目ID的步骤如下：

1. 访问**“我的凭证”页面**。
2. 在“API凭证”页面的项目列表中查看项目ID。

版权所有 © 华为云计算技术有限公司

图 6-1 查看项目 ID

6.4 获取用户名和用户 ID

操作场景

在调用接口的时候，部分URL中需要填入用户名和用户ID，所以需要先在管理控制台上获取。具体方法见**操作步骤**。

操作步骤

1. 登录访问**“我的凭证”页面**。
2. 在“API凭证”页面查看IAM用户名和IAM用户ID。

图 6-2 API 凭证

版权所有 © 华为云计算技术有限公司