主机安全服务

API 参考

文档版本 06

发布日期 2024-07-04

华为云计算技术有限公司

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 概述

欢迎使用主机安全服务（Host Security Service，HSS）。主机安全服务是提升主机整体安全性的服务，通过主机管理、风险预防、入侵检测、高级防御、安全运营、网页防篡改功能，全面识别并管理主机中的信息资产，实时监测主机中的风险并阻止非法入侵行为，帮助企业构建服务器安全体系，降低当前服务器面临的主要安全风险。

您可以使用本文档提供的API对主机安全服务进行相关操作。

在调用主机安全服务API之前，请确保已经充分了解主机安全服务相关概念，详细信息请参见**产品介绍**。

1.2 调用说明

主机安全服务提供了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**。

1.3 终端节点

终端节点（Endpoint）即调用API的**请求地址**，不同服务不同区域的终端节点不同。

主机安全服务的终端节点如**表1-1**所示，请您根据业务需要选择对应区域的终端节点。

表 1-1 主机安全服务的终端节点

区域名称	区域	终端节点（Endpoint）	协议类型
中国-香港	ap-southeast-1	hss.ap-southeast-1.myhuaweicloud.com	HTTPS
亚太-曼谷	ap-southeast-2	hss.ap-southeast-2.myhuaweicloud.com	HTTPS

区域名称	区域	终端节点（Endpoint）	协议类型
亚太-新加坡	ap-southeast-3	hss.ap-southeast-3.myhuaweicloud.com	HTTPS

1.4 约束与限制

单API流量每分钟限制访问次数为600次，其中单用户每分钟访问单API次数最大为5次，单IP地址每分钟访问单API次数最大为5次。

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它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）。

  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。

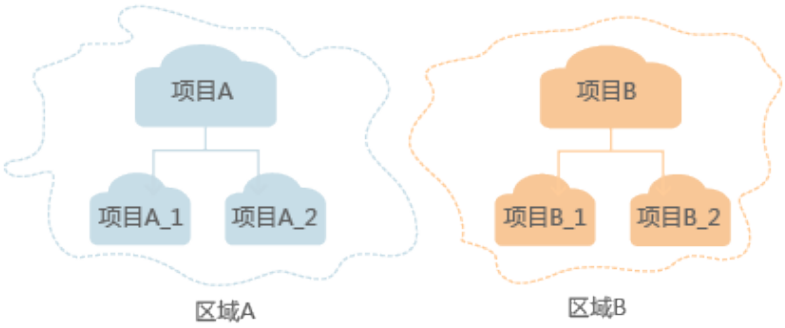
- 可用区（AZ，Availability Zone）

  一个AZ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。

图 1-1 项目隔离模型



- 企业项目

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

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

2 如何调用 API

2.1 构造请求

本节介绍如何构造REST API的请求，并以调用IAM服务的**获取用户Token**说明如何调用API，该API获取用户的Token，Token可以用于调用其他API时鉴权。

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。

- **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“中国-香港”区域的Endpoint为“iam.ap-southeast-1.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”。

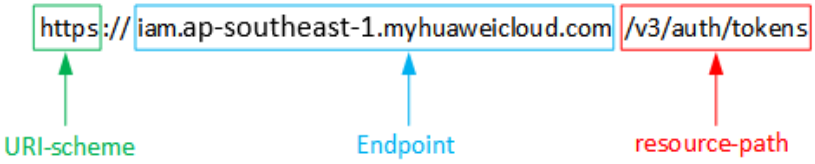
- **query-string**：

  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。

例如您需要获取IAM在“中国-香港”区域的Token，则需使用“中国-香港”区域的Endpoint（iam.ap-southeast-1.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.ap-southeast-1.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，同一个服务的Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

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

在**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.ap-southeast-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。

如下公共消息头需要添加到请求中。

- **Content-Type**：消息体的类型（格式），必选，默认取值为“application/json”，有其他取值时会在具体接口中专门说明。
- **X-Auth-Token**：用户Token，可选，当使用Token方式认证时，必须填充该字段。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。

  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**AK/SK认证**。

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.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

请求消息体

请求消息体通常以结构化格式发出，与请求消息头中Content-type对应，传递除请求消息头之外的内容。若请求消息体中参数支持中文，则中文字符必须为UTF-8编码。

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，您可以从**地区和终端节点**获取，对应地区和终端节点页面的“区域”字段的值。

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.ap-southeast-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。

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

```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "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.ap-southeast-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

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**。

须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

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。

图 2-2 获取用户 Token 响应消息头

响应消息体（可选）

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

对于**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": "xxxxxxxx",
.....
```

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": {
        "message": "The request you have made requires authentication.",
        "title": "Unauthorized"
    }
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

3 API 说明

3.1 资产管理

3.1.1 统计资产信息，账号、端口、进程等

功能介绍

资产统计信息，账号、端口、进程等

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/statistics

表 3-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-2 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**
host_id	否	String	Host ID 最小长度：**1** 最大长度：**128**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**1** 最大长度：**64**

请求参数

表 3-3 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-4 响应 Body 参数

参数	参数类型	描述
account_num	Long	主机账号数量 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
port_num	Long	开放端口数量 最小值：**0** 最大值：**2147483647**
process_num	Long	进程数量 最小值：**0** 最大值：**2147483647**
app_num	Long	软件数量 最小值：**0** 最大值：**2147483647**
auto_launch_num	Long	自启动进程数量 最小值：**0** 最大值：**2147483647**
web_framework_num	Long	web框架数量 最小值：**0** 最大值：**2147483647**
web_site_num	Long	Web站点数量 最小值：**0** 最大值：**2147483647**
jar_package_num	Long	Jar包数量 最小值：**0** 最大值：**2147483647**
kernel_module_num	Long	内核模块数量 最小值：**0** 最大值：**2147483647**
web_service_num	Long	web服务数量 最小值：**0** 最大值：**2147483647**
web_app_num	Long	web应用数量 最小值：**0** 最大值：**2147483647**
database_num	Long	数据库数量 最小值：**0** 最大值：**2147483647**

请求示例

查询主机的指纹信息，账号、端口、进程等的统计数量

```
GET https://{endpoint}/v5/{project_id}/asset/statistics?category=host
```

响应示例

状态码： 200

Asset statistic info

```
{
  "account_num" : 5,
  "port_num" : 5,
  "process_num" : 5,
  "app_num" : 5,
  "auto_launch_num" : 5,
  "web_framework_num" : 5,
  "web_site_num" : 5,
  "jar_package_num" : 5,
  "kernel_module_num" : 5,
  "database_num" : 1,
  "web_app_num" : 8,
  "web_service_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowAssetStatisticSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowAssetStatisticRequest request = new ShowAssetStatisticRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
```

```
        request.withHostId("<host_id>");
        request.withCategory("<category>");
        try {
            ShowAssetStatisticResponse response = client.showAssetStatistic(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowAssetStatisticRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_id = "<host_id>"
        request.category = "<category>"
        response = client.show_asset_statistic(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAssetStatisticRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ShowAssetStatistic(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	Asset statistic info

错误码

请参见**错误码**。

3.1.2 查询账号信息列表

功能介绍

查询账号信息列表，支持通过传入账号名称参数查询对应的服务器数

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/user/statistics

表 3-5 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-6 Query 参数

参数	是否必选	参数类型	描述
user_name	否	String	账号名称，参考windows文件命名规则，支持字母、数字、下划线、中文，特殊字符!@.-等，不包括中文标点符号 最小长度：**1** 最大长度：**128**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**

请求参数

表 3-7 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-8 响应 Body 参数

参数	参数类型	描述
total_num	Integer	账号总数 最小值：**0** 最大值：**10000**
data_list	Array of **UserStatisticInfoResponseInfo** objects	账户统计信息列表 数组长度：**0 - 10000**

表 3-9 UserStatisticInfoResponseInfo

参数	参数类型	描述
user_name	String	账号名称，参考windows文件命名规则，支持字母、数字、下划线，特殊字符!@.-等 最小长度：**1** 最大长度：**128**
num	Integer	当前账号的主机数量 最小值：**0** 最大值：**10000**

请求示例

默认查询前10条账号信息列表

```
GET https://{endpoint}/v5/{project_id}/asset/user/statistics
```

响应示例

状态码： 200

具备该账号的主机数量

```
{
  "total_num" : 1,
  "data_list" : [ {
    "user_name" : "bin",
    "num" : 5
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListUserStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListUserStatisticsRequest request = new ListUserStatisticsRequest();
        request.withUserName("<user_name>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        try {
            ListUserStatisticsResponse response = client.listUserStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListUserStatisticsRequest()
        request.user_name = "<user_name>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        response = client.list_user_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListUserStatisticsRequest{}
    userNameRequest:= "<user_name>"
    request.UserName = &userNameRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ListUserStatistics(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	具备该账号的主机数量

错误码

请参见**错误码**。

3.1.3 查询开放端口统计信息

功能介绍

查询开放端口列表，支持通过传入端口或协议类型查询服务器数

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/port/statistics

表 3-10 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-11 Query 参数

参数	是否必选	参数类型	描述
port	否	Integer	端口号，精确匹配 最小值：**1** 最大值：**65535**
port_string	否	String	端口字符串，用来进行模糊匹配 最小长度：**1** 最大长度：**256**
type	否	String	端口类型 最小长度：**1** 最大长度：**256**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**
sort_key	否	String	排序的key值，目前支持按照端口号port排序 最小长度：**1** 最大长度：**128**
sort_dir	否	String	升序还是降序，默认升序，asc 最小长度：**1** 最大长度：**32**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： • host：主机 • container：容器 最小长度：**0** 最大长度：**64**

请求参数

表 3-12 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-13 响应 Body 参数

参数	参数类型	描述
total_num	Integer	开放端口总数 最小值：**0** 最大值：**10000**
data_list	Array of **PortStatisticResponseInfo** objects	开放端口统计信息列表 数组长度：**0 - 10000**

表 3-14 PortStatisticResponseInfo

参数	参数类型	描述
port	Integer	端口号 最小值：**0** 最大值：**65535**
type	String	端口类型 最小长度：**1** 最大长度：**256**
num	Integer	端口数量 最小值：**0** 最大值：**10000**
status	String	危险类型:danger/unknown 最小长度：**1** 最大长度：**16**

请求示例

默认查询前10条端口为123，类别为主机的开放端口列表

```
GET https://{endpoint}/v5/{project_id}/asset/port/statistics?port=123&category=host
```

响应示例

状态码： 200

返回端口信息，端口号、类型、数量、危险状态

```
{
  "total_num" : 1,
  "data_list" : [ {
    "num" : 4,
    "port" : 123,
    "type" : "UDP",
    "status" : "danger"
  } ]
}
```

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.hss.v5.region.HssRegion;
```

```
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListPortStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPortStatisticsRequest request = new ListPortStatisticsRequest();
        request.withPort(<port>);
        request.withPortString("<port_string>");
        request.withType("<type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withSortKey("<sort_key>");
        request.withSortDir("<sort_dir>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        try {
            ListPortStatisticsResponse response = client.listPortStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListPortStatisticsRequest()
        request.port = <port>
        request.port_string = "<port_string>"
        request.type = "<type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.sort_key = "<sort_key>"
        request.sort_dir = "<sort_dir>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        response = client.list_port_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPortStatisticsRequest{}
    portRequest:= int32(<port>)
    request.Port = &portRequest
    portStringRequest:= "<port_string>"
    request.PortString = &portStringRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    sortKeyRequest:= "<sort_key>"
    request.SortKey = &sortKeyRequest
```

```
    sortDirRequest:= "<sort_dir>"
    request.SortDir = &sortDirRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ListPortStatistics(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	返回端口信息，端口号、类型、数量、危险状态

错误码

请参见**错误码**。

3.1.4 查询进程列表

功能介绍

查询进程列表，通过传入进程路径参数查询对应的服务器数

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/process/statistics

表 3-15 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-16 Query 参数

参数	是否必选	参数类型	描述
path	否	String	可执行进程路径 最小长度：**1** 最大长度：**256**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**

请求参数

表 3-17 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-18 响应 Body 参数

参数	参数类型	描述
total_num	Integer	进程统计信息总数, 最小值：**0** 最大值：**10000**
data_list	Array of **ProcessStatisticResponseInfo** objects	进程统计信息列表 数组长度：**0 - 10000**

表 3-19 ProcessStatisticResponseInfo

参数	参数类型	描述
path	String	进程的可执行文件路径 最小长度：**1** 最大长度：**256**
num	Integer	进程数量 最小值：**0** 最大值：**100000**

请求示例

默认查询前10条类别为主机的进程列表

```
GET https://{endpoint}/v5/{project_id}/asset/process/statistics?category=host
```

响应示例

状态码： 200

具备该进程的主机数量

```
{
  "total_num" : 1,
  "data_list" : [ {
    "num" : 13,
    "path" : "/usr/lib/systemd/systemd-journald"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListProcessStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListProcessStatisticsRequest request = new ListProcessStatisticsRequest();
        request.withPath("<path>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        try {
            ListProcessStatisticsResponse response = client.listProcessStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListProcessStatisticsRequest()
        request.path = "<path>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        response = client.list_process_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProcessStatisticsRequest{}
    pathRequest:= "<path>"
    request.Path = &pathRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ListProcessStatistics(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	具备该进程的主机数量

错误码

请参见**错误码**。

3.1.5 查询软件列表

功能介绍

查询软件列表，支持通过软件名称查询对应的服务器数

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/app/statistics

表 3-20 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-21 Query 参数

参数	是否必选	参数类型	描述
app_name	否	String	软件名称 最小长度：**1** 最大长度：**256**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**

请求参数

表 3-22 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-23 响应 Body 参数

参数	参数类型	描述
total_num	Integer	进程统计信息总数, 最小值：**0** 最大值：**10000**
data_list	Array of **AppStatisticResponseInfo** objects	进程统计信息列表 数组长度：**0 - 10000**

表 3-24 AppStatisticResponseInfo

参数	参数类型	描述
app_name	String	软件名称 最小长度：**1** 最大长度：**128**
num	Integer	进程数量 最小值：**0** 最大值：**100000**

请求示例

默认查询前10条类别为主机的软件列表

```
GET https://{endpoint}/v5/{project_id}/asset/app/statistics?category=host
```

响应示例

状态码： 200

具备该软件的主机数量

```
{
  "total_num" : 1,
  "data_list" : [ {
    "app_name" : "kernel",
    "num" : 13
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAppStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAppStatisticsRequest request = new ListAppStatisticsRequest();
        request.withAppName("<app_name>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        try {
            ListAppStatisticsResponse response = client.listAppStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAppStatisticsRequest()
        request.app_name = "<app_name>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        response = client.list_app_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAppStatisticsRequest{}
    appNameRequest:= "<app_name>"
    request.AppName = &appNameRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ListAppStatistics(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	具备该软件的主机数量

错误码

请参见**错误码**。

3.1.6 查询自启动项信息

功能介绍

查询自启动信息，支持通过传入自启动名称查询启动类型和服务器数

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/auto-launch/statistics

表 3-25 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-26 Query 参数

参数	是否必选	参数类型	描述
name	否	String	自启动项名称 最小长度：**1** 最大长度：**256**
type	否	String	自启动项类型 ● 0： 自启动服务 ● 1： 定时任务 ● 2： 预加载动态库 ● 3： Run注册表键 ● 4： 开机启动文件夹 最小长度：**1** 最大长度：**256**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**

请求参数

表 3-27 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-28 响应 Body 参数

参数	参数类型	描述
total_num	Integer	自启动项统计信息总数, 最小值：**0** 最大值：**10000**
data_list	Array of **AutoLaunchStatisticsResponseInfo** objects	自启动项统计信息列表 数组长度：**0 - 10000**

表 3-29 AutoLaunchStatisticsResponseInfo

参数	参数类型	描述
name	String	自启动项名称 最小长度：**1** 最大长度：**256**
type	String	自启动项类型 • 0： 自启动服务 • 1： 定时任务 • 2： 预加载动态库 • 3： Run注册表键 • 4： 开机启动文件夹 最小长度：**1** 最大长度：**11**
num	Integer	当前自启动项的主机数量 最小值：**0** 最大值：**10000**

请求示例

默认查询前10条自启动项列表

```
GET https://{endpoint}/v5/{project_id}/asset/auto-launch/statistics
```

响应示例

状态码： 200

具备该进程的主机数量

```
{
  "total_num" : 1,
  "data_list" : [ {
    "name" : "S12hostguard",
    "type" : "0",
    "num" : 5
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAutoLaunchStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAutoLaunchStatisticsRequest request = new ListAutoLaunchStatisticsRequest();
        request.withName("<name>");
        request.withType("<type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListAutoLaunchStatisticsResponse response = client.listAutoLaunchStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAutoLaunchStatisticsRequest()
        request.name = "<name>"
        request.type = "<type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_auto_launch_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

```
    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAutoLaunchStatisticsRequest{}
    nameRequest:= "<name>"
    request.Name = &nameRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListAutoLaunchStatistics(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	具备该进程的主机数量

错误码

请参见**错误码**。

3.1.7 查询账号的服务器列表

功能介绍

查询账号的服务器列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/users

表 3-30 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**128**

表 3-31 Query 参数

参数	是否必选	参数类型	描述
host_id	否	String	主机ID 最小长度：**0** 最大长度：**128**
user_name	否	String	账号名称 最小长度：**0** 最大长度：**32**
host_name	否	String	主机名称 最小长度：**0** 最大长度：**128**
private_ip	否	String	服务器私有IP 最小长度：**0** 最大长度：**128**
login_permission	否	Boolean	是否允许登录
root_permission	否	Boolean	是否有root权限
user_group	否	String	主机用户组 最小长度：**0** 最大长度：**128**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： • host：主机 • container：容器 最小长度：**0** 最大长度：**64**
part_match	否	Boolean	是否模糊匹配，默认false表示精确匹配

请求参数

表 3-32 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-33 响应 Body 参数

参数	参数类型	描述
total_num	Integer	账号总数 最小值：**0** 最大值：**10000**
data_list	Array of **UserResponseInfo** objects	账号信息列表 数组长度：**0 - 10000**

表 3-34 UserResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**1** 最大长度：**128**
host_id	String	主机ID 最小长度：**1** 最大长度：**128**
host_name	String	服务器名称 最小长度：**1** 最大长度：**128**
host_ip	String	服务器ip 最小长度：**1** 最大长度：**128**
user_name	String	用户名 最小长度：**1** 最大长度：**128**
login_permission	Boolean	是否有登录权限
root_permission	Boolean	是否有root权限
user_group_name	String	用户组 最小长度：**1** 最大长度：**128**
user_home_dir	String	用户目录 最小长度：**1** 最大长度：**256**
shell	String	用户启动shell 最小长度：**1** 最大长度：**128**
recent_scan_time	Long	最近扫描时间 最小值：**0** 最大值：**4070880000000**
container_id	String	容器id 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
container_name	String	容器名称 最小长度：**1** 最大长度：**256**

请求示例

默认查询账号为daemon的服务器列表

```
GET https://{endpoint}/v5/{project_id}/asset/users?user_name=daemon
```

响应示例

状态码： 200

账号信息列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "0bf792d910xxxxxxxxxx52cb7e63exxx",
    "host_id" : "13xxxxxxxece69",
    "host_ip" : "192.168.0.1",
    "host_name" : "test",
    "login_permission" : false,
    "recent_scan_time" : 1667039707730,
    "root_permission" : false,
    "shell" : "/sbin/nologin",
    "user_group_name" : "bin",
    "user_home_dir" : "/bin",
    "user_name" : "bin",
    "container_id" : "ce794b8a6-xxxx-xxxx-xxxxx-36bedf2c7a4f6083fb82e5bbc82709b50018",
    "container_name" : "hss_imagescan_W73V1WO6"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListUsersSolution {

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

```
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListUsersRequest request = new ListUsersRequest();
        request.withHostId("<host_id>");
        request.withUserName("<user_name>");
        request.withHostName("<host_name>");
        request.withPrivateIp("<private_ip>");
        request.withLoginPermission(<login_permission>);
        request.withRootPermission(<root_permission>);
        request.withUserGroup("<user_group>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        request.withPartMatch(<part_match>);
        try {
            ListUsersResponse response = client.listUsers(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

    client = HssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(HssRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
    try:
        request = ListUsersRequest()
        request.host_id = "<host_id>"
        request.user_name = "<user_name>"
        request.host_name = "<host_name>"
        request.private_ip = "<private_ip>"
        request.login_permission = <LoginPermission>
        request.root_permission = <RootPermission>
        request.user_group = "<user_group>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        request.part_match = <PartMatch>
        response = client.list_users(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListUsersRequest{}
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    userNameRequest:= "<user_name>"
    request.UserName = &userNameRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
    loginPermissionRequest:= <login_permission>
    request.LoginPermission = &loginPermissionRequest
    rootPermissionRequest:= <root_permission>
    request.RootPermission = &rootPermissionRequest
    userGroupRequest:= "<user_group>"
    request.UserGroup = &userGroupRequest
```

```
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    partMatchRequest:= <part_match>
    request.PartMatch = &partMatchRequest
    response, err := client.ListUsers(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	账号信息列表

错误码

请参见**错误码**。

3.1.8 查询单服务器的开放端口列表

功能介绍

查询单服务器的开放端口列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/ports

表 3-35 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-36 Query 参数

参数	是否必选	参数类型	描述
host_id	是	String	主机id 最小长度：**0** 最大长度：**128**
host_name	否	String	主机名称 最小长度：**0** 最大长度：**128**
host_ip	否	String	主机ip 最小长度：**0** 最大长度：**128**
port	否	Integer	端口号 最小值：**1** 最大值：**65535**
type	否	String	端口类型：目前包括TCP，UDP两种 最小长度：**0** 最大长度：**128**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： • host：主机 • container：容器 最小长度：**0** 最大长度：**64**

请求参数

表 3-37 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-38 响应 Body 参数

参数	参数类型	描述
total_num	Integer	开放端口总数 最小值：**0** 最大值：**10000**
data_list	Array of **PortResponseInfo** objects	端口信息列表 数组长度：**0 - 10000**

表 3-39 PortResponseInfo

参数	参数类型	描述
host_id	String	主机id 最小长度：**1** 最大长度：**128**
laddr	String	监听ip 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
status	String	port status, normal, danger or unknow ● "normal" : 正常 ● "danger" : 危险 ● "unknown" : 未知 最小长度：**1** 最大长度：**10**
port	Integer	端口号 最小值：**0** 最大值：**65535**
type	String	端口类型：目前包括TCP，UDP两种 最小长度：**1** 最大长度：**64**
pid	Integer	进程ID 最小值：**1** 最大值：**65535**
path	String	进程可执行文件路径 最小长度：**1** 最大长度：**256**
agent_id	String	Agent ID 最小长度：**1** 最大长度：**64**
container_id	String	容器id 最小长度：**0** 最大长度：**128**

请求示例

默认查询前10条host_id为dd91cd32-a238-4c0e-bc01-3b11653714ac的开放端口列表

```
GET https://{endpoint}/v5/{project_id}/asset/ports?hlimit=10&offset=0&host_id=dd91cd32-a238-4c0e-bc01-3b11653714ac
```

响应示例

状态码： 200

端口信息列表

```
{
  "data_list" : [ {
    "agent_id" : "eb5d03f02fffd85aaf5d0ba5c992d97713244f420e0b076dcf6ae0574c78aa4b",
    "container_id" : "",
```

```
    "host_id" : "dd91cd32-a238-4c0e-bc01-3b11653714ac",
    "laddr" : "0.0.0.0",
    "path" : "/usr/sbin/dhclient",
    "pid" : 1507,
    "port" : 68,
    "status" : "unknow",
    "type" : "UDP"
  }, {
    "agent_id" : "eb5d03f02fffd85aaf5d0ba5c992d97713244f420e0b076dcf6ae0574c78aa4b",
    "container_id" : "",
    "host_id" : "dd91cd32-a238-4c0e-bc01-3b11653714ac",
    "laddr" : "127.0.0.1",
    "path" : "/usr/sbin/chronyd",
    "pid" : 493,
    "port" : 323,
    "status" : "unknow",
    "type" : "UDP"
  } ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListPortsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPortsRequest request = new ListPortsRequest();
        request.withHostId("<host_id>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withPort(<port>);
        request.withType("<type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
```

```
        try {
            ListPortsResponse response = client.listPorts(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListPortsRequest()
        request.host_id = "<host_id>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.port = <port>
        request.type = "<type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        response = client.list_ports(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"
```

```
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPortsRequest{}
    request.HostId = "<host_id>"
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    portRequest:= int32(<port>)
    request.Port = &portRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    response, err := client.ListPorts(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	端口信息列表

错误码

请参见**错误码**。

3.1.9 查询软件的服务器列表

功能介绍

查询软件的服务器列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/apps

表 3-40 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-41 Query 参数

参数	是否必选	参数类型	描述
host_id	是	String	主机id 最小长度：**0** 最大长度：**128**
host_name	否	String	主机名称 最小长度：**0** 最大长度：**128**
app_name	否	String	软件名称 最小长度：**0** 最大长度：**128**
host_ip	否	String	主机ip 最小长度：**0** 最大长度：**128**

参数	是否必选	参数类型	描述
version	否	String	软件版本号 最小长度：**0** 最大长度：**128**
install_dir	否	String	安装目录 最小长度：**0** 最大长度：**512**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**
part_match	否	Boolean	是否模糊匹配，默认false表示精确匹配

请求参数

表 3-42 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-43 响应 Body 参数

参数	参数类型	描述
total_num	Integer	软件总数 最小值：**0** 最大值：**10000**
data_list	Array of **AppResponseInfo** objects	软件列表 数组长度：**0 - 10000**

表 3-44 AppResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
host_id	String	主机id 最小长度：**1** 最大长度：**128**
host_name	String	服务器名称 最小长度：**1** 最大长度：**256**

参数	参数类型	描述
host_ip	String	服务器ip 最小长度：**1** 最大长度：**256**
app_name	String	软件名称 最小长度：**1** 最大长度：**128**
version	String	版本号 最小长度：**1** 最大长度：**128**
update_time	Long	更新时间，最近一次更新的时间，用毫秒表示 最小值：**0** 最大值：**2147483647**
recent_scan_time	Long	最近扫描时间，用毫秒表示 最小值：**0** 最大值：**2147483647**
container_id	String	容器id 最小长度：**1** 最大长度：**128**
container_name	String	容器名称 最小长度：**1** 最大长度：**256**

请求示例

默认查询前10条软件名称为acl的服务器列表

```
GET https://{endpoint}/v5/{project_id}/asset/apps?app_name=acl
```

响应示例

状态码： 200

单台主机安装的app

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8",
    "host_id" : "55dac7fe-d81b-43bc-a4a7-4710fe673972",
    "host_name" : "xxxx",
    "host_ip" : "192.168.0.126",
    "app_name" : "acl",
    "version" : "2.2.51-14.eulerosv2r7",
    "update_time" : 1668150671981,
```

```
  "recent_scan_time" : 1668506044147,
  "container_id" : "ce794b8a6071f5fd7e4d142dab7b36bedf2c7a4f6083fb82e5bbc82709b50018",
  "container_name" : "hss_imagescan_W73V1WO6"
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAppsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAppsRequest request = new ListAppsRequest();
        request.withHostId("<host_id>");
        request.withHostName("<host_name>");
        request.withAppName("<app_name>");
        request.withHostIp("<host_ip>");
        request.withVersion("<version>");
        request.withInstallDir("<install_dir>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withCategory("<category>");
        request.withPartMatch(<part_match>);
        try {
            ListAppsResponse response = client.listApps(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAppsRequest()
        request.host_id = "<host_id>"
        request.host_name = "<host_name>"
        request.app_name = "<app_name>"
        request.host_ip = "<host_ip>"
        request.version = "<version>"
        request.install_dir = "<install_dir>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.category = "<category>"
        request.part_match = <PartMatch>
        response = client.list_apps(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAppsRequest{}
    request.HostId = "<host_id>"
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    appNameRequest:= "<app_name>"
    request.AppName = &appNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    versionRequest:= "<version>"
    request.Version = &versionRequest
    installDirRequest:= "<install_dir>"
    request.InstallDir = &installDirRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    partMatchRequest:= <part_match>
    request.PartMatch = &partMatchRequest
    response, err := client.ListApps(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	单台主机安装的app

错误码

请参见**错误码**。

3.1.10 查询自启动项的服务列表

功能介绍

查询自启动项的服务列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/auto-launchs

表 3-45 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-46 Query 参数

参数	是否必选	参数类型	描述
host_id	否	String	主机id 最小长度：**1** 最大长度：**128**
host_name	否	String	主机名称 最小长度：**1** 最大长度：**128**
name	否	String	自启动项名称 最小长度：**1** 最大长度：**256**
host_ip	否	String	主机ip 最小长度：**1** 最大长度：**128**

参数	是否必选	参数类型	描述
type	否	String	自启动项类型 ● 0： 自启动服务 ● 1： 定时任务 ● 2： 预加载动态库 ● 3： Run注册表键 ● 4： 开机启动文件夹 最小长度：**1** 最大长度：**128**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
part_match	否	Boolean	是否模糊匹配，默认false表示精确匹配

请求参数

表 3-47 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-48 响应 Body 参数

参数	参数类型	描述
total_num	Integer	自启动项总数 最小值：**0** 最大值：**10000**
data_list	Array of **AutoLauchResponseInfo** objects	自启动项列表 数组长度：**0 - 10000**

表 3-49 AutoLauchResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
host_id	String	主机id 最小长度：**1** 最大长度：**128**
host_name	String	服务器名称 最小长度：**1** 最大长度：**256**
host_ip	String	服务器ip 最小长度：**1** 最大长度：**256**
name	String	自启动项名称 最小长度：**1** 最大长度：**256**

参数	参数类型	描述
type	Integer	自启动项类型 ● 0： 自启动服务 ● 1： 定时任务 ● 2： 预加载动态库 ● 3： Run注册表键 ● 4： 开机启动文件夹 最小值：**0** 最大值：**11**
path	String	自启动项的路径 最小长度：**1** 最大长度：**256**
hash	String	采用sha256算法生成的文件hash值 最小长度：**1** 最大长度：**128**
run_user	String	运行用户 最小长度：**1** 最大长度：**128**
recent_scan_time	Long	最近扫描时间 最小值：**0** 最大值：**4824430336000**

请求示例

默认查询前10条自启动项名称为S50multi-queue的服务列表

```
GET https://{endpoint}/v5/{project_id}/asset/auto-launchs?name=S50multi-queue
```

响应示例

状态码： 200

auto launch list

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "9e742932bff2894e3d0869d03989b05cefb27a6cbc201d98c4465296xxxxxxxx",
    "host_id" : "3d0581a5-03b9-4311-9149-c026b0726a7e",
    "host_name" : "name",
    "host_ip" : "3d0581a5-03b9-4311-9149-c026b0726a7e",
    "name" : "S12hostguard",
    "type" : 0,
    "path" : "/etc/hostguard",
    "hash" : "xxxxxxxx227bffa0c04425ba6c8e0024046caa38dfbca6281b40109axxxxxxxx",
    "run_user" : "user",
    "recent_scan_time" : 1668240858425
```

```
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAutoLaunchsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAutoLaunchsRequest request = new ListAutoLaunchsRequest();
        request.withHostId("<host_id>");
        request.withHostName("<host_name>");
        request.withName("<name>");
        request.withHostIp("<host_ip>");
        request.withType("<type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withPartMatch(<part_match>);
        try {
            ListAutoLaunchsResponse response = client.listAutoLaunchs(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAutoLaunchsRequest()
        request.host_id = "<host_id>"
        request.host_name = "<host_name>"
        request.name = "<name>"
        request.host_ip = "<host_ip>"
        request.type = "<type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.part_match = <PartMatch>
        response = client.list_auto_launchs(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

```
        Build()

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAutoLaunchsRequest{}
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    nameRequest:= "<name>"
    request.Name = &nameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    partMatchRequest:= <part_match>
    request.PartMatch = &partMatchRequest
    response, err := client.ListAutoLaunchs(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	auto launch list

错误码

请参见**错误码**。

3.1.11 获取账户变动历史信息

功能介绍

获取账户变动历史记录信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/user/change-history

表 3-50 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-51 Query 参数

参数	是否必选	参数类型	描述
user_name	否	String	账号名 最小长度：**1** 最大长度：**128**
host_id	否	String	主机id 最小长度：**1** 最大长度：**128**
root_permission	否	Boolean	是否有root权限
host_name	否	String	主机名称 最小长度：**1** 最大长度：**128**
private_ip	否	String	服务器私有IP 最小长度：**1** 最大长度：**128**
change_type	否	String	账号变更类型: ● ADD ： 添加 ● DELETE ： 删除 ● MODIFY ： 修改 最小长度：**1** 最大长度：**128**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
start_time	否	Long	变更开始时间，13位时间戳 最小值：**0** 最大值：**4070880000000**
end_time	否	Long	变更结束时间，13位时间戳 最小值：**0** 最大值：**4070880000000**

请求参数

表 3-52 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-53 响应 Body 参数

参数	参数类型	描述
total_num	Integer	账号变动总数 最小值：**0** 最大值：**10000000**

参数	参数类型	描述
data_list	Array of **UserChangeHistoryResponseInfo** objects	账号历史变动记录列表 数组长度：**0 - 200**

表 3-54 UserChangeHistoryResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**1** 最大长度：**128**
change_type	String	变更类型 • ADD ： 添加 • DELETE ： 删除 • MODIFY ： 修改 最小长度：**1** 最大长度：**128**
host_id	String	主机ID 最小长度：**1** 最大长度：**128**
host_name	String	服务器名称 最小长度：**1** 最大长度：**128**
private_ip	String	服务器私有IP 最小长度：**1** 最大长度：**128**
login_permission	Boolean	是否有登录权限
root_permission	Boolean	是否有root权限
user_group_name	String	用户组 最小长度：**1** 最大长度：**128**
user_home_dir	String	用户目录 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
shell	String	用户启动shell 最小长度：**1** 最大长度：**128**
user_name	String	账号名称 最小长度：**1** 最大长度：**128**
expire_time	Long	到期时间，采用时间戳，默认毫秒， 最小值：**0** 最大值：**4070880000000**
recent_scan_time	Long	账号增加、修改、删除等操作的变更时间 最小值：**0** 最大值：**4070880000000**

请求示例

默认查询前10条开始时间为1700446129130，结束时间为1701050929130的账户变动历史记录信息

```
GET https://{endpoint}/v5/{project_id}/asset/user/change-history?
start_time=1700446129130&end_time=1701050929130
```

响应示例

状态码： 200

账号历史变动记录列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "0bf792d910xxxxxxxxxx52cb7e63exxx",
    "host_id" : "13xxxxxxece69",
    "private_ip" : "192.168.0.1",
    "host_name" : "test",
    "user_home_dir" : "/test",
    "login_permission" : false,
    "recent_scan_time" : 1667039707730,
    "expire_time" : 1667039707730,
    "root_permission" : false,
    "shell" : "/sbin/nologin",
    "user_group_name" : "bin",
    "user_name" : "bin",
    "change_type" : "ADD"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListUserChangeHistoriesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListUserChangeHistoriesRequest request = new ListUserChangeHistoriesRequest();
        request.withUserName("<user_name>");
        request.withHostId("<host_id>");
        request.withRootPermission(<root_permission>);
        request.withHostName("<host_name>");
        request.withPrivateIp("<private_ip>");
        request.withChangeType("<change_type>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withStartTime(<start_time>L);
        request.withEndTime(<end_time>L);
        try {
            ListUserChangeHistoriesResponse response = client.listUserChangeHistories(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListUserChangeHistoriesRequest()
        request.user_name = "<user_name>"
        request.host_id = "<host_id>"
        request.root_permission = <RootPermission>
        request.host_name = "<host_name>"
        request.private_ip = "<private_ip>"
        request.change_type = "<change_type>"
        request.limit = <limit>
        request.offset = <offset>
        request.enterprise_project_id = "<enterprise_project_id>"
        request.start_time = <start_time>
        request.end_time = <end_time>
        response = client.list_user_change_histories(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

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

    request := &model.ListUserChangeHistoriesRequest{}
    userNameRequest:= "<user_name>"
    request.UserName = &userNameRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    rootPermissionRequest:= <root_permission>
    request.RootPermission = &rootPermissionRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
    changeTypeRequest:= "<change_type>"
    request.ChangeType = &changeTypeRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    startTimeRequest:= int64(<start_time>)
    request.StartTime = &startTimeRequest
    endTimeRequest:= int64(<end_time>)
    request.EndTime = &endTimeRequest
    response, err := client.ListUserChangeHistories(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	账号历史变动记录列表

错误码

请参见**错误码**。

3.1.12 获取软件信息的历史变动记录

功能介绍

获取软件信息的历史变动记录

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/app/change-history

表 3-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-56 Query 参数

参数	是否必选	参数类型	描述
host_id	否	String	主机id 最小长度：**0** 最大长度：**128**
host_ip	否	String	主机ip 最小长度：**0** 最大长度：**128**
host_name	否	String	主机名称 最小长度：**0** 最大长度：**128**
app_name	否	String	软件名称 最小长度：**0** 最大长度：**128**
variation_type	否	String	变更类型: • add： 新建 • delete： 删除 • modify： 修改 最小长度：**0** 最大长度：**10**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**256**

参数	是否必选	参数类型	描述
sort_key	否	String	排序的key值，目前只支持按照recent_scan_time排序，按照recent_scan_time排序时，根据sort_dir的值决定升序还是降序 最小长度：**1** 最大长度：**128**
sort_dir	否	String	排序方式，默认为降序，当sort_key为按照recent_scan_time排序时，根据当前值决定升序还是降序，当sort_key为其他值时均为降序 ● asc ： 升序 ● desc ： 降序 最小长度：**1** 最大长度：**32**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**
start_time	否	Long	变更开始时间，13位时间戳 最小值：**0** 最大值：**9007199254740992**
end_time	否	Long	变更结束时间，13位时间戳 最小值：**0** 最大值：**9007199254740992**

请求参数

表 3-57 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-58 响应 Body 参数

参数	参数类型	描述
total_num	Integer	软件变动总数 最小值：**0** 最大值：**10000**
data_list	Array of **AppChangeResponseInfo** objects	软件历史变动记录列表 数组长度：**0 - 10000**

表 3-59 AppChangeResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
variation_type	String	the type of change • add ： 新建 • delete ： 删除 • modify ： 修改 最小长度：**0** 最大长度：**10**

参数	参数类型	描述
host_id	String	host_id 最小长度：**1** 最大长度：**128**
app_name	String	软件名称 最小长度：**1** 最大长度：**128**
host_name	String	服务器名称 最小长度：**1** 最大长度：**128**
host_ip	String	服务器ip 最小长度：**1** 最大长度：**256**
version	String	版本号 最小长度：**1** 最大长度：**128**
update_time	Long	软件更新时间 最小值：**0** 最大值：**4824430336000**
recent_scan_time	Long	最近扫描时间 最小值：**0** 最大值：**4824430336000**

请求示例

默认查询前10条开始时间为1700446175490，结束时间为1701050975490的软件信息的历史变动记录

```
GET https://{endpoint}/v5/{project_id}/asset/app/change-history?
start_time=1700446175490&end_time=1701050975490
```

响应示例

状态码： 200

App change history info list

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "d83c7be8a106485a558f97446617443b87604c8116e3cf0453c2a44exxxxxxxx",
    "variation_type" : "add",
    "host_id" : "f4aaca51-xxxx-xxxx-xxxx-891c9e84d885",
    "app_name" : "hostguard",
    "host_name" : "host_name",
```

```
    "host_ip" : "host_ip",
    "version" : "3.2.3",
    "update_time" : 1668246126302,
    "recent_scan_time" : 1668246126302
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAppChangeHistoriesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAppChangeHistoriesRequest request = new ListAppChangeHistoriesRequest();
        request.withHostId("<host_id>");
        request.withHostIp("<host_ip>");
        request.withHostName("<host_name>");
        request.withAppName("<app_name>");
        request.withVariationType("<variation_type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withSortKey("<sort_key>");
        request.withSortDir("<sort_dir>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withStartTime(<start_time>L);
        request.withEndTime(<end_time>L);
        try {
            ListAppChangeHistoriesResponse response = client.listAppChangeHistories(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAppChangeHistoriesRequest()
        request.host_id = "<host_id>"
        request.host_ip = "<host_ip>"
        request.host_name = "<host_name>"
        request.app_name = "<app_name>"
        request.variation_type = "<variation_type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.sort_key = "<sort_key>"
        request.sort_dir = "<sort_dir>"
        request.limit = <limit>
        request.offset = <offset>
        request.start_time = <start_time>
        request.end_time = <end_time>
        response = client.list_app_change_histories(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAppChangeHistoriesRequest{}
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    appNameRequest:= "<app_name>"
    request.AppName = &appNameRequest
    variationTypeRequest:= "<variation_type>"
    request.VariationType = &variationTypeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    sortKeyRequest:= "<sort_key>"
    request.SortKey = &sortKeyRequest
    sortDirRequest:= "<sort_dir>"
    request.SortDir = &sortDirRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    startTimeRequest:= int64(<start_time>)
    request.StartTime = &startTimeRequest
    endTimeRequest:= int64(<end_time>)
    request.EndTime = &endTimeRequest
    response, err := client.ListAppChangeHistories(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	App change history info list

错误码

请参见**错误码**。

3.1.13 获取自启动项的历史变动记录

功能介绍

获取自启动项的历史变动记录

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/auto-launch/change-history

表 3-60 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-61 Query 参数

参数	是否必选	参数类型	描述
host_id	否	String	主机id 最小长度：**0** 最大长度：**128**
host_ip	否	String	主机ip 最小长度：**0** 最大长度：**128**
host_name	否	String	主机名称 最小长度：**0** 最大长度：**128**
auto_launch_name	否	String	自启动项名称 最小长度：**0** 最大长度：**128**

参数	是否必选	参数类型	描述
type	否	Integer	自启动项类型 ● 0： 自启动服务 ● 1： 定时任务 ● 2： 预加载动态库 ● 3： Run注册表键 ● 4： 开机启动文件夹 最小值：**0** 最大值：**100**
variation_type	否	String	变更类型: ● add： 新建 ● delete： 删除 ● modify： 修改 最小长度：**0** 最大长度：**10**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**256**
sort_key	否	String	排序的key值，目前只支持按照recent_scan_time排序，按照recent_scan_time排序时，根据sort_dir的值决定升序还是降序 最小长度：**0** 最大长度：**128**
sort_dir	否	String	排序方式，默认为降序，当sort_key为按照recent_scan_time排序时，根据当前值决定升序还是降序，当sort_key为其他值时均为降序 ● asc： 升序 ● desc： 降序 最小长度：**0** 最大长度：**32**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
start_time	否	Long	变更开始时间，13位时间戳 最小值：**0** 最大值：**9007199254740992**
end_time	否	Long	变更结束时间，13位时间戳 最小值：**0** 最大值：**9007199254740992**

请求参数

表 3-62 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-63 响应 Body 参数

参数	参数类型	描述
total_num	Integer	自启动项变动总数 最小值：**0** 最大值：**10000**
data_list	Array of **AutoLaunchChangeResponseInfo** objects	软件历史变动记录列表 数组长度：**0 - 10000**

表 3-64 AutoLaunchChangeResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
variation_type	String	the type of change ● add ： 新建 ● delete ： 删除 ● modify ： 修改 最小长度：**0** 最大长度：**10**
type	Integer	自启动项类型 ● 0 ： 自启动服务 ● 1 ： 定时任务 ● 2 ： 预加载动态库 ● 3 ： Run注册表键 ● 4 ： 开机启动文件夹 最小值：**0** 最大值：**11**
host_id	String	host_id 最小长度：**1** 最大长度：**128**
host_name	String	弹性服务器名称 最小长度：**1** 最大长度：**256**
host_ip	String	主机IP 最小长度：**1** 最大长度：**256**
path	String	自启动项的路径 最小长度：**1** 最大长度：**256**
hash	String	采用sha256算法生成的文件hash值 最小长度：**1** 最大长度：**128**
run_user	String	运行用户 最小长度：**1** 最大长度：**64**

参数	参数类型	描述
name	String	自启动项名称 最小长度：**1** 最大长度：**256**
recent_scan_time	Long	最近更新时间，13位时间戳 最小值：**0** 最大值：**4824430336000**

请求示例

默认查询前10条开始时间为1693101881568，结束时间为1701050681569的自启动项的历史变动记录

```
GET https://{endpoint}/v5/{project_id}/asset/auto-launch/change-history?
start_time=1693101881568&end_time=1701050681569
```

响应示例

状态码： 200

App change history info list

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "d83c7be8a106485a558f97446617443b87604c8116e3cf0453c2a44exxxxxxxx",
    "variation_type" : "add",
    "type" : 0,
    "host_id" : "host_id",
    "host_name" : "host_name",
    "host_ip" : "host_ip",
    "path" : "/path",
    "hash" : "xxxxxxxx227bffa0c04425ba6c8e0024046caa38dfbca6281b40109axxxxxxxx",
    "run_user" : "SYSTEM",
    "name" : "S12hostguard",
    "recent_scan_time" : 1668246126302
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;
```

```
public class ListAutoLaunchChangeHistoriesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAutoLaunchChangeHistoriesRequest request = new ListAutoLaunchChangeHistoriesRequest();
        request.withHostId("<host_id>");
        request.withHostIp("<host_ip>");
        request.withHostName("<host_name>");
        request.withAutoLaunchName("<auto_launch_name>");
        request.withType(<type>);
        request.withVariationType("<variation_type>");
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withSortKey("<sort_key>");
        request.withSortDir("<sort_dir>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withStartTime(<start_time>L);
        request.withEndTime(<end_time>L);
        try {
            ListAutoLaunchChangeHistoriesResponse response = client.listAutoLaunchChangeHistories(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAutoLaunchChangeHistoriesRequest()
        request.host_id = "<host_id>"
        request.host_ip = "<host_ip>"
        request.host_name = "<host_name>"
        request.auto_launch_name = "<auto_launch_name>"
        request.type = <type>
        request.variation_type = "<variation_type>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.sort_key = "<sort_key>"
        request.sort_dir = "<sort_dir>"
        request.limit = <limit>
        request.offset = <offset>
        request.start_time = <start_time>
        request.end_time = <end_time>
        response = client.list_auto_launch_change_histories(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAutoLaunchChangeHistoriesRequest{}
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
```

```
    autoLaunchNameRequest:= "<auto_launch_name>"
    request.AutoLaunchName = &autoLaunchNameRequest
    typeRequest:= int32(<type>)
    request.Type = &typeRequest
    variationTypeRequest:= "<variation_type>"
    request.VariationType = &variationTypeRequest
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    sortKeyRequest:= "<sort_key>"
    request.SortKey = &sortKeyRequest
    sortDirRequest:= "<sort_dir>"
    request.SortDir = &sortDirRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    startTimeRequest:= int64(<start_time>)
    request.StartTime = &startTimeRequest
    endTimeRequest:= int64(<end_time>)
    request.EndTime = &endTimeRequest
    response, err := client.ListAutoLaunchChangeHistories(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	App change history info list

错误码

请参见**错误码**。

3.1.14 资产指纹-进程-服务器列表

功能介绍

具备该进程的主机/容器信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/processes/detail

表 3-65 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-66 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
host_name	否	String	主机名称 最小长度：**1** 最大长度：**256**
host_ip	否	String	主机ip 最小长度：**1** 最大长度：**256**
path	否	String	进程可执行文件路径 最小长度：**1** 最大长度：**256**
category	否	String	类型，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-67 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-68 响应 Body 参数

参数	参数类型	描述
total_num	Integer	主机统计信息总数, 最小值：**0** 最大值：**10000**
data_list	Array of **ProcessesHostResponseInfo** objects	主机统计信息列表 数组长度：**0 - 10000**

表 3-69 ProcessesHostResponseInfo

参数	参数类型	描述
hash	String	path对应的sha256值 最小长度：**1** 最大长度：**256**
host_ip	String	主机ip 最小长度：**1** 最大长度：**256**
host_name	String	主机名称 最小长度：**1** 最大长度：**256**

参数	参数类型	描述
launch_params	String	启动参数 最小长度：**1** 最大长度：**256**
launch_time	Long	启动时间 最小值：**0** 最大值：**4070880000000**
process_path	String	进程可执行文件路径 最小长度：**1** 最大长度：**256**
process_pid	Integer	进程pid 最小值：**0** 最大值：**65535**
run_permission	String	文件权限 最小长度：**1** 最大长度：**256**
container_id	String	容器id 最小长度：**1** 最大长度：**128**
container_name	String	容器名称 最小长度：**1** 最大长度：**256**

请求示例

默认查询前10条进程路径为/usr/bin/bash的主机列表

```
GET https://{endpoint}/v5/{project_id}/asset/processes/detail?path=/usr/bin/bash
```

响应示例

状态码： 200

具备该进程的主机信息

```
{
  "total_num" : 1,
  "data_list" : [ {
    "hash" : "xxxxxx96a7ceb67731c0158xxxxxxff8456914d8275d221671d1190e888xxxxx",
    "host_ip" : "192.168.0.1",
    "host_name" : "ecs-euler-z00800211",
    "launch_params" : "",
    "launch_time" : 1673504622000,
    "process_path" : "/CloudResetPwdUpdateAgent/bin/wrapper",
    "process_pid" : 888,
```

```
    "run_permission" : "rwx------",
    "container_id" : "ce794b8a6071f5fd7e4d142dab7b36bedf2c7a4f6083fb82e5bbc82709b50018",
    "container_name" : "hss_imagescan_W73V1WO6"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListProcessesHostSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListProcessesHostRequest request = new ListProcessesHostRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withPath("<path>");
        request.withCategory("<category>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListProcessesHostResponse response = client.listProcessesHost(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListProcessesHostRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.path = "<path>"
        request.category = "<category>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_processes_host(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

```
    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProcessesHostRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    pathRequest:= "<path>"
    request.Path = &pathRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListProcessesHost(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	具备该进程的主机信息

错误码

请参见**错误码**。

3.1.15 资产指纹-端口-服务器列表

功能介绍

具备该端口的主机/容器信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/ports/detail

表 3-70 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-71 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**1** 最大长度：**256**
host_name	否	String	主机名称 最小长度：**1** 最大长度：**256**
host_ip	否	String	主机ip 最小长度：**1** 最大长度：**256**
port	是	Integer	端口号 最小值：**1** 最大值：**65535**
type	否	String	端口类型：目前包括TCP，UDP两种 最小长度：**1** 最大长度：**256**
category	否	String	类别，默认为host，包含如下： ● host：主机 ● container：容器 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**10000** 缺省值：**0**

请求参数

表 3-72 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） 最小长度：**32** 最大长度：**4096**

响应参数

状态码： 200

表 3-73 响应 Body 参数

参数	参数类型	描述
total_num	Integer	机器总数 最小值：**0** 最大值：**10000**
data_list	Array of **PortHostResponseInfo** objects	机器信息列表 数组长度：**0 - 10000**

表 3-74 PortHostResponseInfo

参数	参数类型	描述
container_id	String	镜像id 最小长度：**1** 最大长度：**256**
host_id	String	主机id 最小长度：**1** 最大长度：**256**
host_ip	String	主机ip 最小长度：**1** 最大长度：**256**
host_name	String	主机名称 最小长度：**1** 最大长度：**256**
laddr	String	监听ip 最小长度：**1** 最大长度：**256**
path	String	进程可执行文件路径 最小长度：**1** 最大长度：**256**
pid	Integer	pid 最小值：**0** 最大值：**100000**
port	Integer	端口 最小值：**0** 最大值：**65535**
status	String	状态 最小长度：**1** 最大长度：**256**
type	String	端口类型：目前包括TCP，UDP两种 最小长度：**1** 最大长度：**256**
container_name	String	容器名称 最小长度：**1** 最大长度：**256**

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**1** 最大长度：**128**

请求示例

默认查询前10条端口为22的主机列表

```
GET https://{endpoint}/v5/{project_id}/asset/ports/detail?port=22
```

响应示例

状态码： 200

具备该端口的主机信息

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_id" : "03117200-xxxx-xxxx-xxxx-a89a10e66dbe",
    "host_ip" : "192.168.0.1",
    "host_name" : "ecs-eule",
    "laddr" : "0.0.0.0",
    "path" : "C:\\Windows\\system32\\svchost.exe",
    "port" : 888,
    "status" : "unknow",
    "type" : "UDP",
    "container_id" : "ce794b8a6-xxxx-xxxx-xxxxx-36bedf2c7a4f6083fb82e5bbc82709b50018",
    "container_name" : "hss_imagescan_W73V1WO6",
    "agent_id" : "03jjj-xxxx-xxxx-wwwsedf"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListPortHostSolution {

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

```
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPortHostRequest request = new ListPortHostRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withPort(<port>);
        request.withType("<type>");
        request.withCategory("<category>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListPortHostResponse response = client.listPortHost(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListPortHostRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.port = <port>
```

```
        request.type = "<type>"
        request.category = "<category>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_port_host(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPortHostRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    request.Port = int32(<port>)
    typeRequest:= "<type>"
    request.Type = &typeRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListPortHost(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	具备该端口的主机信息

错误码

请参见**错误码**。

3.1.16 查询中间件列表

功能介绍

查询中间件列表，支持通过中间件名称查询对应的服务器树

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/midwares

表 3-75 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-76 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
file_name	否	String	jar包名称 最小长度：**0** 最大长度：**256**
category	否	String	类别，包含如下: • host : 主机 • container : 容器 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-77 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-78 响应 Body 参数

参数	参数类型	描述
total_num	Integer	Jar包统计信息总数 最小值：**0** 最大值：**10000**
data_list	Array of **JarPackageStatisticsResponseInfo** objects	Jar包统计信息列表 数组长度：**0 - 300000**

表 3-79 JarPackageStatisticsResponseInfo

参数	参数类型	描述
file_name	String	Jar包名称 最小长度：**0** 最大长度：**256**
num	Integer	Jar包统计信息总数 最小值：**0** 最大值：**300000**

请求示例

默认查询前10条中间件名称为rt.jar，类别为主机的中间件列表

```
GET https://{endpoint}/v5/{project_id}/asset/midwares?file_name=rt.jar&category=host
```

响应示例

状态码： 200

JarPackage statistics

```
{
  "data_list" : [ {
    "file_name" : "rt.jar",
    "num" : 18
  } ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListJarPackageStatisticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListJarPackageStatisticsRequest request = new ListJarPackageStatisticsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withFileName("<file_name>");
        request.withCategory("<category>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListJarPackageStatisticsResponse response = client.listJarPackageStatistics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListJarPackageStatisticsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.file_name = "<file_name>"
        request.category = "<category>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_jar_package_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListJarPackageStatisticsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    fileNameRequest:= "<file_name>"
    request.FileName = &fileNameRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
```

```
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListJarPackageStatistics(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	JarPackage statistics

错误码

请参见**错误码**。

3.1.17 查询指定中间件的服务器列表

功能介绍

查询指定中间件的服务器列表，通过传入中间件名称参数，返回对应的中间件服务器列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/asset/midwares/detail

表 3-80 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-81 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
file_name	是	String	文件名称 最小长度：**1** 最大长度：**256**
category	否	String	类别，包含如下: ● host : 主机 ● container : 容器 最小长度：**0** 最大长度：**64**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**64**
host_ip	否	String	服务器IP 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**300000** 缺省值：**0**
part_match	否	Boolean	是否模糊匹配，默认false表示精确匹配

请求参数

表 3-82 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-83 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**10000**
data_list	Array of **JarPackageHostInfo** objects	服务器列表 数组长度：**0 - 10000**

表 3-84 JarPackageHostInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**1** 最大长度：**64**
host_id	String	主机id 最小长度：**0** 最大长度：**128**
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
host_ip	String	服务器ip 最小长度：**0** 最大长度：**128**
file_name	String	Jar包名称 最小长度：**0** 最大长度：**256**
name	String	Jar包名称(不带后缀) 最小长度：**0** 最大长度：**256**
catalogue	String	Jar包类型 最小长度：**0** 最大长度：**32**
file_type	String	Jar包后缀 最小长度：**0** 最大长度：**32**
version	String	Jar包版本 最小长度：**0** 最大长度：**64**
path	String	Jar包路径 最小长度：**0** 最大长度：**512**
hash	String	Jar包hash 最小长度：**0** 最大长度：**512**
size	Integer	Jar包大小 最小值：**0** 最大值：**2147483647**
uid	Integer	uid 最小值：**0** 最大值：**2147483647**
gid	Integer	gid 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
mode	String	文件权限 最小长度：**0** 最大长度：**32**
pid	Integer	进程id 最小值：**0** 最大值：**2147483647**
proc_path	String	进程可执行文件路径 最小长度：**0** 最大长度：**1024**
container_id	String	容器实例id 最小长度：**0** 最大长度：**128**
container_name	String	容器名称 最小长度：**0** 最大长度：**256**
package_path	String	包路径 最小长度：**0** 最大长度：**1024**
is_embedded	Integer	显示的是否是嵌套包 最小值：**0** 最大值：**2147483647**
record_time	Long	扫描时间 最小值：**0** 最大值：**4070880000000**

请求示例

默认查询前10条中间件名称为log4j-core-2.8.2.jar，类别为主机的服务器列表

```
GET https://{endpoint}/v5/{project_id}/asset/midwares/detail?file_name=log4j-core-2.8.2.jar&category=host
```

响应示例

状态码： 200

ListJarPackageHostInfo

```
{
  "data_list" : [ {
    "agent_id" : "2d0fe7824005bf001220ad9d892e86f8af44a7d3608dab11165008ce439d3583",
    "catalogue" : "util",
    "container_id" : "",
```

```
    "file_name" : "rt.jar",
    "file_type" : "jar",
    "gid" : 0,
    "hash" : "04bf14e3b1da55d95561ca78cb29caa909410051dbe047e91ad6f5c1dedb8d6d",
    "host_id" : "103ed820-62e5-4754-b0f8-3e47b6dd49d2",
    "host_ip" : "192.168.1.76",
    "host_name" : "正在测试勿删",
    "mode" : "-rw-------",
    "name" : "Java Runtime Environment",
    "path" : "/CloudResetPwdUpdateAgent/depend/jre/lib/rt.jar",
    "pid" : 1614,
    "proc_path" : "/CloudResetPwdUpdateAgent/depend/jre/bin/java",
    "record_time" : 1690513169986,
    "uid" : 0,
    "version" : "1.8.0_252",
    "size" : 128,
    "container_name" : "aaaa",
    "package_path" : "/CloudResetPwdUpdateAgent/depend/jre/bin/java",
    "is_embedded" : 0
  } ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListJarPackageHostInfoSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListJarPackageHostInfoRequest request = new ListJarPackageHostInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withFileName("<file_name>");
        request.withCategory("<category>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
```

```
        request.withPartMatch(<part_match>);
        try {
            ListJarPackageHostInfoResponse response = client.listJarPackageHostInfo(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListJarPackageHostInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.file_name = "<file_name>"
        request.category = "<category>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.limit = <limit>
        request.offset = <offset>
        request.part_match = <PartMatch>
        response = client.list_jar_package_host_info(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"
```

```
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListJarPackageHostInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.FileName = "<file_name>"
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    partMatchRequest:= <part_match>
    request.PartMatch = &partMatchRequest
    response, err := client.ListJarPackageHostInfo(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	ListJarPackageHostInfo

错误码

请参见**错误码**。

3.2 勒索防护

3.2.1 查询勒索防护服务器列表

功能介绍

查询勒索防护服务器列表，与云备份服务配合使用。因此使用勒索相关接口之前确保该局点有云备份服务

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/ransomware/server

表 3-85 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-86 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
host_name	否	String	服务器名称
os_type	否	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。 最小长度：**0** 最大长度：**64**
host_ip	否	String	服务器IP地址 最小长度：**0** 最大长度：**256**
host_status	否	String	主机状态，包含如下3种。 ● 不传参默认为全部。 ● ACTIVE： 正在运行。 ● SHUTOFF： 关机。 最小长度：**1** 最大长度：**32**
last_days	否	Integer	查询时间范围天数，最近7天为last_days=7，若不填，则默认查询一天内的防护事件和已有备份数 最小值：**1** 最大值：**30**

请求参数

表 3-87 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-88 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2097152**
data_list	Array of **ProtectionServerInfo** objects	查询勒索防护服务器列表 数组长度：**0 - 10241**

表 3-89 ProtectionServerInfo

参数	参数类型	描述
host_id	String	服务器ID 最小长度：**0** 最大长度：**128**
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
host_name	String	服务器名称 最小长度：**0** 最大长度：**128**
host_ip	String	弹性公网IP地址 最小长度：**0** 最大长度：**128**
private_ip	String	私有IP地址 最小长度：**0** 最大长度：**128**
os_type	String	操作系统类型，包含如下2种。 • Linux ： Linux。 • Windows ： Windows。 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
os_name	String	系统名称 最小长度：**0** 最大长度：**128**
host_status	String	服务器状态，包含如下2种。 ● ACTIVE： 运行中。 ● SHUTOFF： 关机。 最小长度：**1** 最大长度：**32**
ransom_protection_status	String	勒索防护状态，包含如下4种。 ● closed： 关闭。 ● opened： 开启。 ● opening： 开启中。 ● closing： 关闭中。 最小长度：**0** 最大长度：**128**
agent_version	String	agent版本 最小长度：**1** 最大长度：**128**
protect_status	String	防护状态，包含如下2种。 ● closed： 未防护。 ● opened： 防护中。 最小长度：**1** 最大长度：**32**
group_id	String	服务器组ID 最小长度：**1** 最大长度：**128**
group_name	String	服务器组名称 最小长度：**1** 最大长度：**128**
protect_policy_id	String	防护策略ID 最小长度：**1** 最大长度：**128**
protect_policy_name	String	防护策略名称 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
backup_error	backup_error object	备份错误信息
backup_protection_status	String	是否开启备份，包含如下3种。 • failed_to_turn_on_backup: 无法开启备份 • closed： 关闭。 • opened： 开启。 最小长度：**0** 最大长度：**128**
count_protect_event	Integer	防护事件数 最小值：**0** 最大值：**2097152**
count_backuped	Integer	已有备份数 最小值：**0** 最大值：**2097152**
agent_status	String	Agent状态 最小长度：**1** 最大长度：**128**
version	String	主机开通的版本，包含如下7种输入。 • hss.version.null： 无。 • hss.version.basic： 基础版。 • hss.version.advanced： 专业版。 • hss.version.enterprise： 企业版。 • hss.version.premium： 旗舰版。 • hss.version.wtp： 网页防篡改版。 • hss.version.container.enterprise： 容器版。 最小长度：**1** 最大长度：**32**
host_source	String	服务器类型，包含如下3种输入。 • ecs： ecs。 • outside： 线下主机。 • workspace： 云桌面。 最小长度：**1** 最大长度：**32**
vault_id	String	存储库ID 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
vault_name	String	存储库名称 最小长度：**0** 最大长度：**128**
vault_size	Integer	总容量，单位GB 最小值：**0** 最大值：**2097152**
vault_used	Integer	已使用容量，单位MB 最小值：**0** 最大值：**2097152**
vault_allocated	Integer	已分配容量，单位GB，指绑定的服务器大小 最小值：**0** 最大值：**2097152**
vault_charging_mode	String	存储库创建模式，按需：post_paid，包周期：pre_paid 最小长度：**0** 最大长度：**128**
vault_status	String	存储库状态。 ● available ： 可用。 ● lock ： 被锁定。 ● frozen：冻结。 ● deleting：删除中。 ● error：错误。 最小长度：**0** 最大长度：**128**
backup_policy_id	String	备份策略ID，若为空，则为未绑定状态，若不为空，通过backup_policy_enabled字段判断策略是否启用 最小长度：**1** 最大长度：**128**
backup_policy_name	String	备份策略名称 最小长度：**1** 最大长度：**128**
backup_policy_enabled	Boolean	策略是否启用
resources_num	Integer	已绑定服务器（个） 最小值：**0** 最大值：**2097152**

表 3-90 backup_error

参数	参数类型	描述
error_code	Integer	错误编码，包含如下2种。 ● 0： 无错误信息。 ● 1： 已綁定至其它存储库，无法开启备份。 ● 2： 备份库已超过最大限额。 ● 3： CBR接口调用异常。 最小值：**0** 最大值：**128**
error_description	String	错误描述 最小长度：**1** 最大长度：**128**

请求示例

查询勒索防护服务器列表，不传limit默认返回10条。

```
GET https://{endpoint}/v5/{project_id}/ransomware/server
```

响应示例

状态码： 200

勒索病毒防护服务器列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "2758d2a61598fd9144cfa6b201049e7c0af8c3f1280cd24e3ec95a2f0811a2a2",
    "agent_status" : "online",
    "backup_error" : {
      "error_code" : 1,
      "error_description" : "已綁定至其它存储库，无法开启备份"
    },
    "ransom_protection_status" : "opened",
    "backup_protection_status" : "failed_to_turn_on_backup",
    "count_backuped" : 0,
    "count_protect_event" : 0,
    "group_id" : "7c659ea3-006f-4687-9f1c-6d975d955f37",
    "group_name" : "333",
    "host_id" : "caa958ad-a481-4d46-b51e-6861b8864515",
    "host_ip" : "100.85.119.68",
    "host_name" : "Euler",
    "host_status" : "ACTIVE",
    "os_name" : "EulerOS",
    "os_type" : "Linux",
    "private_ip" : "100.85.123.9",
    "protect_policy_id" : "0253edfd-30e7-439d-8f3f-17c54c99706",
    "protect_policy_name" : "tst",
    "protect_status" : "opened"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListProtectionServerSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListProtectionServerRequest request = new ListProtectionServerRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withHostName("<host_name>");
        request.withOsType("<os_type>");
        request.withHostIp("<host_ip>");
        request.withHostStatus("<host_status>");
        request.withLastDays(<last_days>);
        try {
            ListProtectionServerResponse response = client.listProtectionServer(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListProtectionServerRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.offset = <offset>
        request.limit = <limit>
        request.host_name = "<host_name>"
        request.os_type = "<os_type>"
        request.host_ip = "<host_ip>"
        request.host_status = "<host_status>"
        request.last_days = <last_days>
        response = client.list_protection_server(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
        Build())

    request := &model.ListProtectionServerRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    osTypeRequest:= "<os_type>"
    request.OsType = &osTypeRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    hostStatusRequest:= "<host_status>"
    request.HostStatus = &hostStatusRequest
    lastDaysRequest:= int32(<last_days>)
    request.LastDays = &lastDaysRequest
    response, err := client.ListProtectionServer(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	勒索病毒防护服务器列表

错误码

请参见**错误码**。

3.2.2 查询勒索病毒的防护策略列表

功能介绍

查询勒索病毒的防护策略列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/ransomware/protection/policy

表 3-91 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-92 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**200** 缺省值：**10**
policy_name	否	String	防护策略名称 最小长度：**0** 最大长度：**128**
protect_policy_id	否	String	防护策略id 最小长度：**0** 最大长度：**128**
operating_system	否	String	策略支持的操作系统，包含如下： ● Windows : Windows系统 ● Linux : Linux系统 最小长度：**0** 最大长度：**128**

请求参数

表 3-93 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-94 响应 Body 参数

参数	参数类型	描述
total_num	Integer	策略总数 最小值：**0** 最大值：**2097152**
data_list	Array of **ProtectionPolicyInfo** objects	查询防护策略列表 数组长度：**0 - 10241**

表 3-95 ProtectionPolicyInfo

参数	参数类型	描述
policy_id	String	策略ID 最小长度：**0** 最大长度：**128**
policy_name	String	策略名称 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
protection_mode	String	防护动作，包含如下2种。 ● alarm_and_isolation： 告警并自动隔离。 ● alarm_only： 仅告警。 最小长度：**0** 最大长度：**128**
bait_protection_status	String	是否开启诱饵防护，包含如下1种, 默认为开启防护诱饵防护。 ● opened： 开启。 ● closed： 关闭。 最小长度：**0** 最大长度：**128**
deploy_mode	String	是否开启动态诱饵防护，包含如下2种, 默认为关闭动态诱饵防护。 ● opened： 开启。 ● closed： 关闭。 最小长度：**0** 最大长度：**128**
protection_directory	String	防护目录 最小长度：**1** 最大长度：**128**
protection_type	String	防护文件类型，例如：docx，txt，avi 最小长度：**0** 最大长度：**128**
exclude_directory	String	排除目录，选填 最小长度：**1** 最大长度：**128**
runtime_detection_status	String	是否运行时检测，包含如下2种，暂时只有关闭一种状态，为保留字段。 ● opened： 开启。 ● closed： 关闭。 最小长度：**0** 最大长度：**128**
runtime_detection_directory	String	运行时检测目录，现在为保留字段 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
count_associated_server	Integer	关联server个数 最小值：**0** 最大值：**2097152**
operating_system	String	操作系统类型。 • Linux • Windows 最小长度：**0** 最大长度：**128**
process_whitelist	Array of **TrustProcessInfo** objects	进程白名单 数组长度：**0 - 20**
default_policy	Integer	是否为默认策略，包含如下2种。 • 0 ： 非默认策略。 • 1 ： 默认策略 最小值：**0** 最大值：**10**

表 3-96 TrustProcessInfo

参数	参数类型	描述
path	String	进程路径 最小长度：**0** 最大长度：**128**
hash	String	进程hash 最小长度：**0** 最大长度：**128**

请求示例

查询勒索病毒的防护策略列表，不传limit参数默认返回10条数据。

```
GET https://{endpoint}/v5/{project_id}/ransomware/protection/policy
```

响应示例

状态码： 200

防护策略列表

```
{
  "total_num" : 1,
```

```
 "data_list" : [ {
   "bait_protection_status" : "opened",
   "exclude_directory" : "/opt",
   "count_associated_server" : 0,
   "operating_system" : "Linux",
   "protection_mode" : "alarm_only",
   "policy_id" : "4117d16-074b-41ae-b7d7-9cc25ee258",
   "policy_name" : "test",
   "protection_directory" : "/dd",
   "protection_type" : "docx",
   "runtime_detection_status" : "closed"
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListProtectionPolicySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListProtectionPolicyRequest request = new ListProtectionPolicyRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withPolicyName("<policy_name>");
        request.withProtectPolicyId("<protect_policy_id>");
        request.withOperatingSystem("<operating_system>");
        try {
            ListProtectionPolicyResponse response = client.listProtectionPolicy(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListProtectionPolicyRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.offset = <offset>
        request.limit = <limit>
        request.policy_name = "<policy_name>"
        request.protect_policy_id = "<protect_policy_id>"
        request.operating_system = "<operating_system>"
        response = client.list_protection_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProtectionPolicyRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    policyNameRequest:= "<policy_name>"
    request.PolicyName = &policyNameRequest
    protectPolicyIdRequest:= "<protect_policy_id>"
    request.ProtectPolicyId = &protectPolicyIdRequest
    operatingSystemRequest:= "<operating_system>"
    request.OperatingSystem = &operatingSystemRequest
    response, err := client.ListProtectionPolicy(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	防护策略列表

错误码

请参见**错误码**。

3.2.3 修改勒索防护策略

功能介绍

修改勒索防护策略

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/ransomware/protection/policy

表 3-97 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-98 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-99 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-100 请求 Body 参数

参数	是否必选	参数类型	描述
policy_id	是	String	策略ID 最小长度：**0** 最大长度：**128**
policy_name	是	String	策略名称 最小长度：**0** 最大长度：**128**
protection_mode	是	String	防护动作，包含如下2种。 ● alarm_and_isolation ： 告警并自动隔离。 ● alarm_only ： 仅告警。 最小长度：**0** 最大长度：**128**
bait_protection_status	否	String	是否开启诱饵防护，包含如下1种, 默认为开启防护诱饵防护。 ● opened ： 开启。 ● closed ： 关闭。 最小长度：**0** 最大长度：**128**
protection_directory	是	String	防护目录,多个目录请用英文分号隔开，最多支持填写20个防护目录 最小长度：**1** 最大长度：**128**
protection_type	是	String	防护文件类型，例如：docx，txt，avi 最小长度：**1** 最大长度：**128**
exclude_directory	否	String	排除目录(选填)，多个目录请用英文分号隔开，最多支持填写20个排除目录 最小长度：**1** 最大长度：**128**
agent_id_list	否	Array of strings	开启了此勒索防护策略的agent的id列表 最小长度：**1** 最大长度：**128** 数组长度：**0 - 10000**

参数	是否必选	参数类型	描述
operating_system	是	String	支持该策略的操作系统，包含如下： • Windows : Windows系统 • Linux : Linux系统 最小长度：**0** 最大长度：**64**
runtime_detection_status	否	String	是否运行时检测，包含如下2种，暂时只有关闭一种状态，为保留字段。 • opened ： 开启。 • closed ： 关闭。 最小长度：**0** 最大长度：**128**
process_whitelist	否	Array of **TrustProcessInfo** objects	进程白名单 数组长度：**0 - 20**

表 3-101 TrustProcessInfo

参数	是否必选	参数类型	描述
path	否	String	进程路径 最小长度：**0** 最大长度：**128**
hash	否	String	进程hash 最小长度：**0** 最大长度：**128**

响应参数

无

请求示例

修改勒索病毒防护策略，目标服务器操作系统类型为Linux，防护策略ID为0253edfd-30e7-439d-8f3f-17c54c997064，防护动作为仅告警。

```
PUT https://{endpoint}/v5/{project_id}/ransomware/protection/policy

{
  "bait_protection_status" : "opened",
  "protection_type" : "docx",
  "exclude_directory" : "",
```

```
  "operating_system" : "Linux",
  "policy_id" : "0253edfd-30e7-439d-8f3f-17c54c997064",
  "policy_name" : "aaa",
  "protection_mode" : "alarm_only",
  "protection_directory" : "/root",
  "runtime_detection_status" : "closed",
  "agent_id_list" : [ "" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改勒索病毒防护策略，目标服务器操作系统类型为Linux，防护策略ID为0253edfd-30e7-439d-8f3f-17c54c997064，防护动作为仅告警。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class UpdateProtectionPolicySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateProtectionPolicyRequest request = new UpdateProtectionPolicyRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        UpdateProtectionPolicyInfoRequestInfo body = new UpdateProtectionPolicyInfoRequestInfo();
        List<String> listbodyAgentIdList = new ArrayList<>();
        listbodyAgentIdList.add("");
        body.withRuntimeDetectionStatus("closed");
        body.withOperatingSystem("Linux");
        body.withAgentIdList(listbodyAgentIdList);
        body.withExcludeDirectory("");
        body.withProtectionType("docx");
        body.withProtectionDirectory("/root");
```

```
        body.withBaitProtectionStatus("opened");
        body.withProtectionMode("alarm_only");
        body.withPolicyName("aaa");
        body.withPolicyId("0253edfd-30e7-439d-8f3f-17c54c997064");
        request.withBody(body);
        try {
            UpdateProtectionPolicyResponse response = client.updateProtectionPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改勒索病毒防护策略，目标服务器操作系统类型为Linux，防护策略ID为0253edfd-30e7-439d-8f3f-17c54c997064，防护动作为仅告警。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = UpdateProtectionPolicyRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listAgentIdListbody = [
            ""
        ]
        request.body = UpdateProtectionPolicyInfoRequestInfo(
            runtime_detection_status="closed",
            operating_system="Linux",
            agent_id_list=listAgentIdListbody,
            exclude_directory="",
            protection_type="docx",
            protection_directory="/root",
            bait_protection_status="opened",
            protection_mode="alarm_only",
            policy_name="aaa",
            policy_id="0253edfd-30e7-439d-8f3f-17c54c997064"
        )
```

```
        response = client.update_protection_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

修改勒索病毒防护策略，目标服务器操作系统类型为Linux，防护策略ID为0253edfd-30e7-439d-8f3f-17c54c997064，防护动作为仅告警。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateProtectionPolicyRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listAgentIdListbody = []string{
        "",
    }
    runtimeDetectionStatusUpdateProtectionPolicyInfoRequestInfo:= "closed"
    excludeDirectoryUpdateProtectionPolicyInfoRequestInfo:= ""
    baitProtectionStatusUpdateProtectionPolicyInfoRequestInfo:= "opened"
    request.Body = &model.UpdateProtectionPolicyInfoRequestInfo{
        RuntimeDetectionStatus: &runtimeDetectionStatusUpdateProtectionPolicyInfoRequestInfo,
        OperatingSystem: "Linux",
        AgentIdList: &listAgentIdListbody,
        ExcludeDirectory: &excludeDirectoryUpdateProtectionPolicyInfoRequestInfo,
        ProtectionType: "docx",
        ProtectionDirectory: "/root",
        BaitProtectionStatus: &baitProtectionStatusUpdateProtectionPolicyInfoRequestInfo,
        ProtectionMode: "alarm_only",
        PolicyName: "aaa",
        PolicyId: "0253edfd-30e7-439d-8f3f-17c54c997064",
    }
    response, err := client.UpdateProtectionPolicy(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	success

错误码

请参见**错误码**。

3.2.4 开启勒索病毒防护

功能介绍

开启勒索病毒防护,请保证该region有cbr云备份服务，勒索服务与云备份服务有关联关系

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/ransomware/protection/open

表 3-102 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-103 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-104 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-105 请求 Body 参数

参数	是否必选	参数类型	描述
operating_system	是	String	需要开启防护的主机的操作系统，包含如下： ● Windows : Windows系统 ● Linux : Linux系统 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
ransom_protection_status	是	String	勒索防护是否开启，包含如下： • closed： 关闭。 • opened： 开启。 若选择开启，protection_policy_id或者create_protection_policy必填一项 最小长度：**0** 最大长度：**64**
protection_policy_id	否	String	勒索防护策略ID,若选择已有策略防护,则该字段必选 最小长度：**0** 最大长度：**64**
create_protection_policy	否	**ProtectionProxyInfoRequestInfo** object	创建防护策略。若新建防护策略，则protection_policy_id为空，create_protection_policy必选
backup_protection_status	是	String	是否服务器备份，包含如下： • closed： 关闭。 • opened： 开启。 若选择开启服务器备份，则backup_cycle必填 最小长度：**0** 最大长度：**64**
backup_resources	否	**BackupResources** object	开启备份功能新版参数，必填；若为空代表兼容之前绑定HSS_projectid的存储库
backup_policy_id	否	String	备份策略ID 最小长度：**0** 最大长度：**64**
backup_cycle	否	**UpdateBackupPolicyRequestInfo1** object	备份策略
agent_id_list	是	Array of strings	开启防护的Agent id列表 最小长度：**0** 最大长度：**64** 数组长度：**0 - 24**

参数	是否必选	参数类型	描述
host_id_list	是	Array of strings	开启防护的host id列表 最小长度：**0** 最大长度：**64** 数组长度：**0 - 24**

表 3-106 ProtectionProxyInfoRequestInfo

参数	是否必选	参数类型	描述
policy_id	否	String	策略ID，新建策略可不填 最小长度：**0** 最大长度：**64**
policy_name	否	String	策略名称，新建防护策略则必填 最小长度：**0** 最大长度：**64**
protection_mode	否	String	防护动作，新建防护策略则必填。包含如下： ● alarm_and_isolation ： 告警并自动隔离。 ● alarm_only ： 仅告警。 最小长度：**0** 最大长度：**64**
bait_protection_status	否	String	是否开启诱饵防护，新建防护策略则必填。包含如下1种, 默认为开启防护诱饵防护。 ● opened ： 开启。 ● closed ： 关闭。 最小长度：**0** 最大长度：**64**
protection_directory	否	String	防护目录，新建防护策略则必填 最小长度：**0** 最大长度：**64**
protection_type	否	String	防护类型，新建防护策略则必填 最小长度：**0** 最大长度：**64**
exclude_directory	否	String	排除目录，可选填 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
runtime_detection_status	否	String	是否运行时检测，选填。包含如下2种，暂时只有关闭一种状态，为保留字段。 • opened： 开启。 • closed： 关闭。 最小长度：**0** 最大长度：**64**
operating_system	否	String	操作系统，新建防护策略则必填。包含如下： • Windows : Windows系统 • Linux : Linux系统 最小长度：**0** 最大长度：**64**
process_whitelist	否	Array of **TrustProcessInfo** objects	进程白名单 数组长度：**0 - 20**

表 3-107 TrustProcessInfo

参数	是否必选	参数类型	描述
path	否	String	进程路径 最小长度：**0** 最大长度：**128**
hash	否	String	进程hash 最小长度：**0** 最大长度：**128**

表 3-108 BackupResources

参数	是否必选	参数类型	描述
vault_id	否	String	选择需要绑定的存储库ID，不为空 最小长度：**0** 最大长度：**64**
resource_list	否	Array of **ResourceInfo** objects	需要开启备份功能的主机情况列表 数组长度：**0 - 20**

表 3-109 ResourceInfo

参数	是否必选	参数类型	描述
host_id	否	String	主机id 最小长度：**0** 最大长度：**128**
history_backup_status	否	String	历史开启备份状态，通过筛选可用服务器的error_message或者status判断，如果error_message为空，则没有开启备份，该字段为closed；若不为空，则为opened 最小长度：**0** 最大长度：**128**

表 3-110 UpdateBackupPolicyRequestInfo1

参数	是否必选	参数类型	描述
enabled	否	Boolean	策略是否启用，缺省值：true
policy_id	否	String	策略ID,若开启防护时开启备份防护，该字段必选 最小长度：**1** 最大长度：**256**
operation_definition	否	**OperationDefinitionRequestInfo** object	调度参数
trigger	否	**BackupTriggerRequestInfo1** object	策略时间调度规则

表 3-111 OperationDefinitionRequestInfo

参数	是否必选	参数类型	描述
day_backups	否	Integer	保留日备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0,最大值：100 最小值：**0** 最大值：**100**

参数	是否必选	参数类型	描述
max_backups	否	Integer	单个备份对象自动备份的最大备份数。取值为-1或0-99999。-1代表不按备份数清理。若该字段和retention_duration_days字段同时为空，备份会永久保留。最小值：1,最大值：99999,缺省值：-1 最小值：**-1** 最大值：**99999**
month_backups	否	Integer	保留月备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0,最大值：100 最小值：**0** 最大值：**100**
retention_duration_days	否	Integer	备份保留时长，单位天。最长支持99999天。-1代表不按时间清理。若该字段和max_backups参数同时为空，备份会永久保留。最小值：1, 最大值：99999, 缺省值：-1 最小值：**-1** 最大值：**99999**
timezone	否	String	用户所在时区,格式形如UTC+08:00,若没有选择年备，月备，周备，日备中任一参数，则不能选择该参数。 最小长度：**0** 最大长度：**256**
week_backups	否	Integer	保留周备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。 最小值：**0** 最大值：**100**
year_backups	否	Integer	保留年备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0，最大值：100 最小值：**0** 最大值：**100**

表 3-112 BackupTriggerRequestInfo1

参数	是否必选	参数类型	描述
properties	否	**BackupTriggerPropertiesRequestInfo1** object	策略执行时间规则，若开启勒索防护时开启备份功能，则该字段必选

表 3-113 BackupTriggerPropertiesRequestInfo1

参数	是否必选	参数类型	描述
pattern	否	Array of strings	调度规则。若开启勒索防护时开启备份功能，则该字段必选。限制24条规则。调度器的调度规则，可参照iCalendar RFC 2445规范中的事件规则，但仅支持FREQ、BYDAY、BYHOUR、BYMINUTE、INTERVAL等参数，其中FREQ仅支持WEEKLY和DAILY，BYDAY支持一周七天（MO、TU、WE、TH、FR、SA、SU），BYHOUR支持0-23小时，BYMINUTE支持0-59分钟，并且间隔不能小于一小时，一天最大24个时间点。例如，周一到周天，每天14:00调度，其规则为：'FREQ=WEEKLY;BYDAY=MO,TU,WE,TH,FR,SA,SU;BYHOUR=14;BYMINUTE=00'。每天14:00调度，其规则为'FREQ=DAILY;INTERVAL=1;BYHOUR=14;BYMINUTE=00'。 最小长度：**1** 最大长度：**256** 数组长度：**0 - 24**

响应参数

无

请求示例

开启服务器勒索病毒防护，目标服务器操作系统类型为Linux，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为

c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8，不开启服务器备份。

```
POST https://{endpoint}/v5/{project_id}/ransomware/protection/open

{
  "ransom_protection_status" : "opened",
  "backup_protection_status" : "closed",
  "operating_system" : "Linux",
  "protection_policy_id" : "",
  "agent_id_list" : [ "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8" ],
  "host_id_list" : [ "71a15ecc-049f-4cca-bd28-5e90aca1817f" ],
  "create_protection_policy" : {
    "bait_protection_status" : "opened",
    "exclude_directory" : "",
    "protection_mode" : "alarm_only",
    "policy_name" : "test111",
    "protection_directory" : "/etc/test",
    "protection_type" : "docx"
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启服务器勒索病毒防护，目标服务器操作系统类型为Linux，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8，不开启服务器备份。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class StartProtectionSolution {

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

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

```
        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        StartProtectionRequest request = new StartProtectionRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ProtectionInfoRequestInfo body = new ProtectionInfoRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("71a15ecc-049f-4cca-bd28-5e90aca1817f");
        List<String> listbodyAgentIdList = new ArrayList<>();
        listbodyAgentIdList.add("c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8");
        ProtectionProxyInfoRequestInfo createProtectionPolicybody = new ProtectionProxyInfoRequestInfo();
        createProtectionPolicybody.withPolicyName("test111")
            .withProtectionMode("alarm_only")
            .withBaitProtectionStatus("opened")
            .withProtectionDirectory("/etc/test")
            .withProtectionType("docx")
            .withExcludeDirectory("");
        body.withHostIdList(listbodyHostIdList);
        body.withAgentIdList(listbodyAgentIdList);
        body.withBackupProtectionStatus("closed");
        body.withCreateProtectionPolicy(createProtectionPolicybody);
        body.withProtectionPolicyId("");
        body.withRansomProtectionStatus("opened");
        body.withOperatingSystem("Linux");
        request.withBody(body);
        try {
            StartProtectionResponse response = client.startProtection(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启服务器勒索病毒防护，目标服务器操作系统类型为Linux，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8，不开启服务器备份。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    credentials = BasicCredentials(ak, sk)

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

    try:
        request = StartProtectionRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListbody = [
            "71a15ecc-049f-4cca-bd28-5e90aca1817f"
        ]
        listAgentIdListbody = [
            "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8"
        ]
        createProtectionPolicybody = ProtectionProxyInfoRequestInfo(
            policy_name="test111",
            protection_mode="alarm_only",
            bait_protection_status="opened",
            protection_directory="/etc/test",
            protection_type="docx",
            exclude_directory=""
        )
        request.body = ProtectionInfoRequestInfo(
            host_id_list=listHostIdListbody,
            agent_id_list=listAgentIdListbody,
            backup_protection_status="closed",
            create_protection_policy=createProtectionPolicybody,
            protection_policy_id="",
            ransom_protection_status="opened",
            operating_system="Linux"
        )
        response = client.start_protection(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启服务器勒索病毒防护，目标服务器操作系统类型为Linux，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8，不开启服务器备份。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

    auth := basic.NewCredentialsBuilder().
```

```
        WithAk(ak).
        WithSk(sk).
        Build()

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartProtectionRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdListbody = []string{
        "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    }
    var listAgentIdListbody = []string{
        "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8",
    }
    policyNameCreateProtectionPolicy:= "test111"
    protectionModeCreateProtectionPolicy:= "alarm_only"
    baitProtectionStatusCreateProtectionPolicy:= "opened"
    protectionDirectoryCreateProtectionPolicy:= "/etc/test"
    protectionTypeCreateProtectionPolicy:= "docx"
    excludeDirectoryCreateProtectionPolicy:= ""
    createProtectionPolicybody := &model.ProtectionProxyInfoRequestInfo{
        PolicyName: &policyNameCreateProtectionPolicy,
        ProtectionMode: &protectionModeCreateProtectionPolicy,
        BaitProtectionStatus: &baitProtectionStatusCreateProtectionPolicy,
        ProtectionDirectory: &protectionDirectoryCreateProtectionPolicy,
        ProtectionType: &protectionTypeCreateProtectionPolicy,
        ExcludeDirectory: &excludeDirectoryCreateProtectionPolicy,
    }
    protectionPolicyIdProtectionInfoRequestInfo:= ""
    request.Body = &model.ProtectionInfoRequestInfo{
        HostIdList: listHostIdListbody,
        AgentIdList: listAgentIdListbody,
        BackupProtectionStatus: "closed",
        CreateProtectionPolicy: createProtectionPolicybody,
        ProtectionPolicyId: &protectionPolicyIdProtectionInfoRequestInfo,
        RansomProtectionStatus: "opened",
        OperatingSystem: "Linux",
    }
    response, err := client.StartProtection(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	开启勒索病毒防护成功

错误码

请参见**错误码**。

3.2.5 关闭勒索病毒防护

功能介绍

关闭勒索病毒防护

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/ransomware/protection/close

表 3-114 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-115 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-116 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-117 请求 Body 参数

参数	是否必选	参数类型	描述
host_id_list	是	Array of strings	需要关闭勒索防护的主机ID列表 最小长度：**0** 最大长度：**64** 数组长度：**0 - 20**
agent_id_list	是	Array of strings	需要关闭勒索防护的agentID列表 最小长度：**0** 最大长度：**64** 数组长度：**0 - 20**
close_protection_type	是	String	关闭防护类型，包含如下： • close_anti : 关闭勒索防护；暂不支持关闭备份防护，若需要解绑存储库，请前往cbr服务进行操作。 最小长度：**0** 最大长度：**64**

响应参数

无

请求示例

关闭服务器勒索病毒防护，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8。

```
POST https://{endpoint}/v5/{project_id}/ransomware/protection/close

{
  "close_protection_type" : "close_anti",
  "host_id_list" : [ "71a15ecc-049f-4cca-bd28-5e90aca1817f" ],
  "agent_id_list" : [ "c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

关闭服务器勒索病毒防护，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class StopProtectionSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        StopProtectionRequest request = new StopProtectionRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        CloseProtectionInfoRequestInfo body = new CloseProtectionInfoRequestInfo();
```

```
        List<String> listbodyAgentIdList = new ArrayList<>();
        listbodyAgentIdList.add("c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8");
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("71a15ecc-049f-4cca-bd28-5e90aca1817f");
        body.withCloseProtectionType("close_anti");
        body.withAgentIdList(listbodyAgentIdList);
        body.withHostIdList(listbodyHostIdList);
        request.withBody(body);
        try {
            StopProtectionResponse response = client.stopProtection(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

关闭服务器勒索病毒防护，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = StopProtectionRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listAgentIdListbody = [
            "c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8"
        ]
        listHostIdListbody = [
            "71a15ecc-049f-4cca-bd28-5e90aca1817f"
        ]
        request.body = CloseProtectionInfoRequestInfo(
            close_protection_type="close_anti",
            agent_id_list=listAgentIdListbody,
            host_id_list=listHostIdListbody
```

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

Go

关闭服务器勒索病毒防护，目标服务器ID为71a15ecc-049f-4cca-bd28-5e90aca1817f，目标服务器的Agent ID为c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopProtectionRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listAgentIdListbody = []string{
        "c9bed5397db449ebdfba15e85fcfc36accee954daf5cab0528bab59bd8",
    }
    var listHostIdListbody = []string{
        "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    }
    request.Body = &model.CloseProtectionInfoRequestInfo{
        CloseProtectionType: "close_anti",
        AgentIdList: listAgentIdListbody,
        HostIdList: listHostIdListbody,
    }
    response, err := client.StopProtection(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	关闭勒索病毒防护成功

错误码

请参见**错误码**。

3.2.6 查询 HSS 存储库绑定的备份策略信息

功能介绍

查询HSS存储库绑定的备份策略信息,确保已经购买了勒索防护存储库，可以从cbr云备份服务进行验证，确保已经存在HSS_projectid命名的存储库已经购买

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/backup/policy

表 3-118 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-119 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-120 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-121 响应 Body 参数

参数	参数类型	描述
enabled	Boolean	策略是否启用
id	String	策略ID 最小长度：**1** 最大长度：**128**
name	String	策略名称 最小长度：**1** 最大长度：**128**
operation_type	String	备份类型。备份（backup）、复制(replication)，包含如下2种。 • backup： 备份。 • replication： 复制。 最小长度：**1** 最大长度：**128**
operation_definition	**OperationDefinitionInfo** object	策略属性 保留规则
trigger	**BackupTriggerInfo** object	备份:策略时间调度规则

表 3-122 OperationDefinitionInfo

参数	参数类型	描述
day_backups	Integer	保留日备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone也必选。最小值：0,最大值：100 最小值：**0** 最大值：**100**
max_backups	Integer	单个备份对象自动备份的最大备份数。取值为-1或0-99999。-1代表不按备份数清理。若该字段和retention_duration_days字段同时为空，备份会永久保留。最小值：1,最大值：99999,缺省值：-1 最小值：**-1** 最大值：**99999**
month_backups	Integer	保留月备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone也必选。最小值：0, 最大值：100 最小值：**0** 最大值：**100**
retention_duration_days	Integer	备份保留时长，单位天。最长支持99999天。-1代表不按时间清理。若该字段和max_backups 参数同时为空，备份会永久保留。最小值：1, 最大值：99999, 缺省值：-1 最小值：**-1** 最大值：**99999**
timezone	String	用户所在时区,格式形如UTC+08:00,若没有选择年备，月备，周备，日备中任一参数，则不能选择该参数。 最小长度：**0** 最大长度：**256**
week_backups	Integer	保留周备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone也必选。 最小值：**0** 最大值：**100**
year_backups	Integer	保留年备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone也必选。最小值：0，最大值：100 最小值：**0** 最大值：**100**

表 3-123 BackupTriggerInfo

参数	参数类型	描述
id	String	调度器id 最小长度：**0** 最大长度：**256**
name	String	调度器名称 最小长度：**0** 最大长度：**256**
type	String	调度器类型,目前只支持 time,定时调度。 最小长度：**0** 最大长度：**256**
properties	**BackupTriggerPropertiesInfo** object	调度器属性

表 3-124 BackupTriggerPropertiesInfo

参数	参数类型	描述
pattern	Array of strings	调度器的调度策略，长度限制为10240个字符，参照iCalendar RFC 2445规范，但仅支持FREQ、BYDAY、BYHOUR、BYMINUTE四个参数，其中FREQ仅支持WEEKLY和DAILY，BYDAY支持一周七天（MO、TU、WE、TH、FR、SA、SU），BYHOUR支持0-23小时，BYMINUTE支持0-59分钟，并且时间点间隔不能小于一小时，一个备份策略可以同时设置多个备份时间点，一天最多可以设置24个时间点。 最小长度：**0** 最大长度：**256** 数组长度：**0 - 24**
start_time	String	调度器开始时间，例如：2020-01-08 09:59:49 最小长度：**0** 最大长度：**256**

请求示例

查询HSS存储库绑定的备份策略信息。

```
GET https://{endpoint}/v5/{project_id}/backup/policy
```

响应示例

状态码： 200

备份策略信息

```
{
  "enabled" : true,
  "id" : "af4d08ad-2b60-4916-a5cf-8d6a23956dda",
  "name" : "HSS_84b5266c14ae489fa6549827f032dc62",
  "operation_type" : "backup",
  "operation_definition" : {
    "day_backups" : 0,
    "max_backups" : "-1",
    "month_backups" : 0,
    "retention_duration_days" : 5,
    "timezone" : "UTC+08:00",
    "week_backups" : 0,
    "year_backups" : 0
  },
  "trigger" : {
    "properties" : {
      "pattern" : [ "FREQ=DAILY;INTERVAL=2;BYHOUR=14;BYMINUTE=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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowBackupPolicyInfoSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowBackupPolicyInfoRequest request = new ShowBackupPolicyInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        try {
```

```
            ShowBackupPolicyInfoResponse response = client.showBackupPolicyInfo(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowBackupPolicyInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        response = client.show_backup_policy_info(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowBackupPolicyInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    response, err := client.ShowBackupPolicyInfo(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	备份策略信息

错误码

请参见**错误码**。

3.2.7 修改存储库绑定的备份策略

功能介绍

修改存储库绑定的备份策略

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/backup/policy

表 3-125 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-126 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-127 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-128 请求 Body 参数

参数	是否必选	参数类型	描述
enabled	否	Boolean	策略是否启用，缺省值：true
policy_id	是	String	备份策略ID 最小长度：**1** 最大长度：**256**

参数	是否必选	参数类型	描述
operation_definition	否	**OperationDefinitionRequestInfo** object	调度参数
trigger	否	**BackupTriggerRequestInfo** object	策略时间调度规则

表 3-129 OperationDefinitionRequestInfo

参数	是否必选	参数类型	描述
day_backups	否	Integer	保留日备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0, 最大值：100 最小值：**0** 最大值：**100**
max_backups	否	Integer	单个备份对象自动备份的最大备份数。取值为-1或0-99999。-1代表不按备份数清理。若该字段和retention_duration_days字段同时为空，备份会永久保留。最小值：1,最大值：99999,缺省值：-1 最小值：**-1** 最大值：**99999**
month_backups	否	Integer	保留月备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0, 最大值：100 最小值：**0** 最大值：**100**
retention_duration_days	否	Integer	备份保留时长，单位天。最长支持99999天。-1代表不按时间清理。若该字段和max_backups参数同时为空，备份会永久保留。最小值：1, 最大值：99999, 缺省值：-1 最小值：**-1** 最大值：**99999**

参数	是否必选	参数类型	描述
timezone	否	String	用户所在时区,格式形如UTC +08:00,若没有选择年备，月备，周备，日备中任一参数，则不能选择该参数。 最小长度：**0** 最大长度：**256**
week_backups	否	Integer	保留周备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。 最小值：**0** 最大值：**100**
year_backups	否	Integer	保留年备个数，该备份不受保留最大备份数限制。取值为0到100。若选择该参数，则timezone 也必选。最小值：0，最大值：100 最小值：**0** 最大值：**100**

表 3-130 BackupTriggerRequestInfo

参数	是否必选	参数类型	描述
properties	是	**BackupTriggerPropertiesRequestInfo** object	策略执行时间规则

表 3-131 BackupTriggerPropertiesRequestInfo

参数	是否必选	参数类型	描述
pattern	是	Array of strings	调度规则。限制24条规则。调度器的调度规则，可参照iCalendar RFC 2445规范中的事件规则，但仅支持FREQ、BYDAY、BYHOUR、BYMINUTE、INTERVAL等参数，其中FREQ仅支持WEEKLY和DAILY，BYDAY支持一周七天（MO、TU、WE、TH、FR、SA、SU），BYHOUR支持0-23小时，BYMINUTE支持0-59分钟，并且间隔不能小于一小时，一天最大24个时间点。例如，周一到周天，每天14:00调度，其规则为：'FREQ=WEEKLY;BYDAY=MO,TU,WE,TH,FR,SA,SU;BYHOUR=14;BYMINUTE=00'。每天14:00调度，其规则为'FREQ=DAILY;INTERVAL=1;BYHOUR=14;BYMINUTE=00'。 最小长度：**1** 最大长度：**256** 数组长度：**0 - 24**

响应参数

无

请求示例

修改备份策略，目标备份策略ID为af4d08ad-2b60-4916-a5cf-8d6a23956dda。

```
PUT https://{endpoint}/v5/{project_id}/backup/policy

{
  "enabled" : true,
  "policy_id" : "af4d08ad-2b60-4916-a5cf-8d6a23956dda",
  "operation_definition" : {
    "day_backups" : 0,
    "max_backups" : -1,
    "month_backups" : 0,
    "retention_duration_days" : 5,
    "timezone" : "UTC+08:00",
    "week_backups" : 0,
    "year_backups" : 0
  },
  "trigger" : {
    "properties" : {
      "pattern" : [ "FREQ=DAILY;INTERVAL=2;BYHOUR=14;BYMINUTE=00" ]
    }
```

```
 }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改备份策略，目标备份策略ID为af4d08ad-2b60-4916-a5cf-8d6a23956dda。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class UpdateBackupPolicyInfoSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateBackupPolicyInfoRequest request = new UpdateBackupPolicyInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        UpdateBackupPolicyRequestInfo body = new UpdateBackupPolicyRequestInfo();
        List<String> listPropertiesPattern = new ArrayList<>();
        listPropertiesPattern.add("FREQ=DAILY;INTERVAL=2;BYHOUR=14;BYMINUTE=00");
        BackupTriggerPropertiesRequestInfo propertiesTrigger = new BackupTriggerPropertiesRequestInfo();
        propertiesTrigger.withPattern(listPropertiesPattern);
        BackupTriggerRequestInfo triggerbody = new BackupTriggerRequestInfo();
        triggerbody.withProperties(propertiesTrigger);
        OperationDefinitionRequestInfo operationDefinitionbody = new OperationDefinitionRequestInfo();
        operationDefinitionbody.withDayBackups(0)
            .withMaxBackups(-1)
            .withMonthBackups(0)
            .withRetentionDurationDays(5)
            .withTimezone("UTC+08:00")
            .withWeekBackups(0)
            .withYearBackups(0);
        body.withTrigger(triggerbody);
```

```
        body.withOperationDefinition(operationDefinitionbody);
        body.withPolicyId("af4d08ad-2b60-4916-a5cf-8d6a23956dda");
        body.withEnabled(true);
        request.withBody(body);
        try {
            UpdateBackupPolicyInfoResponse response = client.updateBackupPolicyInfo(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改备份策略，目标备份策略ID为af4d08ad-2b60-4916-a5cf-8d6a23956dda。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = UpdateBackupPolicyInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listPatternProperties = [
            "FREQ=DAILY;INTERVAL=2;BYHOUR=14;BYMINUTE=00"
        ]
        propertiesTrigger = BackupTriggerPropertiesRequestInfo(
            pattern=listPatternProperties
        )
        triggerbody = BackupTriggerRequestInfo(
            properties=propertiesTrigger
        )
        operationDefinitionbody = OperationDefinitionRequestInfo(
            day_backups=0,
            max_backups=-1,
            month_backups=0,
            retention_duration_days=5,
            timezone="UTC+08:00",
            week_backups=0,
            year_backups=0
```

```
        )
        request.body = UpdateBackupPolicyRequestInfo(
            trigger=triggerbody,
            operation_definition=operationDefinitionbody,
            policy_id="af4d08ad-2b60-4916-a5cf-8d6a23956dda",
            enabled=True
        )
        response = client.update_backup_policy_info(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改备份策略，目标备份策略ID为af4d08ad-2b60-4916-a5cf-8d6a23956dda。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateBackupPolicyInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listPatternProperties = []string{
        "FREQ=DAILY;INTERVAL=2;BYHOUR=14;BYMINUTE=00",
    }
    propertiesTrigger := &model.BackupTriggerPropertiesRequestInfo{
        Pattern: listPatternProperties,
    }
    triggerbody := &model.BackupTriggerRequestInfo{
        Properties: propertiesTrigger,
    }
    dayBackupsOperationDefinition:= int32(0)
    maxBackupsOperationDefinition:= int32(-1)
    monthBackupsOperationDefinition:= int32(0)
    retentionDurationDaysOperationDefinition:= int32(5)
    timezoneOperationDefinition:= "UTC+08:00"
    weekBackupsOperationDefinition:= int32(0)
    yearBackupsOperationDefinition:= int32(0)
    operationDefinitionbody := &model.OperationDefinitionRequestInfo{
```

```
        DayBackups: &dayBackupsOperationDefinition,
        MaxBackups: &maxBackupsOperationDefinition,
        MonthBackups: &monthBackupsOperationDefinition,
        RetentionDurationDays: &retentionDurationDaysOperationDefinition,
        Timezone: &timezoneOperationDefinition,
        WeekBackups: &weekBackupsOperationDefinition,
        YearBackups: &yearBackupsOperationDefinition,
    }
    enabledUpdateBackupPolicyRequestInfo:= true
    request.Body = &model.UpdateBackupPolicyRequestInfo{
        Trigger: triggerbody,
        OperationDefinition: operationDefinitionbody,
        PolicyId: "af4d08ad-2b60-4916-a5cf-8d6a23956dda",
        Enabled: &enabledUpdateBackupPolicyRequestInfo,
    }
    response, err := client.UpdateBackupPolicyInfo(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	修改备份策略

错误码

请参见**错误码**。

3.3 基线管理

3.3.1 查询弱口令检测结果列表

功能介绍

查询弱口令检测结果列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/weak-password-users

表 3-132 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-133 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**256**
host_ip	否	String	服务器IP地址 最小长度：**0** 最大长度：**256**
user_name	否	String	弱口令账号名称 最小长度：**0** 最大长度：**32**
host_id	否	String	主机ID，不赋值时，查租户所有主机 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页数量 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-134 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-135 响应 Body 参数

参数	参数类型	描述
total_num	Long	弱口令总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **WeakPwdListInfoResponseInfo** objects	弱口令列表 数组长度：**0 - 2147483647**

表 3-136 WeakPwdListInfoResponseInfo

参数	参数类型	描述
host_id	String	主机ID 最小长度：**0** 最大长度：**64**
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**
host_ip	String	服务器IP（私有IP），为兼容用户使用，不删除此字段 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
private_ip	String	服务器私有IP 最小长度：**0** 最大长度：**256**
public_ip	String	服务器公网IP 最小长度：**0** 最大长度：**256**
weak_pwd_accounts	Array of **WeakPwdAccountInfoResponseInfo** objects	弱口令账号列表 数组长度：**0 - 2147483647**

表 3-137 WeakPwdAccountInfoResponseInfo

参数	参数类型	描述
user_name	String	弱口令账号名称 最小长度：**0** 最大长度：**32**
service_type	String	账号类型，包含如下: • system • mysql • redis 最小长度：**0** 最大长度：**32**
duration	Integer	弱口令使用时长，单位天 最小值：**0** 最大值：**2147483647**

请求示例

查询企业项目id为xxx下的主机弱口令检测结果。默认返回第一页（前10条）数据。

```
GET https://{endpoint}/v5/{project_id}/baseline/weak-password-users?enterprise_project_id=xxx
```

响应示例

状态码： 200

弱口令检测结果列表

```
{
  "total_num" : 2,
```

```
 "data_list" : [ {
   "host_id" : "caa958adxxxxxxa481",
   "host_name" : "ubuntu1",
   "host_ip" : "192.168.0.8",
   "private_ip" : "192.168.0.8",
   "public_ip" : "100.85.85.85",
   "weak_pwd_accounts" : [ {
     "user_name" : "localhost1",
     "service_type" : "system",
     "duration" : 2147483647
   } ]
 }, {
   "host_id" : "caa958adxxxxxxa482",
   "host_name" : "ubuntu2",
   "host_ip" : "192.168.0.9",
   "private_ip" : "192.168.0.8",
   "public_ip" : "",
   "weak_pwd_accounts" : [ {
     "user_name" : "localhost2",
     "service_type" : "system",
     "duration" : 2147483647
   } ]
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListWeakPasswordUsersSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListWeakPasswordUsersRequest request = new ListWeakPasswordUsersRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withUserName("<user_name>");
        request.withHostId("<host_id>");
```

```
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListWeakPasswordUsersResponse response = client.listWeakPasswordUsers(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListWeakPasswordUsersRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.user_name = "<user_name>"
        request.host_id = "<host_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_weak_password_users(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"
```

```
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListWeakPasswordUsersRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    userNameRequest:= "<user_name>"
    request.UserName = &userNameRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListWeakPasswordUsers(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	弱口令检测结果列表

错误码

请参见**错误码**。

3.3.2 查询口令复杂度策略检测报告

功能介绍

查询口令复杂度策略检测报告

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/password-complexity

表 3-138 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-139 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**256**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**128**
host_ip	否	String	服务器IP地址 最小长度：**0** 最大长度：**128**
host_id	否	String	主机id，不赋值时，查租户所有主机 最小长度：**0** 最大长度：**128**
limit	否	Integer	每页显示数量 最小值：**0** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-140 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-141 响应 Body 参数

参数	参数类型	描述
total_num	Long	记录总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **PwdPolicyInfoResponseInfo** objects	口令复杂度策略检测列表 数组长度：**0 - 2147483647**

表 3-142 PwdPolicyInfoResponseInfo

参数	参数类型	描述
host_id	String	主机id 最小长度：**0** 最大长度：**64**
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**
host_ip	String	服务器IP（私有IP），为兼容用户使用，不删除此字段 最小长度：**0** 最大长度：**256**
private_ip	String	服务器私有IP 最小长度：**0** 最大长度：**256**
public_ip	String	服务器公网IP 最小长度：**0** 最大长度：**256**
min_length	Boolean	口令最小长度的设置是否符合要求，符合为true，不符合为false
uppercase_letter	Boolean	大写字母的设置是否符合要求，符合为true，不符合为false
lowercase_letter	Boolean	小写字母的设置是否符合要求，符合为true，不符合为false
number	Boolean	数字的设置是否符合要求，符合为true，不符合为false
special_character	Boolean	特殊字符的设置是否符合要求，符合为true，不符合为false
suggestion	String	修改建议 最小长度：**0** 最大长度：**65534**

请求示例

查询企业项目id为xxx下的主机口令复杂度检测结果。默认返回第一页（前10条）数据。

```
GET https://{endpoint}/v5/{project_id}/baseline/password-complexity?enterprise_project_id=xxx
```

响应示例

状态码： 200

口令复杂度策略检测报告

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_id" : "76fa440a-5a08-43fa-ac11-d12183ab3a14",
    "host_ip" : "192.168.0.59",
    "private_ip" : "192.168.0.8",
    "public_ip" : "100.85.85.85",
    "host_name" : "ecs-6b96",
    "lowercase_letter" : false,
    "min_length" : true,
    "number" : false,
    "special_character" : false,
    "suggestion" : "The password should contain at least 3 of the following character types: uppercase letters, lowercase letters, digits, and special characters. ",
    "uppercase_letter" : 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;
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListPasswordComplexitySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPasswordComplexityRequest request = new ListPasswordComplexityRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withHostId("<host_id>");
        request.withLimit(<limit>);
```

```
        request.withOffset(<offset>);
        try {
            ListPasswordComplexityResponse response = client.listPasswordComplexity(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListPasswordComplexityRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.host_id = "<host_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_password_complexity(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPasswordComplexityRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListPasswordComplexity(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	口令复杂度策略检测报告

错误码

请参见**错误码**。

3.3.3 查询租户的服务器安全配置检测结果列表

功能介绍

查询租户的服务器安全配置检测结果列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/risk-configs

表 3-143 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-144 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**256**
check_name	否	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**
group_id	否	String	策略组ID 最小长度：**0** 最大长度：**128**
severity	否	String	风险等级，包含如下: • Security：安全 • Low：低危 • Medium：中危 • High：高危 最小长度：**1** 最大长度：**32**

参数	是否必选	参数类型	描述
standard	否	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 云安全实践标准 最小长度：**1** 最大长度：**32**
host_id	否	String	主机id 最小长度：**0** 最大长度：**128**
limit	否	Integer	每页显示数量 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-145 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-146 响应 Body 参数

参数	参数类型	描述
total_num	Long	记录总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **SecurityCheckInfoResponseInfo** objects	服务器配置检测结果列表 数组长度：**0 - 2147483647**

表 3-147 SecurityCheckInfoResponseInfo

参数	参数类型	描述
severity	String	风险等级，包含如下: ● Low : 低危 ● Medium : 中危 ● High : 高危 最小长度：**1** 最大长度：**32**
check_name	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**
check_type	String	配置检查（基线）的类型,Linux系统支持的基线一般check_type和check_name相同,例如SSH、CentOS 7。 Windows系统支持的基线一般check_type和check_name不相同，例如check_name为Windows的配置检查（基线），它的check_type包含Windows Server 2019 R2、Windows Server 2016 R2等。 最小长度：**0** 最大长度：**256**
standard	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 云安全实践标准 最小长度：**1** 最大长度：**16**

参数	参数类型	描述
check_rule_num	Integer	当前配置检查（基线）类型下，用户共检测了多少个检查项。例如标准类型为hw_standard的SSH基线，主机安全提供了17个检查项，但用户所有主机都只检测了SSH基线的其中5个检查项，check_rule_num就是5。用户有一台主机进行了全量检查项检测，check_rule_num就是17。 最小值：**0** 最大值：**2097152**
failed_rule_num	Integer	未通过的检查项数量，check_rule_num中只要有一台主机没通过某个检查项，这个检查项就会被计算在failed_rule_num中 最小值：**0** 最大值：**2097152**
host_num	Integer	受影响的服务器的数量，进行了当前基线检测的服务器数量 最小值：**0** 最大值：**2097152**
scan_time	Long	最新检测时间(ms) 最小值：**0** 最大值：**2097152**
check_type_desc	String	对配置检查（基线）类型的描述信息，概括当前基线包含的检查项是根据什么标准制定的，能够审计哪些方面的问题。 最小长度：**0** 最大长度：**65534**

请求示例

查询企业项目id为xxx下的主机基线配置检测结果列表。默认返回第一页（前10条）数据。

```
GET https://{endpoint}/v5/{project_id}/baseline/risk-configs?enterprise_project_id=xxx
```

响应示例

状态码： 200

服务器安全配置检测结果列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "check_name" : "Docker",
    "check_rule_num" : 25,
    "check_type" : "Docker",
    "check_type_desc" : "Configuring security audit of Docker's host configurations and container-running-
```

```
related contents based on Docker Container Security Specifications V1_0.",
    "failed_rule_num" : 20,
    "host_num" : 0,
    "scan_time" : 1661716860935,
    "severity" : "High",
    "standard" : "hw_standard"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListRiskConfigsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListRiskConfigsRequest request = new ListRiskConfigsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withCheckName("<check_name>");
        request.withGroupId("<group_id>");
        request.withSeverity("<severity>");
        request.withStandard("<standard>");
        request.withHostId("<host_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListRiskConfigsResponse response = client.listRiskConfigs(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListRiskConfigsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.check_name = "<check_name>"
        request.group_id = "<group_id>"
        request.severity = "<severity>"
        request.standard = "<standard>"
        request.host_id = "<host_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_risk_configs(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListRiskConfigsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    checkNameRequest:= "<check_name>"
    request.CheckName = &checkNameRequest
    groupIdRequest:= "<group_id>"
    request.GroupId = &groupIdRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    standardRequest:= "<standard>"
    request.Standard = &standardRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListRiskConfigs(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	服务器安全配置检测结果列表

错误码

请参见**错误码**。

3.3.4 查询指定安全配置项的检查结果

功能介绍

查询指定安全配置项的检查结果

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/risk-config/{check_name}/detail

表 3-148 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**
check_name	是	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**

表 3-149 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
standard	是	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 云安全实践标准 最小长度：**0** 最大长度：**32**
host_id	否	String	主机ID，不赋值时，查租户所有主机 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页数量 最小值：**0** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-150 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-151 响应 Body 参数

参数	参数类型	描述
severity	String	风险等级，包含如下: ● Low : 低危 ● Medium : 中危 ● High : 高危 最小长度：**0** 最大长度：**65534**
check_type	String	配置检查（基线）的类型,例如SSH、CentOS 7、Windows Server 2019 R2、Windows Server 2016 R2、MySQL5-Windows 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
check_type_desc	String	对配置检查（基线）类型的描述信息，概括当前基线包含的检查项是根据什么标准制定的，能够审计哪些方面的问题。 最小长度：**0** 最大长度：**65534**
check_rule_num	Integer	当前配置检查（基线）类型下，用户共检测了多少个检查项。例如标准类型为hw_standard的SSH基线，主机安全提供了17个检查项，但用户所有主机都只检测了SSH基线的其中5个检查项，check_rule_num就是5。用户有一台主机进行了全量检查项检测，check_rule_num就是17。 最小值：**0** 最大值：**2147483647**
failed_rule_num	Integer	未通过的检查项数量，check_rule_num中只要有一台主机没通过某个检查项，这个检查项就会被计算在failed_rule_num中 最小值：**0** 最大值：**2147483647**
passed_rule_num	Integer	已通过的检查项数量，check_rule_num中只要有一台主机通过了某个检查项，这个检查项就会被计算在passed_rule_num中 最小值：**0** 最大值：**2147483647**
ignored_rule_num	Integer	已忽略的检查项数量，check_rule_num中只要有一台主机忽略了某个检查项，这个检查项就会被计算在ignored_rule_num中 最小值：**0** 最大值：**2147483647**
host_num	Long	受影响的服务器的数量，进行了当前基线检测的服务器数量 最小值：**0** 最大值：**2147483647**

请求示例

查询企业项目id为xxx下的基线名称为SSH、标准类型是“云安全实践”标准的配置检测结果列表。

```
GET https://{endpoint}/v5/{project_id}/baseline/risk-config/SSH/detail?
standard=hw_standard&enterprise_project_id=xxx
```

响应示例

状态码： 200

安全配置项的检查结果

```
{
  "check_rule_num" : 17,
  "check_type_desc" : "This policy checks the basic security configuration items of the SSH service to improve the security of the SSH service.",
  "failed_rule_num" : 15,
  "host_num" : 2,
  "ignored_rule_num" : 1,
  "passed_rule_num" : 14,
  "severity" : "Medium"
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowRiskConfigDetailSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowRiskConfigDetailRequest request = new ShowRiskConfigDetailRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withStandard("<standard>");
        request.withHostId("<host_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ShowRiskConfigDetailResponse response = client.showRiskConfigDetail(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowRiskConfigDetailRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.standard = "<standard>"
        request.host_id = "<host_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.show_risk_config_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowRiskConfigDetailRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.Standard = "<standard>"
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ShowRiskConfigDetail(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	安全配置项的检查结果

错误码

请参见**错误码**。

3.3.5 查询指定安全配置项的检查项列表

功能介绍

查询指定安全配置项的检查项列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/risk-config/{check_name}/check-rules

表 3-152 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**
check_name	是	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**

表 3-153 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
standard	是	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 云安全实践标准 最小长度：**0** 最大长度：**32**
result_type	否	String	结果类型，包含如下: ● safe： 已通过 ● unhandled : 未通过，且未忽略的 ● ignored : 未通过，且已忽略的 缺省值：**unhandled** 最小长度：**0** 最大长度：**64**
check_rule_name	否	String	检查项（检查规则）名称，支持模糊匹配 最小长度：**0** 最大长度：**2048**

参数	是否必选	参数类型	描述
severity	否	String	风险等级，包含如下: • Security : 安全 • Low : 低危 • Medium : 中危 • High : 高危 • Critical : 危急 最小长度：**0** 最大长度：**255**
host_id	否	String	主机ID，不赋值时，查租户所有主机 最小长度：**0** 最大长度：**64**
limit	否	Integer	每页数量 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-154 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-155 响应 Body 参数

参数	参数类型	描述
total_num	Long	风险总数 最小值：**0** 最大值：**9223372036854775807**
data_list	Array of **CheckRuleRiskInfoResponseInfo** objects	数据列表 数组长度：**0 - 2147483647**

表 3-156 CheckRuleRiskInfoResponseInfo

参数	参数类型	描述
severity	String	风险等级，包含如下: ● Low : 低危 ● Medium : 中危 ● High : 高危 最小长度：**0** 最大长度：**255**
check_name	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**
check_type	String	配置检查（基线）的类型,Linux系统支持的基线一般check_type和check_name相同,例如SSH、CentOS 7。 Windows系统支持的基线一般check_type和check_name不相同，例如check_name为Windows的配置检查（基线），它的check_type包含Windows Server 2019 R2、Windows Server 2016 R2等。 最小长度：**0** 最大长度：**256**
standard	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 云安全实践标准 最小长度：**0** 最大长度：**16**

参数	参数类型	描述
check_rule_name	String	检查项（检查规则）名称 最小长度：**0** 最大长度：**2048**
check_rule_id	String	检查项ID 最小长度：**0** 最大长度：**64**
host_num	Integer	受影响的服务器的数量，进行了当前基线检测的服务器数量 最小值：**0** 最大值：**2147483647**
scan_result	String	检测结果，包含如下： • pass • failed 最小长度：**0** 最大长度：**64**
status	String	状态，包含如下： • safe：无需处理 • ignored：已忽略 • unhandled：未处理 • fixing：修复中 • fix-failed：修复失败 • verifying：验证中 最小长度：**0** 最大长度：**64**
enable_fix	Integer	是否支持一键修复,1:支持一键修复,0:不支持 最小值：**0** 最大值：**2147483647**
enable_click	Boolean	该检查项的修复&忽略&验证按钮是否可单击,true:按钮可单击,false:按钮不可单击
rule_params	Array of **CheckRuleFixParamInfo** objects	支持传递参数修复的检查项可传递参数的范围，只有支持传递参数修复的检查项才返回此数据 数组长度：**0 - 2147483647**

表 3-157 CheckRuleFixParamInfo

参数	参数类型	描述
rule_param_id	Integer	检查项参数ID 最小值：**0** 最大值：**10**
rule_desc	String	检查项参数描述 最小长度：**0** 最大长度：**256**
default_value	Integer	检查项参数默认值 最小值：**0** 最大值：**2147483647**
range_min	Integer	检查项参数可取最小值 最小值：**0** 最大值：**2147483647**
range_max	Integer	检查项参数可取最大值 最小值：**0** 最大值：**2147483647**

请求示例

查询企业项目id为xxx下的基线名称为SSH、检查标准为“云安全实践”的检查项列表。

```
GET https://{endpoint}/v5/{project_id}/baseline/risk-config/SSH/check-rules?
standard=hw_standard&enterprise_project_id=xxx

{
  "standard" : "hw_standard"
}
```

响应示例

状态码： 200

指定安全配置项的检查项列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "check_rule_id" : "1.1",
    "check_rule_name" : "Rule:Ensure that permissions on /etc/ssh/sshd_config are configured.",
    "check_type" : "SSH",
    "host_num" : 2,
    "standard" : "hw_standard",
    "scan_result" : "failed",
    "severity" : "High",
    "status" : "unhandled",
    "enable_fix" : 1,
    "enable_click" : true,
    "rule_params" : [ {
```

```
    "rule_param_id" : 1,
    "rule_desc" : "设置超时时间",
    "default_value" : 5,
    "range_min" : 1,
    "range_max" : 10
   }, {
    "rule_param_id" : 2,
    "rule_desc" : "设置重启次数",
    "default_value" : 10,
    "range_min" : 1,
    "range_max" : 20
   } ]
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询企业项目id为xxx下的基线名称为SSH、检查标准为“云安全实践”的检查项列表。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListRiskConfigCheckRulesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListRiskConfigCheckRulesRequest request = new ListRiskConfigCheckRulesRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withStandard("<standard>");
        request.withResultType("<result_type>");
        request.withCheckRuleName("<check_rule_name>");
        request.withSeverity("<severity>");
        request.withHostId("<host_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListRiskConfigCheckRulesResponse response = client.listRiskConfigCheckRules(request);
            System.out.println(response.toString());
```

```
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询企业项目id为xxx下的基线名称为SSH、检查标准为“云安全实践”的检查项列表。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListRiskConfigCheckRulesRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.standard = "<standard>"
        request.result_type = "<result_type>"
        request.check_rule_name = "<check_rule_name>"
        request.severity = "<severity>"
        request.host_id = "<host_id>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_risk_config_check_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

查询企业项目id为xxx下的基线名称为SSH、检查标准为“云安全实践”的检查项列表。

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListRiskConfigCheckRulesRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.Standard = "<standard>"
    resultTypeRequest:= "<result_type>"
    request.ResultType = &resultTypeRequest
    checkRuleNameRequest:= "<check_rule_name>"
    request.CheckRuleName = &checkRuleNameRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListRiskConfigCheckRules(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	指定安全配置项的检查项列表

错误码

请参见**错误码**。

3.3.6 查询指定安全配置项的受影响服务器列表

功能介绍

查询指定安全配置项的受影响服务器列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/risk-config/{check_name}/hosts

表 3-158 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**
check_name	是	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**

表 3-159 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
standard	是	String	标准类型，包含如下: • cn_standard : 等保合规标准 • hw_standard : 云安全实践标准 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**256**
host_ip	否	String	服务器IP地址 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页数量 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-160 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-161 响应 Body 参数

参数	参数类型	描述
total_num	Long	受配置检测影响的服务器数据总量 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
data_list	Array of SecurityCheckHostInfoResponseInfo objects	数据列表 数组长度：**0 - 2147483647**

表 3-162 SecurityCheckHostInfoResponseInfo

参数	参数类型	描述
host_id	String	主机ID 最小长度：**0** 最大长度：**64**
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**
host_public_ip	String	服务器公网IP 最小长度：**0** 最大长度：**128**
host_private_ip	String	服务器私网IP 最小长度：**0** 最大长度：**256**
scan_time	Long	扫描时间(ms) 最小值：**0** 最大值：**9223372036854775807**
failed_num	Integer	风险项数量 最小值：**0** 最大值：**2147483647**
passed_num	Integer	通过项数量 最小值：**0** 最大值：**2147483647**

请求示例

查询企业项目id为xxx下的基线名称为SSH、检查标准为“云安全实践”的受影响服务器列表。

```
GET https://{endpoint}/v5/{project_id}/baseline/risk-config/SSH/hosts?
standard=hw_standard&enterprise_project_id=xxx
```

响应示例

状态码： 200

安全配置项的受影响服务器列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "failed_num" : 6,
    "host_id" : "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    "host_name" : "zhangxiaodong2",
    "host_private_ip" : "192.168.0.129",
    "host_public_ip" : "*.*.*.10",
    "passed_num" : 10,
    "scan_time" : 1661716860935
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListRiskConfigHostsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListRiskConfigHostsRequest request = new ListRiskConfigHostsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withStandard("<standard>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListRiskConfigHostsResponse response = client.listRiskConfigHosts(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListRiskConfigHostsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.standard = "<standard>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_risk_config_hosts(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListRiskConfigHostsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.Standard = "<standard>"
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListRiskConfigHosts(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	安全配置项的受影响服务器列表

错误码

请参见**错误码**。

3.3.7 查询配置检查项检测报告

功能介绍

查询配置检查项检测报告

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/baseline/check-rule/detail

表 3-163 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-164 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
check_name	是	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**255**
check_type	是	String	配置检查（基线）的类型,Linux系统支持的基线一般check_type和check_name相同,例如SSH、CentOS 7。 Windows系统支持的基线一般check_type和check_name不相同，例如check_name为Windows的配置检查（基线），它的check_type包含Windows Server 2019 R2、Windows Server 2016 R2等。check_type的值可以通过这个接口的返回数据获得：/v5/{project_id}/baseline/risk-configs 最小长度：**0** 最大长度：**255**

参数	是否必选	参数类型	描述
check_rule_id	是	String	检查项ID，值可以通过这个接口的返回数据获得：/v5/{project_id}/baseline/risk-config/{check_name}/check-rules 最小长度：**0** 最大长度：**255**
standard	是	String	标准类型，包含如下: • cn_standard : 等保合规标准 • hw_standard : 云安全实践标准 最小长度：**0** 最大长度：**32**
host_id	否	String	主机ID 最小长度：**0** 最大长度：**64**

请求参数

表 3-165 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） 最小长度：**32** 最大长度：**2097152**

响应参数

状态码： 200

表 3-166 响应 Body 参数

参数	参数类型	描述
description	String	当前检查项（检测规则）的描述 最小长度：**0** 最大长度：**2048**

参数	参数类型	描述
reference	String	当前检查项（检测规则）的制定依据 最小长度：**0** 最大长度：**255**
audit	String	当前检查项（检测规则）的审计描述 最小长度：**0** 最大长度：**65534**
remediation	String	当前检查项（检测规则）的修改建议 最小长度：**0** 最大长度：**65534**
check_info_list	Array of **CheckRuleCheckCaseResponseInfo** objects	检测用例信息 数组长度：**0 - 2147483647**

表 3-167 CheckRuleCheckCaseResponseInfo

参数	参数类型	描述
check_description	String	检测用例描述 最小长度：**0** 最大长度：**65534**
current_value	String	当前结果 最小长度：**0** 最大长度：**65534**
suggest_value	String	期待结果 最小长度：**0** 最大长度：**65534**

请求示例

查询企业项目id为xxx下的基线名称为SSH、检查项ID为1.12、检查标准为云安全实践标准的配置检查项检测报告。

```
GET https://{endpoint}/v5/{project_id}/baseline/check-rule/detail?standard=hw_standard&enterprise_project_id=xxx&check_name=SSH&check_type=SSH&check_rule_id=1.12
```

响应示例

状态码： 200

配置检查项检测报告

{
  "audit" : "Run the following commands and verify that ClientAliveInterval is smaller than 300 and ClientAliveCountMax is 3 or less: \n#grep '^ClientAliveInterval' /etc/ssh/sshd_config\nClientAliveInterval 300(default is 0) \n#grep '^ClientAliveCountMax' /etc/ssh/sshd_config\nClientAliveCountMax 0(default is 3)",
  "description" : "The two options ClientAliveInterval and ClientAliveCountMax control the timeout of SSH sessions. The ClientAliveInterval parameter sets a timeout interval in seconds after which if no data has been received from the client, sshd will send a message through the encrypted channel to request a response from the client. The ClientAliveCountMax parameter sets the number of client alive messages which may be sent without sshd receiving any messages back from the client. For example, if the ClientAliveInterval is set to 15s and the ClientAliveCountMax is set to 3, unresponsive SSH clients will be disconnected after approximately 45s.",
  "reference" : "",
  "remediation" : "Edit the /etc/ssh/sshd_config file to set the parameter as follows: \nClientAliveInterval 300 \nClientAliveCountMax 0"
}

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowCheckRuleDetailSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowCheckRuleDetailRequest request = new ShowCheckRuleDetailRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withCheckName("<check_name>");
        request.withCheckType("<check_type>");
        request.withCheckRuleId("<check_rule_id>");
        request.withStandard("<standard>");
        request.withHostId("<host_id>");
        try {
            ShowCheckRuleDetailResponse response = client.showCheckRuleDetail(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowCheckRuleDetailRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.check_name = "<check_name>"
        request.check_type = "<check_type>"
        request.check_rule_id = "<check_rule_id>"
        request.standard = "<standard>"
        request.host_id = "<host_id>"
        response = client.show_check_rule_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowCheckRuleDetailRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.CheckName = "<check_name>"
    request.CheckType = "<check_type>"
    request.CheckRuleId = "<check_rule_id>"
    request.Standard = "<standard>"
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    response, err := client.ShowCheckRuleDetail(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	配置检查项检测报告

错误码

请参见**错误码**。

3.3.8 对未通过的配置检查项进行忽略/取消忽略/修复/验证操作

功能介绍

对未通过的配置检查项进行忽略/取消忽略/修复/验证操作

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/baseline/check-rule/action

表 3-168 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-169 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
host_id	否	String	主机ID，不赋值时，查租户所有主机 最小长度：**0** 最大长度：**64**
action	是	String	动作 ● "ignore" ● "unignore" ● "fix" ● "verify" 缺省值：**ignore** 最小长度：**0** 最大长度：**32**

请求参数

表 3-170 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） 最小长度：**32** 最大长度：**2097152**

表 3-171 请求 Body 参数

参数	是否必选	参数类型	描述
check_rules	否	Array of **CheckRuleKeyInfoRequestInfo** objects	检查项ID列表 数组长度：**0 - 2147483647**

表 3-172 CheckRuleKeyInfoRequestInfo

参数	是否必选	参数类型	描述
check_name	否	String	配置检查（基线）的名称，例如SSH、CentOS 7、Windows 最小长度：**0** 最大长度：**256**
check_rule_id	否	String	检查项ID，值可以通过这个接口的返回数据获得：/v5/{project_id}/baseline/risk-config/{check_name}/check-rules 最小长度：**0** 最大长度：**64**
standard	否	String	基线标准, 类别包含如下： • cn_standard#等保合规标准 • hw_standard#云安全实践标准 最小长度：**0** 最大长度：**16**

参数	是否必选	参数类型	描述
fix_values	否	Array of CheckRuleFixValuesInfo objects	用户键入的检查项修复参数数组 数组长度：**0 - 10000**

表 3-173 CheckRuleFixValuesInfo

参数	是否必选	参数类型	描述
rule_param_id	否	Integer	检查项的参数ID 最小值：**0** 最大值：**2147483647**
fix_value	否	Integer	检查项的参数值 最小值：**0** 最大值：**2147483647**

响应参数

无

请求示例

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行忽略操作，此操作针对这条检查项的所有受影响主机。

```
PUT https://{endpoint}/v5/{project_id}/baseline/check-rule/action?
enterprise_project_id=xxx&action=ignore

{
  "check_rules" : [ {
    "check_name" : "SSH",
    "check_rule_id" : "1.11",
    "standard" : "hw_standard"
  } ]
}
```

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行修复操作，此操作只针对主机id为xxx的主机，修复参数为：将ID为1的修复项值设置为5，将ID为2的修复项值设置为20。

```
PUT https://{endpoint}/v5/{project_id}/baseline/check-rule/action?
enterprise_project_id=xxx&host_id=xxx&action=fix

{
  "check_rules" : [ {
    "check_name" : "SSH",
    "check_rule_id" : "1.11",
    "standard" : "hw_standard",
    "fix_values" : [ {
      "rule_param_id" : 1,
      "fix_value" : 5
    }, {
      "rule_param_id" : 2,
      "fix_value" : 20
```

```
  } ]
 } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行忽略操作，此操作针对这条检查项的所有受影响主机。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeCheckRuleActionSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeCheckRuleActionRequest request = new ChangeCheckRuleActionRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostId("<host_id>");
        request.withAction("<action>");
        CheckRuleIdListRequestInfo body = new CheckRuleIdListRequestInfo();
        List<CheckRuleKeyInfoRequestInfo> listbodyCheckRules = new ArrayList<>();
        listbodyCheckRules.add(
            new CheckRuleKeyInfoRequestInfo()
                .withCheckName("SSH")
                .withCheckRuleId("1.11")
                .withStandard("hw_standard")
        );
        body.withCheckRules(listbodyCheckRules);
        request.withBody(body);
        try {
            ChangeCheckRuleActionResponse response = client.changeCheckRuleAction(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());
        }
    }
}
```

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行修复操作，此操作只针对主机id为xxx的主机，修复参数为：将ID为1的修复项值设置为5，将ID为2的修复项值设置为20。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeCheckRuleActionSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeCheckRuleActionRequest request = new ChangeCheckRuleActionRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostId("<host_id>");
        request.withAction("<action>");
        CheckRuleIdListRequestInfo body = new CheckRuleIdListRequestInfo();
        List<CheckRuleFixValuesInfo> listCheckRulesFixValues = new ArrayList<>();
        listCheckRulesFixValues.add(
            new CheckRuleFixValuesInfo()
                .withRuleParamId(1)
                .withFixValue(5)
        );
        listCheckRulesFixValues.add(
            new CheckRuleFixValuesInfo()
                .withRuleParamId(2)
                .withFixValue(20)
        );
        List<CheckRuleKeyInfoRequestInfo> listbodyCheckRules = new ArrayList<>();
```

```
        listbodyCheckRules.add(
            new CheckRuleKeyInfoRequestInfo()
                .withCheckName("SSH")
                .withCheckRuleId("1.11")
                .withStandard("hw_standard")
                .withFixValues(listCheckRulesFixValues)
        );
        body.withCheckRules(listbodyCheckRules);
        request.withBody(body);
        try {
            ChangeCheckRuleActionResponse response = client.changeCheckRuleAction(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

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行忽略操作，此操作针对这条检查项的所有受影响主机。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeCheckRuleActionRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_id = "<host_id>"
        request.action = "<action>"
        listCheckRulesbody = [
            CheckRuleKeyInfoRequestInfo(
                check_name="SSH",
                check_rule_id="1.11",
                standard="hw_standard"
            )
        ]
        request.body = CheckRuleIdListRequestInfo(
```

```
            check_rules=listCheckRulesbody
        )
        response = client.change_check_rule_action(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行修复操作，此操作只针对主机id为xxx的主机，修复参数为：将ID为1的修复项值设置为5，将ID为2的修复项值设置为20。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeCheckRuleActionRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_id = "<host_id>"
        request.action = "<action>"
        listFixValuesCheckRules = [
            CheckRuleFixValuesInfo(
                rule_param_id=1,
                fix_value=5
            ),
            CheckRuleFixValuesInfo(
                rule_param_id=2,
                fix_value=20
            )
        ]
        listCheckRulesbody = [
            CheckRuleKeyInfoRequestInfo(
                check_name="SSH",
                check_rule_id="1.11",
                standard="hw_standard",
                fix_values=listFixValuesCheckRules
            )
        ]
        request.body = CheckRuleIdListRequestInfo(
            check_rules=listCheckRulesbody
        )
        response = client.change_check_rule_action(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

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行忽略操作，此操作针对这条检查项的所有受影响主机。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeCheckRuleActionRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    request.Action = "<action>"
    checkNameCheckRules:= "SSH"
    checkRuleIdCheckRules:= "1.11"
    standardCheckRules:= "hw_standard"
    var listCheckRulesbody = []model.CheckRuleKeyInfoRequestInfo{
        {
            CheckName: &checkNameCheckRules,
            CheckRuleId: &checkRuleIdCheckRules,
            Standard: &standardCheckRules,
        },
    }
    request.Body = &model.CheckRuleIdListRequestInfo{
        CheckRules: &listCheckRulesbody,
    }
    response, err := client.ChangeCheckRuleAction(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 对企业项目id为xxx下的基线名称为SSH、检查项ID为1.11、检查标准为云安全实践标准的配置检查项进行修复操作，此操作只针对主机id为xxx的主机，修复参数为：将ID为1的修复项值设置为5，将ID为2的修复项值设置为20。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeCheckRuleActionRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    request.Action = "<action>"
    ruleParamIdFixValues:= int32(1)
    fixValueFixValues:= int32(5)
    ruleParamIdFixValues1:= int32(2)
    fixValueFixValues1:= int32(20)
    var listFixValuesCheckRules = []model.CheckRuleFixValuesInfo{
        {
            RuleParamId: &ruleParamIdFixValues,
            FixValue: &fixValueFixValues,
        },
        {
            RuleParamId: &ruleParamIdFixValues1,
            FixValue: &fixValueFixValues1,
        },
    }
    checkNameCheckRules:= "SSH"
    checkRuleIdCheckRules:= "1.11"
    standardCheckRules:= "hw_standard"
    var listCheckRulesbody = []model.CheckRuleKeyInfoRequestInfo{
        {
            CheckName: &checkNameCheckRules,
            CheckRuleId: &checkRuleIdCheckRules,
            Standard: &standardCheckRules,
            FixValues: &listFixValuesCheckRules,
        },
    }
    request.Body = &model.CheckRuleIdListRequestInfo{
        CheckRules: &listCheckRulesbody,
    }
    response, err := client.ChangeCheckRuleAction(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	执行成功

错误码

请参见**错误码**。

3.4 配额管理

3.4.1 查询配额信息

功能介绍

查询配额信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/billing/quotas

表 3-174 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**128**

表 3-175 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
version	否	String	主机开通的版本，包含如下7种输入。 ● hss.version.null ： 无。 ● hss.version.basic ： 基础版。 ● hss.version.advanced ： 专业版。 ● hss.version.enterprise ： 企业版。 ● hss.version.premium ： 旗舰版。 ● hss.version.wtp ： 网页防篡改版。 ● hss.version.container.enterprise：容器版。 最小长度：**1** 最大长度：**64**
charging_mode	否	String	收费模式，包含如下2种。 ● packet_cycle ： 包年/包月。 ● on_demand ： 按需。 最小长度：**1** 最大长度：**32**

请求参数

表 3-176 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） 最小长度：**32** 最大长度：**4096**

参数	是否必选	参数类型	描述
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-177 响应 Body 参数

参数	参数类型	描述
data_list	Array of **ResourceQuotasInfo** objects	配额统计列表 数组长度：**0 - 200**

表 3-178 ResourceQuotasInfo

参数	参数类型	描述
version	String	主机开通的版本，包含如下7种输入。 • hss.version.null ： 无。 • hss.version.basic ： 基础版。 • hss.version.advanced ： 专业版。 • hss.version.enterprise ： 企业版。 • hss.version.premium ： 旗舰版。 • hss.version.wtp ： 网页防篡改版。 • hss.version.container.enterprise：容器版。 最小长度：**1** 最大长度：**64**
total_num	Integer	总配额数 最小值：**0** 最大值：**2000000**
used_num	Integer	已使用配额数 最小值：**0** 最大值：**2000000**

参数	参数类型	描述
available_num	Integer	可用总配额数 最小值：**0** 最大值：**2000000**
available_resources_list	Array of **AvailableResourceIdsInfo** objects	可用资源列表 数组长度：**0 - 200**

表 3-179 AvailableResourceIdsInfo

参数	参数类型	描述
resource_id	String	资源ID 最小长度：**1** 最大长度：**256**
current_time	String	当前时间 最小长度：**1** 最大长度：**64**
shared_quota	String	是否共享配额 • shared：共享的 • unshared：非共享的 最小长度：**1** 最大长度：**64**

请求示例

查询所有企业项目下的基础版配额信息

```
GET https://{endpoint}/v5/{project_id}/billing/quotas?
version=hss.version.basic&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

配额统计列表

```
{
  "data_list" : [ {
    "available_num" : 1,
    "available_resources_list" : [ {
      "current_time" : "2022-09-17T17:00:24Z",
      "resource_id" : "9ecb83a7-8b03-4e37-a26d-c3e90ca97eea",
      "shared_quota" : "shared"
    } ],
    "total_num" : 2,
```

```
    "used_num" : 1,
    "version" : "hss.version.basic"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowResourceQuotasSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowResourceQuotasRequest request = new ShowResourceQuotasRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withVersion("<version>");
        request.withChargingMode("<charging_mode>");
        try {
            ShowResourceQuotasResponse response = client.showResourceQuotas(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowResourceQuotasRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.version = "<version>"
        request.charging_mode = "<charging_mode>"
        response = client.show_resource_quotas(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowResourceQuotasRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
```

```
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    versionRequest:= "<version>"
    request.Version = &versionRequest
    chargingModeRequest:= "<charging_mode>"
    request.ChargingMode = &chargingModeRequest
    response, err := client.ShowResourceQuotas(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	配额统计列表

错误码

请参见**错误码**。

3.4.2 查询配额详情

功能介绍

查询配额详情

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/billing/quotas-detail

表 3-180 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-181 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
version	否	String	主机开通的版本，包含如下7种输入。 ● hss.version.null ： 无。 ● hss.version.basic ： 基础版。 ● hss.version.advanced ： 专业版。 ● hss.version.enterprise ： 企业版。 ● hss.version.premium ： 旗舰版。 ● hss.version.wtp ： 网页防篡改版。 ● hss.version.container.enterprise：容器版。 最小长度：**1** 最大长度：**64**
category	否	String	类别，包含如下几种： ● host_resource ： HOST_RESOURCE ● container_resource ： CONTAINER_RESOURCE 最小长度：**1** 最大长度：**64**
quota_status	否	String	配额状态，包含如下几种： ● normal ： QUOTA_STATUS_NORMAL ● expired ： QUOTA_STATUS_EXPIRED ● freeze ： QUOTA_STATUS_FREEZE 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
used_status	否	String	使用状态，包含如下几种： ● idle： USED_STATUS_IDLE ● used： USED_STATUS_USED 最小长度：**1** 最大长度：**64**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**128**
resource_id	否	String	HSS配额的资源ID 最小长度：**0** 最大长度：**128**
charging_mode	否	String	收费模式，包含如下2种。 ● packet_cycle： 包年/包月。 ● on_demand： 按需。 最小长度：**1** 最大长度：**32**
limit	否	Integer	每页数量 最小值：**10** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-182 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） 最小长度：**32** 最大长度：**4096**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-183 响应 Body 参数

参数	参数类型	描述
packet_cycle_num	Integer	包周期配额数 最小值：**0** 最大值：**10000000**
on_demand_num	Integer	按需配额数 最小值：**0** 最大值：**10000000**
used_num	Integer	已使用配额数 最小值：**0** 最大值：**10000000**
idle_num	Integer	空闲配额数 最小值：**0** 最大值：**10000000**
normal_num	Integer	正常配额数 最小值：**0** 最大值：**10000000**
expired_num	Integer	过期配额数 最小值：**0** 最大值：**10000000**

参数	参数类型	描述
freeze_num	Integer	冻结配额数 最小值：**0** 最大值：**10000000**
quota_statistics_list	Array of **QuotaStatisticsResponseInfo** objects	配额统计列表 数组长度：**0 - 200**
total_num	Integer	配额总数 最小值：**0** 最大值：**10000000**
data_list	Array of **QuotaResourcesResponseInfo** objects	配额列表 数组长度：**0 - 200**

表 3-184 QuotaStatisticsResponseInfo

参数	参数类型	描述
version	String	资源规格编码，包含如下: • hss.version.basic : 基础版 • hss.version.advanced : 专业版 • hss.version.enterprise : 企业版 • hss.version.premium : 旗舰版 • hss.version.wtp : 网页防篡改版 • hss.version.container : 容器版 最小长度：**1** 最大长度：**64**
total_num	Integer	配额总数 最小值：**0** 最大值：**10000000**

表 3-185 QuotaResourcesResponseInfo

参数	参数类型	描述
resource_id	String	主机安全配额的资源ID 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
version	String	资源规格编码，包含如下: ● hss.version.basic : 基础版 ● hss.version.advanced : 专业版 ● hss.version.enterprise : 企业版 ● hss.version.premium : 旗舰版 ● hss.version.wtp : 网页防篡改版 ● hss.version.container : 容器版 最小长度：**1** 最大长度：**64**
quota_status	String	配额状态 ● normal : 正常 ● expired : 已过期 ● freeze : 已冻结 最小长度：**1** 最大长度：**64**
used_status	String	使用状态 ● idle : 空闲 ● used : 使用中 最小长度：**1** 最大长度：**64**
host_id	String	主机ID 最小长度：**1** 最大长度：**64**
host_name	String	服务器名称 最小长度：**1** 最大长度：**128**
charging_mode	String	计费模式 ● packet_cycle : 包周期 ● on_demand : 按需 最小长度：**1** 最大长度：**64**
tags	Array of **TagInfo** objects	标签 数组长度：**0 - 2097152**

参数	参数类型	描述
expire_time	Long	过期时间，-1表示没有到期时间 最小值：**0** 最大值：**2147483647**
shared_quota	String	是否共享配额 ● shared：共享的 ● unshared：非共享的 最小长度：**1** 最大长度：**64**
enterprise_project_id	String	企业项目ID 最小长度：**0** 最大长度：**256**
enterprise_project_name	String	所属企业项目名称 最小长度：**0** 最大长度：**256**

表 3-186 TagInfo

参数	参数类型	描述
key	String	键。最大长度128个unicode字符。 key不能为空 最小长度：**1** 最大长度：**128**
value	String	值。最大长度255个unicode字符。 最小长度：**1** 最大长度：**255**

请求示例

查询所有企业项目下的配额详情

```
GET https://{endpoint}/v5/{project_id}/billing/quotas-detail?
offset=0&limit=100&version=hss.version.basic&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

配额详情列表

```
{
  "data_list" : [ {
    "charging_mode" : "packet_cycle",
    "expire_time" : -1,
```

```
    "host_id" : "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    "host_name" : "zhangxiaodong2",
    "quota_status" : "normal",
    "resource_id" : "af4d08ad-2b60-4916-a5cf-8d6a23956dda",
    "shared_quota" : "shared",
    "tags" : [ {
      "key" : "服务",
      "value" : "HSS"
    } ],
    "used_status" : "used",
    "version" : "hss.version.wtp"
  } ],
  "expired_num" : 0,
  "freeze_num" : 0,
  "idle_num" : 20,
  "normal_num" : 60,
  "on_demand_num" : 0,
  "packet_cycle_num" : 60,
  "quota_statistics_list" : [ {
    "total_num" : 8,
    "version" : "hss.version.basic"
  } ],
  "total_num" : 60,
  "used_num" : 40
}
```


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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListQuotasDetailSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListQuotasDetailRequest request = new ListQuotasDetailRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withVersion("<version>");
        request.withCategory("<category>");
        request.withQuotaStatus("<quota_status>");
```

```
        request.withUsedStatus("<used_status>");
        request.withHostName("<host_name>");
        request.withResourceId("<resource_id>");
        request.withChargingMode("<charging_mode>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListQuotasDetailResponse response = client.listQuotasDetail(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListQuotasDetailRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.version = "<version>"
        request.category = "<category>"
        request.quota_status = "<quota_status>"
        request.used_status = "<used_status>"
        request.host_name = "<host_name>"
        request.resource_id = "<resource_id>"
        request.charging_mode = "<charging_mode>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_quotas_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListQuotasDetailRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    versionRequest:= "<version>"
    request.Version = &versionRequest
    categoryRequest:= "<category>"
    request.Category = &categoryRequest
    quotaStatusRequest:= "<quota_status>"
    request.QuotaStatus = &quotaStatusRequest
    usedStatusRequest:= "<used_status>"
    request.UsedStatus = &usedStatusRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    resourceIdRequest:= "<resource_id>"
    request.ResourceId = &resourceIdRequest
    chargingModeRequest:= "<charging_mode>"
    request.ChargingMode = &chargingModeRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListQuotasDetail(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	配额详情列表

错误码

请参见**错误码**。

3.4.3 HSS 服务创建订单订购配额

功能介绍

HSS服务创建订单订购配额，只支持包周期计费模式

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/quotas/orders

表 3-187 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**512**

表 3-188 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**

请求参数

表 3-189 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） 最小长度：**0** 最大长度：**4096**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**
region	是	String	Region ID 最小长度：**0** 最大长度：**32**

表 3-190 请求 Body 参数

参数	是否必选	参数类型	描述
resource_spec_code	是	String	资源规格 ● hss.version.basic ： 基础版。 ● hss.version.advanced ： 专业版。 ● hss.version.enterprise ： 企业版。 ● hss.version.premium ： 旗舰版。 ● hss.version.wtp ： 网页防篡改版。 ● hss.version.container.enterprise：容器版。 最小长度：**0** 最大长度：**128**
period_type	是	Integer	订购周期类型 ● 2 : 月 ● 3 : 年 最小值：**0** 最大值：**100**

参数	是否必选	参数类型	描述
period_num	是	Integer	订购周期数 最小值：**0** 最大值：**1000**
is_auto_renew	否	Boolean	是否支持自动续订，true表示自动续订，false表示不自动续订，默认值为false
is_auto_pay	否	Boolean	是否支持自动支付，true表示支持，false表示不支持，默认值为false
subscription_num	是	Integer	订购数量 最小值：**0** 最大值：**500**

响应参数

状态码： 200

表 3-191 响应 Body 参数

参数	参数类型	描述
order_id	String	订单ID 最小长度：**0** 最大长度：**256**

请求示例

创建订单订购企业主机安全企业版防护配额，订单的相关信息：计费模式为包周期，订购数量为1，订购周期数为1，订购周期类型为按月，不自动续订，自动支付

```
POST https://{endpoint}/v5/{project_id}/quotas/orders

{
  "resource_spec_code" : "hss.version.enterprise",
  "subscription_num" : 1,
  "period_num" : 1,
  "period_type" : 2,
  "is_auto_renew" : false,
  "is_auto_pay" : false
}
```

响应示例

状态码： 200

订购相关信息

```
{
  "order_id" : "CS2404171642AAAAA"
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建订单订购企业主机安全企业版防护配额，订单的相关信息：计费模式为包周期，订购数量为1，订购周期数为1，订购周期类型为按月，不自动续订，自动支付

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class CreateQuotasOrderSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateQuotasOrderRequest request = new CreateQuotasOrderRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        CreateQuotasOrderRequestInfo body = new CreateQuotasOrderRequestInfo();
        body.withSubscriptionNum(1);
        body.withIsAutoPay(false);
        body.withIsAutoRenew(false);
        body.withPeriodNum(1);
        body.withPeriodType(2);
        body.withResourceSpecCode("hss.version.enterprise");
        request.withBody(body);
        try {
            CreateQuotasOrderResponse response = client.createQuotasOrder(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

创建订单订购企业主机安全企业版防护配额，订单的相关信息：计费模式为包周期，订购数量为1，订购周期数为1，订购周期类型为按月，不自动续订，自动支付

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = CreateQuotasOrderRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.body = CreateQuotasOrderRequestInfo(
            subscription_num=1,
            is_auto_pay=False,
            is_auto_renew=False,
            period_num=1,
            period_type=2,
            resource_spec_code="hss.version.enterprise"
        )
        response = client.create_quotas_order(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

创建订单订购企业主机安全企业版防护配额，订单的相关信息：计费模式为包周期，订购数量为1，订购周期数为1，订购周期类型为按月，不自动续订，自动支付

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateQuotasOrderRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    isAutoPayCreateQuotasOrderRequestInfo:= false
    isAutoRenewCreateQuotasOrderRequestInfo:= false
    request.Body = &model.CreateQuotasOrderRequestInfo{
        SubscriptionNum: int32(1),
        IsAutoPay: &isAutoPayCreateQuotasOrderRequestInfo,
        IsAutoRenew: &isAutoRenewCreateQuotasOrderRequestInfo,
        PeriodNum: int32(1),
        PeriodType: int32(2),
        ResourceSpecCode: "hss.version.enterprise",
    }
    response, err := client.CreateQuotasOrder(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	订购相关信息

错误码

请参见**错误码**。

3.4.4 查询产商品信息

功能介绍

查询产商品信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/product/productdata/offering-infos

表 3-192 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**128**

表 3-193 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
site_code	否	String	站点信息： • HWC_CN ： 中国站 • HWC_HK ： 国际站 • HWC_EU : 欧洲站 缺省值：**HWC_CN** 最小长度：**1** 最大长度：**256**

请求参数

表 3-194 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） 最小长度：**32** 最大长度：**4096**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-195 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of **ResourceProductDataObjectInfo** objects	商品数据列表

表 3-196 ResourceProductDataObjectInfo

参数	参数类型	描述
charging_mode	String	计费模式 • packet_cycle : 包周期 • on_demand : 按需 最小长度：**1** 最大长度：**32**
is_auto_renew	Boolean	是否自动续费

参数	参数类型	描述
version_info	Map<String,Array<**ShowPeriodResponseInfo**>>	版本信息,key对应的值为主机开通的版本，包含如下6种输入： ● hss.version.basic ： 基础版。 ● hss.version.advanced ： 专业版。 ● hss.version.enterprise ： 企业版。 ● hss.version.premium ： 旗舰版。 ● hss.version.wtp ： 网页防篡改版。 ● hss.version.container.enterprise ： 容器版。

表 3-197 ShowPeriodResponseInfo

参数	参数类型	描述
period_vals	String	购买时长数值串，多个用逗号分隔，如1,2,3,4,5,6,7,8,9 最小长度：**1** 最大长度：**32**
period_unit	String	购买时长单位 ● year ： 年 ● month ： 月 ● day ： 日 最小长度：**1** 最大长度：**32**

请求示例

无

响应示例

状态码： 200

产商品信息列表

```
{
  "data_list" : [ {
    "charging_mode" : "packet_cycle",
    "is_auto_renew" : false,
    "version_info" : {
      "hss.version.enterprise" : [ {
        "period_vals" : "1,2,3,4,5,6,7,8,9",
        "period_unit" : "month"
      }, {
        "period_vals" : "1,2,3,5",
        "period_unit" : "year"
      } ],
```

```
    "hss.version.premium" : [ {
      "period_vals" : "1,2,3,4,5,6,7,8,9",
      "period_unit" : "month"
    }, {
      "period_vals" : "1,2,3,5",
      "period_unit" : "year"
    } ]
  }
 }, {
  "charging_mode" : "on_demand",
  "is_auto_renew" : false,
  "version_info" : {
    "hss.version.enterprise" : [ {
      "period_vals" : "1,2,3,4,5,6,7,8,9",
      "period_unit" : "month"
    }, {
      "period_vals" : "1,2,3,5",
      "period_unit" : "year"
    } ]
  }
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowProductdataOfferingInfosSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowProductdataOfferingInfosRequest request = new ShowProductdataOfferingInfosRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withSiteCode("<site_code>");
        try {
            ShowProductdataOfferingInfosResponse response = client.showProductdataOfferingInfos(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowProductdataOfferingInfosRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.site_code = "<site_code>"
        response = client.show_productdata_offering_infos(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowProductdataOfferingInfosRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    siteCodeRequest:= "<site_code>"
    request.SiteCode = &siteCodeRequest
    response, err := client.ShowProductdataOfferingInfos(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	产商品信息列表

错误码

请参见**错误码**。

3.5 容器管理

3.5.1 查询容器节点列表

功能介绍

查询容器节点列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/container/nodes

表 3-198 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-199 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**200** 缺省值：**10**
host_name	否	String	节点名称 最小长度：**0** 最大长度：**128**
agent_status	否	String	Agent状态，包含如下3种。 ● not_installed ： 未安装 ● online ： 在线 ● offline ： 离线 最小长度：**1** 最大长度：**32**

参数	是否必选	参数类型	描述
protect_status	否	String	防护状态，包含如下2种。 ● closed ： 关闭 ● opened ： 开启 最小长度：**1** 最大长度：**32**
container_tags	否	String	标签：用来识别cce容器节点和自建 ● cce：cce节点 ● self：自建节点 ● other：其他节点 最小长度：**1** 最大长度：**32**

请求参数

表 3-200 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-201 响应 Body 参数

参数	参数类型	描述
total_num	Integer	容器节点总数 最小值：**0** 最大值：**65535**

参数	参数类型	描述
data_list	Array of ContainerNodeInfo objects	容器节点列表 数组长度：**0 - 65535**

表 3-202 ContainerNodeInfo

参数	参数类型	描述
agent_id	String	Agent ID 最小长度：**0** 最大长度：**64**
host_id	String	服务器ID 最小长度：**0** 最大长度：**128**
host_name	String	节点名称 最小长度：**0** 最大长度：**128**
host_status	String	服务器状态，包含如下4种。 • ACTIVE： 正在运行。 • SHUTOFF： 关机。 • BUILDING： 创建中。 • ERROR： 故障。 最小长度：**1** 最大长度：**32**
agent_status	String	Agent状态，包含如下3种。 • not_installed： 未安装。 • online： 在线。 • offline： 离线。 最小长度：**1** 最大长度：**32**
protect_status	String	防护状态，包含如下2种。 • closed： 关闭。 • opened： 开启。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
protect_interrupt	Boolean	防护是否中断
container_tags	String	标签：用来识别cce容器节点和自建 ● cce：cce节点 ● self：自建节点 ● other：其他节点 最小长度：**1** 最大长度：**32**
private_ip	String	私有IP地址 最小长度：**0** 最大长度：**128**
public_ip	String	弹性公网IP地址 最小长度：**0** 最大长度：**128**
resource_id	String	主机安全配额ID（UUID） 最小长度：**0** 最大长度：**128**
group_name	String	服务器组名称 最小长度：**1** 最大长度：**128**
enterprise_project_name	String	所属企业项目名称 最小长度：**0** 最大长度：**256**
detect_result	String	云主机安全检测结果，包含如下4种。 ● undetected： 未检测。 ● clean： 无风险。 ● risk： 有风险。 ● scanning： 检测中。 最小长度：**1** 最大长度：**32**
asset	Integer	资产风险 最小值：**0** 最大值：**2097152**
vulnerability	Integer	漏洞风险 最小值：**0** 最大值：**2097152**

参数	参数类型	描述
intrusion	Integer	入侵风险 最小值：**0** 最大值：**2097152**
policy_group_id	String	策略组ID 最小长度：**1** 最大长度：**128**
policy_group_name	String	策略组名称 最小长度：**1** 最大长度：**128**

请求示例

查询容器节点列表，不传limit参数默认返回10条。

```
GET https://{endpoint}/v5/{project_id}/container/nodes
```

响应示例

状态码： 200

容器节点列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "2d0fe7824005bf001220ad9d892e86f8af44XXXXXXXXXX",
    "agent_status" : "online",
    "host_id" : "host_id",
    "host_name" : "host_name",
    "host_status" : "ACTIVE",
    "protect_status" : "opened",
    "protect_interrupt" : false,
    "private_ip" : "192.168.0.114",
    "public_ip" : "100.85.218.122",
    "resource_id" : "ef5eb4fd-7376-48ac-886f-16fd057776f3",
    "group_name" : "as(All projects)",
    "enterprise_project_name" : "default",
    "detect_result" : "risk",
    "asset" : 0,
    "vulnerability" : 14,
    "intrusion" : 0,
    "policy_group_id" : "ce4d5e95-0cbf-4102-9c77-ef1bcb6b35aa",
    "policy_group_name" : "tenant_linux_enterprise_default_policy_group (All projects)"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListContainerNodesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListContainerNodesRequest request = new ListContainerNodesRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withHostName("<host_name>");
        request.withAgentStatus("<agent_status>");
        request.withProtectStatus("<protect_status>");
        request.withContainerTags("<container_tags>");
        try {
            ListContainerNodesResponse response = client.listContainerNodes(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListContainerNodesRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.offset = <offset>
        request.limit = <limit>
        request.host_name = "<host_name>"
        request.agent_status = "<agent_status>"
        request.protect_status = "<protect_status>"
        request.container_tags = "<container_tags>"
        response = client.list_container_nodes(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListContainerNodesRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
```

```
    agentStatusRequest:= "<agent_status>"
    request.AgentStatus = &agentStatusRequest
    protectStatusRequest:= "<protect_status>"
    request.ProtectStatus = &protectStatusRequest
    containerTagsRequest:= "<container_tags>"
    request.ContainerTags = &containerTagsRequest
    response, err := client.ListContainerNodes(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	容器节点列表

错误码

请参见**错误码**。

3.5.2 查询容器基本信息列表

功能介绍

查询容器基本信息列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/container/kubernetes

表 3-203 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**128**

表 3-204 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
container_name	否	String	容器名称 最小长度：**0** 最大长度：**128**
pod_name	否	String	所属Pod名称 最小长度：**0** 最大长度：**128**
image_name	否	String	镜像名称 最小长度：**0** 最大长度：**128**
cluster_container	否	Boolean	是否是集群纳管的容器 缺省值：**false**
limit	否	Integer	每页显示个数 最小值：**1** 最大值：**1000** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-205 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。 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-206 响应 Body 参数

参数	参数类型	描述
total_num	Integer	容器总数 最小值：**0** 最大值：**2147483647**
last_update_time	Long	最近更新时间 最小值：**0** 最大值：**4071095999000**
data_list	Array of **ContainerBaseInfo** objects	容器基本信息列表 数组长度：**0 - 2147483647**

表 3-207 ContainerBaseInfo

参数	参数类型	描述
container_id	String	容器ID 最小长度：**0** 最大长度：**255**
container_name	String	容器名称 最小长度：**0** 最大长度：**255**
image_name	String	镜像名称 最小长度：**0** 最大长度：**255**
status	String	容器状态，包含以下几种： -Running 运行中 -Terminated 终止 -Waiting 等待 最小长度：**0** 最大长度：**64**
create_time	Long	创建时间 最小值：**0** 最大值：**4071095999000**
cpu_limit	String	cpu限制 最小长度：**0** 最大长度：**64**

参数	参数类型	描述
memory_limit	String	内存限制 最小长度：**0** 最大长度：**64**
restart_count	Integer	重启次数 最小值：**0** 最大值：**20**
pod_name	String	所属pod名称 最小长度：**0** 最大长度：**64**
cluster_name	String	所属集群 最小长度：**0** 最大长度：**64**
cluster_id	String	集群id 最小长度：**0** 最大长度：**64**
cluster_type	String	集群类型，包含以下几种： -k8s 原生集群 -cce CCE集群 -ali 阿里云集群 -tencent 腾讯云集群 -azure 微软云集群 -aws 亚马逊集群 -self_built_hw 华为云自建集群 -self_built_idc IDC自建集群 最小长度：**0** 最大长度：**255**
risky	Boolean	是否有风险
low_risk	Integer	低危风险数量 最小值：**0** 最大值：**2147483647**
medium_risk	Integer	中危风险数量 最小值：**0** 最大值：**2147483647**
high_risk	Integer	高危风险数量 最小值：**0** 最大值：**2147483647**
fatal_risk	Integer	致命风险数量 最小值：**0** 最大值：**2147483647**

请求示例

查询前10条容器名称为install-agent-ds的容器列表

```
GET https://{endpoint}/v5/{project_id}/container/kubernetes?
offset=0&limit=10&enterprise_project_id=all_granted_eps&contianer_name=install-agent-ds
```

响应示例

状态码： 200

容器基本信息列表

```
{
  "total_num" : 1,
  "last_update_time" : 1710097200019,
  "data_list" : [ {
    "container_id" : "d7354abc9c18c68ab3e48e3481624125e415766b6d03eefc5770ff70c5391c8d",
    "container_name" : "node-exporter",
    "image_name" : "official/node-exporter:3.9.5",
    "status" : "Running",
    "create_time" : 1708453039000,
    "cpu_limit" : "500m",
    "memory_limit" : "1Gi",
    "restart_count" : 0,
    "pod_name" : "node-exporter-l4m75",
    "cluster_name" : "glz-hss",
    "cluster_id" : "352f4ef1-ce57-11ee-8cb3-0255ac100b0f",
    "cluster_type" : "cce",
    "risky" : false,
    "low_risk" : 0,
    "medium_risk" : 0,
    "high_risk" : 0,
    "fatal_risk" : 0
  } ]
}
```

状态码

状态码	描述
200	容器基本信息列表

错误码

请参见**错误码**。

3.6 事件管理

3.6.1 查询已拦截 IP 列表

功能介绍

查询已拦截IP列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/event/blocked-ip

表 3-208 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-209 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
last_days	否	Integer	查询时间范围天数，与自定义查询时间begin_time，end_time互斥 最小值：**1** 最大值：**30**
host_name	否	String	服务器名称 最小长度：**1** 最大长度：**64**
src_ip	否	String	攻击源IP
intercept_status	否	String	拦截状态，包含如下: • intercepted : 已拦截 • canceled : 已解除拦截 • cancelling : 待解除拦截 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**1000** 缺省值：**10**

请求参数

表 3-210 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-211 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数
data_list	Array of **BlockedIpResponseInfo** objects	已拦截IP详情 数组长度：**0 - 100**

表 3-212 BlockedIpResponseInfo

参数	参数类型	描述
host_id	String	主机ID
host_name	String	服务器名称
src_ip	String	攻击源IP
login_type	String	登录类型，包含如下: ● "mysql" # mysql服务 ● "rdp" # rdp服务服务 ● "ssh" # ssh服务 ● "vsftp" # vsftp服务
intercept_num	Integer	拦截次数 最小值：**0** 最大值：**2147483647**
intercept_status	String	拦截状态，包含如下: ● "intercepted" # 已拦截 ● "canceled" # 已解除拦截 ● "cancelling" # 待解除拦截
block_time	Long	开始拦截时间，毫秒 最小值：**0** 最大值：**9223372036854775807**
latest_time	Long	最近拦截时间，毫秒 最小值：**0** 最大值：**9223372036854775807**

请求示例

查询前10条已拦截的IP列表

```
GET https://{endpoint}/v5/{project_id}/event/blocked-ip?limit=10&offset=0&enterprise_project_id=xxx
```

响应示例

状态码： 200

已拦截IP列表

```
{
  "data_list" : [ {
    "block_time" : 1698715135407,
    "host_id" : "1c62fe52-0c84-4ee4-8dba-d892c5ad0ab0",
    "host_name" : "dfx-a00607964-0011",
    "intercept_num" : 230,
    "intercept_status" : "canceled",
    "latest_time" : 1698715296786,
```

```
    "login_type" : "ssh",
    "src_ip" : "100.85.239.180"
  } ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListBlockedIpSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListBlockedIpRequest request = new ListBlockedIpRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLastDays(<last_days>);
        request.withHostName("<host_name>");
        request.withSrcIp("<src_ip>");
        request.withInterceptStatus("<intercept_status>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListBlockedIpResponse response = client.listBlockedIp(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListBlockedIpRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.last_days = <last_days>
        request.host_name = "<host_name>"
        request.src_ip = "<src_ip>"
        request.intercept_status = "<intercept_status>"
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_blocked_ip(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

```
    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListBlockedIpRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    lastDaysRequest:= int32(<last_days>)
    request.LastDays = &lastDaysRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    srcIpRequest:= "<src_ip>"
    request.SrcIp = &srcIpRequest
    interceptStatusRequest:= "<intercept_status>"
    request.InterceptStatus = &interceptStatusRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListBlockedIp(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	已拦截IP列表

错误码

请参见**错误码**。

3.6.2 解除已拦截 IP

功能介绍

解除已拦截IP

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/event/blocked-ip

表 3-213 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-214 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**

请求参数

表 3-215 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-216 请求 Body 参数

参数	是否必选	参数类型	描述
data_list	否	Array of **BlockedIpRequestInfo** objects	需要解除拦截的IP列表 数组长度：**1 - 100**

表 3-217 BlockedIpRequestInfo

参数	是否必选	参数类型	描述
host_id	是	String	主机ID
src_ip	是	String	攻击源IP
login_type	是	String	登录类型，包含如下: ● "mysql" # mysql服务 ● "rdp" # rdp服务服务 ● "ssh" # ssh服务 ● "vsftp" # vsftp服务

响应参数

无

请求示例

将以SSH方式登录主机af423efds-214432fgsdaf-gfdsaggbvf的被拦截ip192.168.1.6从已拦截IP列表中解除

```
PUT https://{endpoint}/v5/{project_id}/event/blocked-ip

{
  "data_list" : [ {
    "host_id" : "af423efds-214432fgsdaf-gfdsaggbvf",
    "src_ip" : "192.168.1.6",
    "login_type" : "ssh"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

将以SSH方式登录主机af423efds-214432fgsdaf-gfdsaggbvf的被拦截ip192.168.1.6从已拦截IP列表中解除

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;
```

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

public class ChangeBlockedIpSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeBlockedIpRequest request = new ChangeBlockedIpRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ChangeBlockedIpRequestInfo body = new ChangeBlockedIpRequestInfo();
        List<BlockedIpRequestInfo> listbodyDataList = new ArrayList<>();
        listbodyDataList.add(
            new BlockedIpRequestInfo()
                .withHostId("af423efds-214432fgsdaf-gfdsaggbvf")
                .withSrcIp("192.168.1.6")
                .withLoginType("ssh")
        );
        body.withDataList(listbodyDataList);
        request.withBody(body);
        try {
            ChangeBlockedIpResponse response = client.changeBlockedIp(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

将以SSH方式登录主机af423efds-214432fgsdaf-gfdsaggbvf的被拦截ip192.168.1.6从已拦截IP列表中解除

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeBlockedIpRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listDataListbody = [
            BlockedIpRequestInfo(
                host_id="af423efds-214432fgsdaf-gfdsaggbvf",
                src_ip="192.168.1.6",
                login_type="ssh"
            )
        ]
        request.body = ChangeBlockedIpRequestInfo(
            data_list=listDataListbody
        )
        response = client.change_blocked_ip(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

将以SSH方式登录主机af423efds-214432fgsdaf-gfdsaggbvf的被拦截ip192.168.1.6从已拦截IP列表中解除

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeBlockedIpRequest{}
```

```
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listDataListbody = []model.BlockedIpRequestInfo{
        {
            HostId: "af423efds-214432fgsdaf-gfdsaggbvf",
            SrcIp: "192.168.1.6",
            LoginType: "ssh",
        },
    }
    request.Body = &model.ChangeBlockedIpRequestInfo{
        DataList: &listDataListbody,
    }
    response, err := client.ChangeBlockedIp(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	successful response

错误码

请参见**错误码**。

3.6.3 查询已隔离文件列表

功能介绍

查询已隔离文件列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/event/isolated-file

表 3-218 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-219 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
file_path	否	String	文件路径
host_name	否	String	服务器名称 最小长度：**1** 最大长度：**64**
private_ip	否	String	服务器私有IP 最小长度：**1** 最大长度：**256**
public_ip	否	String	服务器公网IP 最小长度：**1** 最大长度：**256**
file_hash	否	String	文件hash,当前为sha256
asset_value	否	String	资产重要性，包含如下3种 ● important： 重要资产 ● common： 一般资产 ● test： 测试资产 最小长度：**0** 最大长度：**128**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

参数	是否必选	参数类型	描述
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**1000** 缺省值：**10**

请求参数

表 3-220 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-221 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数
data_list	Array of **IsolatedFileResponseInfo** objects	已隔离文件详情 数组长度：**0 - 100**

表 3-222 IsolatedFileResponseInfo

参数	参数类型	描述
os_type	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。
host_id	String	主机ID
host_name	String	服务器名称
file_hash	String	文件哈希
file_path	String	文件路径
file_attr	String	文件属性
isolation_status	String	隔离状态，包含如下: ● isolated : 已隔离 ● restored : 已恢复 ● isolating : 已下发隔离任务 ● restoring : 已下发恢复任务
private_ip	String	服务器私有IP
public_ip	String	弹性公网IP地址
asset_value	String	资产重要性
update_time	Integer	更新时间，毫秒
agent_version	String	agent版本
isolate_source	String	隔离来源，包含如下: ● event : 安全告警事件 ● antivirus : 病毒查杀
event_name	String	事件名称
agent_event_info	**IsolateEventResponseInfo** object	隔离事件详情
antivirus_result_info	**AntivirusResultDetailInfo** object	病毒查杀结果详情

表 3-223 IsolateEventResponseInfo

参数	参数类型	描述
event_id	String	事件编号

参数	参数类型	描述
event_class_id	String	事件分类，包含如下: ● container_1001：容器命名空间 ● container_1002：容器开放端口 ● container_1003：容器安全选项 ● container_1004：容器挂载目录 ● containerescape_0001：容器高危系统调用 ● containerescape_0002：Shocker攻击 ● containerescape_0003：DirtCow攻击 ● containerescape_0004：容器文件逃逸攻击 ● dockerfile_001：用户自定义容器保护文件被修改 ● dockerfile_002：容器文件系统可执行文件被修改 ● dockerproc_001：容器进程异常事件上报 ● fileprotect_0001：文件提权 ● fileprotect_0002：关键文件变更 ● fileprotect_0003：关键文件路径变更 ● fileprotect_0004：文件/目录变更 ● av_1002：病毒 ● av_1003：蠕虫 ● av_1004：木马 ● av_1005：僵尸网络 ● av_1006：后门 ● av_1007：间谍软件 ● av_1008：恶意广告软件 ● av_1009：钓鱼 ● av_1010：Rootkit ● av_1011：勒索软件 ● av_1012：黑客工具 ● av_1013：灰色软件 ● av_1015：Webshell ● av_1016：挖矿软件 ● login_0001：尝试暴力破解 ● login_0002：爆破成功 ● login_1001：登录成功 ● login_1002：异地登录 ● login_1003：弱口令 ● malware_0001：shell变更事件上报

参数	参数类型	描述
		• malware_0002：反弹shell事件上报 • malware_1001：恶意程序 • procdet_0001：进程异常行为检测 • procdet_0002：进程提权 • procreport_0001：危险命令 • user_1001：账号变更 • user_1002：风险账号 • vmescape_0001：虚拟机敏感命令执行 • vmescape_0002：虚拟化进程访问敏感文件 • vmescape_0003：虚拟机异常端口访问 • webshell_0001：网站后门 • network_1001：恶意挖矿 • network_1002：对外DDoS攻击 • network_1003：恶意扫描 • network_1004：敏感区域攻击 • ransomware_0001：勒索攻击 • ransomware_0002：勒索攻击 • ransomware_0003：勒索攻击 • fileless_0001：进程注入 • fileless_0002：动态库注入进程 • fileless_0003：关键配置变更 • fileless_0004：环境变量变更 • fileless_0005：内存文件进程 • fileless_0006：vdso劫持 • crontab_1001：Crontab可疑任务 • vul_exploit_0001：Redis漏洞利用攻击 • vul_exploit_0002：Hadoop漏洞利用攻击 • vul_exploit_0003：MySQL漏洞利用攻击 • rootkit_0001：可疑rootkit文件 • rootkit_0002：可疑内核模块 • RASP_0004：上传Webshell • RASP_0018：无文件Webshell • blockexec_001：已知勒索攻击 • hips_0001：Windows Defender防护被禁用 • hips_0002：可疑的黑客工具 • hips_0003：可疑的勒索加密行为 • hips_0004：隐藏账号创建

参数	参数类型	描述
		• hips_0005：读取用户密码凭据 • hips_0006：可疑的SAM文件导出 • hips_0007：可疑shadow copy删除操作 • hips_0008：备份文件删除 • hips_0009：可疑勒索病毒操作注册表 • hips_0010：可疑的异常进程行为 • hips_0011：可疑的扫描探测 • hips_0012：可疑的勒索病毒脚本运行 • hips_0013：可疑的挖矿命令执行 • hips_0014：可疑的禁用windows安全中心 • hips_0015：可疑的停止防火墙服务行为 • hips_0016：可疑的系统自动恢复禁用 • hips_0017：Offies 创建可执行文件 • hips_0018：带宏Offies文件异常创建 • hips_0019：可疑的注册表操作 • hips_0020：Confluence远程代码执行 • hips_0021：MSDT远程代码执行 • portscan_0001：通用端口扫描 • portscan_0002：秘密端口扫描 • k8s_1001：Kubernetes事件删除 • k8s_1002：创建特权Pod • k8s_1003：Pod中使用交互式shell • k8s_1004：创建敏感目录Pod • k8s_1005：创建主机网络的Pod • k8s_1006：创建主机Pid空间的Pod • k8s_1007：普通pod访问APIserver认证失败 • k8s_1008：普通Pod通过Curl访问APIServer • k8s_1009：系统管理空间执行exec • k8s_1010：系统管理空间创建Pod • k8s_1011：创建静态Pod • k8s_1012：创建DaemonSet • k8s_1013：创建集群计划任务 • k8s_1014：Secrets操作 • k8s_1015：枚举用户可执行的操作 • k8s_1016：高权限RoleBinding或ClusterRoleBinding • k8s_1017：ServiceAccount创建

参数	参数类型	描述
		• k8s_1018 : 创建Cronjob • k8s_1019 : Pod中exec使用交互式shell • k8s_1020 : 无权限访问Apiserver • k8s_1021 : 使用curl访问APIServer • k8s_1022 : Ingress漏洞 • k8s_1023 : 中间人攻击 • k8s_1024 : 蠕虫挖矿木马 • k8s_1025 : K8s事件删除 • k8s_1026 : SelfSubjectRulesReview场景 • imgblock_0001 : 镜像白名单阻断 • imgblock_0002 : 镜像黑名单阻断 • imgblock_0003 : 镜像标签白名单阻断 • imgblock_0004 : 镜像标签黑名单阻断 • imgblock_0005 : 创建容器白名单阻断 • imgblock_0006 : 创建容器黑名单阻断 • imgblock_0007 : 容器mount proc阻断 • imgblock_0008 : 容器seccomp unconfined阻断 • imgblock_0009 : 容器特权阻断 • imgblock_0010 : 容器capabilities阻断

参数	参数类型	描述
event_type	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 ： 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 ： 系统安全防护被禁用 ● 3030 ： 备份删除 ● 3031 ： 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描 ● 6003 : 主机扫描

参数	参数类型	描述
		● 13001 : Kubernetes事件删除 ● 13002 : Pod异常行为 ● 13003 : 枚举用户信息 ● 13004 : 绑定集群用户角色
event_name	String	事件名称
severity	String	威胁等级，包含如下: ● Security : 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 ● Critical : 危急
container_name	String	容器实例名称，只有容器类型的告警有
image_name	String	镜像名称，只有容器类型的告警有
host_name	String	服务器名称
host_id	String	主机ID
private_ip	String	服务器私有IP
public_ip	String	弹性公网IP地址
os_type	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。
host_status	String	服务器状态，包含如下4种。 ● ACTIVE： 运行中。 ● SHUTOFF： 关机。 ● BUILDING： 创建中。 ● ERROR： 故障。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
agent_status	String	Agent状态，包含如下5种。 ● installed： 已安装。 ● not_installed： 未安装。 ● online： 在线。 ● offline： 离线。 ● install_failed： 安装失败。 ● installing： 安装中。 最小长度：**1** 最大长度：**32**
protect_status	String	防护状态，包含如下2种。 ● closed： 未防护。 ● opened： 防护中。 最小长度：**1** 最大长度：**32**
asset_value	String	资产重要性，包含如下4种 ● important： 重要资产 ● common： 一般资产 ● test： 测试资产 最小长度：**0** 最大长度：**128**
attack_phase	String	攻击阶段，包含如下： ● reconnaissance : 侦查跟踪 ● weaponization : 武器构建 ● delivery : 载荷投递 ● exploit : 漏洞利用 ● installation : 安装植入 ● command_and_control : 命令与控制 ● actions : 目标达成
attack_tag	String	攻击标识，包含如下： ● attack_success : 攻击成功 ● attack_attempt : 攻击尝试 ● attack_blocked : 攻击被阻断 ● abnormal_behavior : 异常行为 ● collapsible_host : 主机失陷 ● system_vulnerability : 系统脆弱性
occur_time	Integer	发生时间，毫秒

参数	参数类型	描述
handle_time	Integer	处理时间，毫秒，已处理的告警才有
handle_status	String	处理状态，包含如下: ● unhandled ： 未处理 ● handled : 已处理
handle_method	String	处理方式，已处理的告警才有，包含如下: ● mark_as_handled : 手动处理 ● ignore : 忽略 ● add_to_alarm_whitelist : 加入告警白名单 ● add_to_login_whitelist : 加入登录白名单 ● isolate_and_kill : 隔离查杀
handler	String	备注信息，已处理的告警才有
recommendation	String	处置建议
description	String	告警说明 最小长度：**0** 最大长度：**1024**
event_abstract	String	告警摘要 最小长度：**0** 最大长度：**512**
event_count	Integer	事件发生次数 最小值：**0** 最大值：**2147483647**

表 3-224 AntivirusResultDetailInfo

参数	参数类型	描述
result_id	String	病毒查杀结果ID
malware_name	String	病毒名称
file_path	String	文件路径
file_hash	String	文件哈希
file_size	Integer	文件大小
file_owner	String	文件属主
file_attr	String	文件属性

参数	参数类型	描述
file_ctime	Integer	文件创建时间
file_mtime	Integer	文件更新时间
update_time	Integer	更新时间，毫秒
agent_id	String	Agent ID

请求示例

查询前10条已隔离的文件列表

```
GET https://{endpoint}/v5/{project_id}/event/isolated-file?limit=10&offset=0&enterprise_project_id=xxx
```

响应示例

状态码： 200

已隔离文件列表

```
{
  "data_list" : [ {
    "file_attr" : "0",
    "file_hash" : "58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde",
    "file_path" : "C:\\Users\\Public\\Public Docker\\system32.exe",
    "os_type" : "Linux",
    "host_id" : "5a41ca47-8ea7-4a65-a8fb-950d03d8638e",
    "host_name" : "ecs-wi-800211",
    "isolation_status" : "isolated",
    "private_ip" : "127.0.0.2",
    "public_ip" : "127.0.0.1",
    "asset_value" : "common",
    "update_time" : 1698304933717,
    "agent_version" : "3.2.10",
    "isolate_source" : "event",
    "event_name" : "间谍软件",
    "antivirus_result_info" : {
      "result_id" : "5a41ca47-8ea7-4a65-a8fb-950d03d8638e",
      "malware_name" : "Win32.Virus.Hidrag",
      "file_attr" : "0",
      "file_hash" : "58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde",
      "file_path" : "C:\\Users\\Public\\Public Docker\\system32.exe",
      "file_size" : 58460,
      "file_owner" : "Administrators",
      "file_ctime" : 1700039800,
      "file_mtime" : 1700039800,
      "update_time" : 1698304933717,
      "agent_id" : "5a41ca47-8ea7-4a65-a8fb-950d03d8638e"
    },
    "agent_event_info" : {
      "attack_phase" : "exploit",
      "attack_tag" : "abnormal_behavior",
      "event_class_id" : "lgin_1002",
      "event_id" : "d8a12cf7-6a43-4cd6-92b4-aabf1e917",
      "event_name" : "different locations",
      "event_type" : 4004,
      "handle_status" : "unhandled",
      "host_name" : "xxx",
      "occur_time" : 1661593036627,
      "private_ip" : "127.0.0.1",
      "severity" : "Medium",
```

```
    "os_type" : "Linux",
    "agent_status" : "online",
    "asset_value" : "common",
    "protect_status" : "opened",
    "host_status" : "ACTIVE",
    "description" : "",
    "event_abstract" : "",
    "image_name" : "image",
    "container_name" : "test",
    "host_id" : "5a41ca47-8ea7-4a65-a8fb-950d03d8638e",
    "public_ip" : "127.0.0.2",
    "handle_time" : 1698304933717,
    "handle_method" : "ignore",
    "recommendation" : "建议处置",
    "event_count" : 1
   }
 } ],
 "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListIsolatedFileSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListIsolatedFileRequest request = new ListIsolatedFileRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withFilePath("<file_path>");
        request.withHostName("<host_name>");
        request.withPrivateIp("<private_ip>");
        request.withPublicIp("<public_ip>");
        request.withFileHash("<file_hash>");
        request.withAssetValue("<asset_value>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
```

```
            ListIsolatedFileResponse response = client.listIsolatedFile(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListIsolatedFileRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.file_path = "<file_path>"
        request.host_name = "<host_name>"
        request.private_ip = "<private_ip>"
        request.public_ip = "<public_ip>"
        request.file_hash = "<file_hash>"
        request.asset_value = "<asset_value>"
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_isolated_file(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
```

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListIsolatedFileRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    filePathRequest:= "<file_path>"
    request.FilePath = &filePathRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
    publicIpRequest:= "<public_ip>"
    request.PublicIp = &publicIpRequest
    fileHashRequest:= "<file_hash>"
    request.FileHash = &fileHashRequest
    assetValueRequest:= "<asset_value>"
    request.AssetValue = &assetValueRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListIsolatedFile(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	已隔离文件列表

错误码

请参见**错误码**。

3.6.4 恢复已隔离文件

功能介绍

恢复已隔离文件

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/event/isolated-file

表 3-225 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-226 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**

请求参数

表 3-227 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） 最小长度：**1** 最大长度：**32768**

参数	是否必选	参数类型	描述
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-228 请求 Body 参数

参数	是否必选	参数类型	描述
data_list	否	Array of **IsolatedFileRequestInfo** objects	需要恢复的文件列表 数组长度：**0 - 100**

表 3-229 IsolatedFileRequestInfo

参数	是否必选	参数类型	描述
host_id	否	String	主机ID
file_hash	否	String	文件哈希
file_path	否	String	文件路径
file_attr	否	String	文件属性

响应参数

无

请求示例

将主机5a41ca47-8ea7-4a65-a8fb-950d03d8638e已被隔离的文件C:\Users\Public\test.exe从隔离文件中恢复

```
PUT https://{endpoint}/v5/{project_id}/event/isolated-file

{
  "data_list" : [ {
    "file_attr" : "0",
    "file_hash" : "58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde",
    "file_path" : "C:\\Users\\Public\\test.exe",
    "host_id" : "5a41ca47-8ea7-4a65-a8fb-950d03d8638e"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

将主机5a41ca47-8ea7-4a65-a8fb-950d03d8638e已被隔离的文件C:\Users\Public\test.exe从隔离文件中恢复

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeIsolatedFileSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeIsolatedFileRequest request = new ChangeIsolatedFileRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ChangeIsolatedFileRequestInfo body = new ChangeIsolatedFileRequestInfo();
        List<IsolatedFileRequestInfo> listbodyDataList = new ArrayList<>();
        listbodyDataList.add(
            new IsolatedFileRequestInfo()
                .withHostId("5a41ca47-8ea7-4a65-a8fb-950d03d8638e")
                .withFileHash("58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde")
                .withFilePath("C:\Users\Public\test.exe")
                .withFileAttr("0")
        );
        body.withDataList(listbodyDataList);
        request.withBody(body);
        try {
            ChangeIsolatedFileResponse response = client.changeIsolatedFile(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

将主机5a41ca47-8ea7-4a65-a8fb-950d03d8638e已被隔离的文件C:\Users\Public\test.exe从隔离文件中恢复

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeIsolatedFileRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listDataListbody = [
            IsolatedFileRequestInfo(
                host_id="5a41ca47-8ea7-4a65-a8fb-950d03d8638e",
                file_hash="58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde",
                file_path="C:\Users\Public\test.exe",
                file_attr="0"
            )
        ]
        request.body = ChangeIsolatedFileRequestInfo(
            data_list=listDataListbody
        )
        response = client.change_isolated_file(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

将主机5a41ca47-8ea7-4a65-a8fb-950d03d8638e已被隔离的文件C:\Users\Public\test.exe从隔离文件中恢复

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeIsolatedFileRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdDataList:= "5a41ca47-8ea7-4a65-a8fb-950d03d8638e"
    fileHashDataList:= "58693382bc0c9f60ef86e5b37cf3c2f3a9c9ec46936901eaa9131f7ee4a09bde"
    filePathDataList:= "C:\Users\Public\test.exe"
    fileAttrDataList:= "0"
    var listDataListbody = []model.IsolatedFileRequestInfo{
        {
            HostId: &hostIdDataList,
            FileHash: &fileHashDataList,
            FilePath: &filePathDataList,
            FileAttr: &fileAttrDataList,
        },
    }
    request.Body = &model.ChangeIsolatedFileRequestInfo{
        DataList: &listDataListbody,
    }
    response, err := client.ChangeIsolatedFile(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	successful response

错误码

请参见**错误码**。

3.7 入侵检测

3.7.1 处理告警事件

功能介绍

处理告警事件

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/event/operate

表 3-230 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-231 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
container_name	否	String	容器实例名称
container_id	否	String	容器Id

请求参数

表 3-232 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-233 请求 Body 参数

参数	是否必选	参数类型	描述
operate_type	是	String	处理方式，包含如下: • mark_as_handled : 手动处理 • ignore : 忽略 • add_to_alarm_whitelist : 加入告警白名单 • add_to_login_whitelist : 加入登录白名单 • isolate_and_kill : 隔离查杀 • unhandle : 取消手动处理 • do_not_ignore : 取消忽略 • remove_from_alarm_whitelist : 删除告警白名单 • remove_from_login_whitelist : 删除登录白名单 • do_not_isolate_or_kill : 取消隔离查杀
handler	否	String	备注信息，已处理的告警才有
operate_event_list	是	Array of **OperateEventRequestInfo** objects	操作的事件列表 数组长度：**0 - 100**

参数	是否必选	参数类型	描述
event_white_rule_list	否	Array of **EventWhiteRuleListRequestInfo** objects	用户自定义告警白名单规则列表 数组长度：**0 - 100**

表 3-234 OperateEventRequestInfo

参数	是否必选	参数类型	描述
event_class_id	是	String	事件分类，包含如下: ● container_1001：容器命名空间 ● container_1002：容器开放端口 ● container_1003：容器安全选项 ● container_1004：容器挂载目录 ● containerescape_0001：容器高危系统调用 ● containerescape_0002：Shocker攻击 ● containerescape_0003：DirtCow攻击 ● containerescape_0004：容器文件逃逸攻击 ● dockerfile_001：用户自定义容器保护文件被修改 ● dockerfile_002：容器文件系统可执行文件被修改 ● dockerproc_001：容器进程异常事件上报 ● fileprotect_0001：文件提权 ● fileprotect_0002：关键文件变更 ● fileprotect_0003：关键文件路径变更 ● fileprotect_0004：文件/目录变更 ● av_1002：病毒 ● av_1003：蠕虫 ● av_1004：木马 ● av_1005：僵尸网络 ● av_1006：后门 ● av_1007：间谍软件 ● av_1008：恶意广告软件 ● av_1009：钓鱼 ● av_1010：Rootkit ● av_1011：勒索软件

参数	是否必选	参数类型	描述
			• av_1012 : 黑客工具 • av_1013 : 灰色软件 • av_1015 : Webshell • av_1016 : 挖矿软件 • login_0001 : 尝试暴力破解 • login_0002 : 爆破成功 • login_1001 : 登录成功 • login_1002 : 异地登录 • login_1003 : 弱口令 • malware_0001 : shell变更事件上报 • malware_0002 : 反弹shell事件上报 • malware_1001 : 恶意程序 • procdet_0001 : 进程异常行为检测 • procdet_0002 : 进程提权 • procreport_0001 : 危险命令 • user_1001 : 账号变更 • user_1002 : 风险账号 • vmescape_0001 : 虚拟机敏感命令执行 • vmescape_0002 : 虚拟化进程访问敏感文件 • vmescape_0003 : 虚拟机异常端口访问 • webshell_0001 : 网站后门 • network_1001 : 恶意挖矿 • network_1002 : 对外DDoS攻击 • network_1003 : 恶意扫描 • network_1004 : 敏感区域攻击 • ransomware_0001 : 勒索攻击 • ransomware_0002 : 勒索攻击 • ransomware_0003 : 勒索攻击 • fileless_0001 : 进程注入

参数	是否必选	参数类型	描述
			• fileless_0002：动态库注入进程 • fileless_0003：关键配置变更 • fileless_0004：环境变量变更 • fileless_0005：内存文件进程 • fileless_0006：vdso劫持 • crontab_1001：Crontab可疑任务 • vul_exploit_0001：Redis漏洞利用攻击 • vul_exploit_0002：Hadoop漏洞利用攻击 • vul_exploit_0003：MySQL漏洞利用攻击 • rootkit_0001：可疑rootkit文件 • rootkit_0002：可疑内核模块 • RASP_0004：上传Webshell • RASP_0018：无文件Webshell • blockexec_001：已知勒索攻击 • hips_0001：Windows Defender防护被禁用 • hips_0002：可疑的黑客工具 • hips_0003：可疑的勒索加密行为 • hips_0004：隐藏账号创建 • hips_0005：读取用户密码凭据 • hips_0006：可疑的SAM文件导出 • hips_0007：可疑shadow copy删除操作 • hips_0008：备份文件删除 • hips_0009：可疑勒索病毒操作注册表 • hips_0010：可疑的异常进程行为 • hips_0011：可疑的扫描探测

参数	是否必选	参数类型	描述
			• hips_0012 : 可疑的勒索病毒脚本运行 • hips_0013 : 可疑的挖矿命令执行 • hips_0014 : 可疑的禁用windows安全中心 • hips_0015 : 可疑的停止防火墙服务行为 • hips_0016 : 可疑的系统自动恢复禁用 • hips_0017 : Offies 创建可执行文件 • hips_0018 : 带宏Offies文件异常创建 • hips_0019 : 可疑的注册表操作 • hips_0020 : Confluence远程代码执行 • hips_0021 : MSDT远程代码执行 • portscan_0001 : 通用端口扫描 • portscan_0002 : 秘密端口扫描 • k8s_1001 : Kubernetes事件删除 • k8s_1002 : 创建特权Pod • k8s_1003 : Pod中使用交互式shell • k8s_1004 : 创建敏感目录Pod • k8s_1005 : 创建主机网络的Pod • k8s_1006 : 创建主机Pid空间的Pod • k8s_1007 : 普通pod访问APIserver认证失败 • k8s_1008 : 普通Pod通过Curl访问APIServer • k8s_1009 : 系统管理空间执行exec • k8s_1010 : 系统管理空间创建Pod

参数	是否必选	参数类型	描述
			• k8s_1011：创建静态Pod • k8s_1012：创建DaemonSet • k8s_1013：创建集群计划任务 • k8s_1014：Secrets操作 • k8s_1015：枚举用户可执行的操作 • k8s_1016：高权限RoleBinding或ClusterRoleBinding • k8s_1017：ServiceAccount创建 • k8s_1018：创建Cronjob • k8s_1019：Pod中exec使用交互式shell • k8s_1020：无权限访问Apiserver • k8s_1021：使用curl访问APIServer • k8s_1022：Ingress漏洞 • k8s_1023：中间人攻击 • k8s_1024：蠕虫挖矿木马 • k8s_1025：K8s事件删除 • k8s_1026：SelfSubjectRulesReview场景 • imgblock_0001：镜像白名单阻断 • imgblock_0002：镜像黑名单阻断 • imgblock_0003：镜像标签白名单阻断 • imgblock_0004：镜像标签黑名单阻断 • imgblock_0005：创建容器白名单阻断 • imgblock_0006：创建容器黑名单阻断 • imgblock_0007：容器mount proc阻断 • imgblock_0008：容器seccomp unconfined阻断

参数	是否必选	参数类型	描述
			• imgblock_0009：容器特权阻断 • imgblock_0010：容器capabilities阻断
event_id	是	String	事件编号

参数	是否必选	参数类型	描述
event_type	是	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 : 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 : 系统安全防护被禁用 ● 3030 : 备份删除 ● 3031 : 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描 ● 6003 : 主机扫描

参数	是否必选	参数类型	描述
			• 13001：Kubernetes事件删除 • 13002：Pod异常行为 • 13003：枚举用户信息 • 13004：绑定集群用户角色
occur_time	是	Integer	发生时间，毫秒
operate_detail_list	是	Array of **EventDetailRequestInfo** objects	操作详情信息列表，当operate_type 为add_to_alarm_whitelist 或remove_from_alarm_whitelist时，必传 keyword 和 hash；当operate_type 为add_to_login_whitelist 或remove_from_login_whitelist时，必传 login_ip， private_ip和 login_user_name；当operate_type 为isolate_and_kill 或do_not_isolate_or_kill 时，必传agent_id，file_hash，file_path，process_pid；其余情况可不填写内容。 数组长度：**0 - 100**

表 3-235 EventDetailRequestInfo

参数	是否必选	参数类型	描述
agent_id	否	String	Agent ID
process_pid	否	Integer	进程id
file_hash	否	String	文件哈希
file_path	否	String	文件路径
file_attr	否	String	文件属性
keyword	否	String	告警事件关键字，仅用于告警白名单
hash	否	String	告警事件hash，仅用于告警白名单
private_ip	否	String	服务器私有IP
login_ip	否	String	登录源IP

参数	是否必选	参数类型	描述
login_user_name	否	String	登录用户名
container_id	否	String	容器ID 最小长度：**64** 最大长度：**64**
container_name	否	String	容器名称 最小长度：**1** 最大长度：**128**

表 3-236 EventWhiteRuleListRequestInfo

参数	是否必选	参数类型	描述
event_type	是	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 : 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 : 系统安全防护被禁用 ● 3030 : 备份删除 ● 3031 : 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描

参数	是否必选	参数类型	描述
			• 6003：主机扫描 • 13001：Kubernetes事件删除 • 13002：Pod异常行为 • 13003：枚举用户信息 • 13004：绑定集群用户角色
field_key	是	String	加白字段，包含如下: • "file/process hash" # 进程/文件hash • "file_path" # 文件路径 • "process_path" # 进程路径 • "login_ip" # 登录ip • "reg_key" #注册表key • "process_cmdline" # 进程命令行 • "username" # 用户名 最小长度：**1** 最大长度：**20**
field_value	是	String	加白字段值 最小长度：**1** 最大长度：**128**
judge_type	是	String	通配符，包含如下: • "equal" # 相等 • "contain" # 包含 最小长度：**1** 最大长度：**10**

响应参数

无

请求示例

手动处理告警事件类型为Rootkit、告警事件编号为2a71e1e2-60f4-4d56-b314-2038fdc39de6的入侵告警事件。

```
POST https://{endpoint}/v5/{project_id}/event/operate?enterprise_project_id=xxx

{
  "operate_type" : "mark_as_handled",
  "handler" : "test",
  "operate_event_list" : [ {
```

```
  "event_class_id" : "rootkit_0001",
  "event_id" : "2a71e1e2-60f4-4d56-b314-2038fdc39de6",
  "occur_time" : 1672046760353,
  "event_type" : 1010,
  "operate_detail_list" : [ {
    "agent_id" : "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8",
    "file_hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "file_path" : "/usr/test",
    "process_pid" : 3123,
    "file_attr" : 33261,
    "keyword" : "file_path=/usr/test",
    "hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "login_ip" : "127.0.0.1",
    "private_ip" : "127.0.0.2",
    "login_user_name" : "root",
    "container_id" : "containerid",
    "container_name" : "/test"
  } ]
 } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

手动处理告警事件类型为Rootkit、告警事件编号为2a71e1e2-60f4-4d56-b314-2038fdc39de6的入侵告警事件。

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

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeEventSolution {

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

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

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

```
            .build();
        ChangeEventRequest request = new ChangeEventRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withContainerName("<container_name>");
        request.withContainerId("<container_id>");
        ChangeEventRequestInfo body = new ChangeEventRequestInfo();
        List<EventDetailRequestInfo> listOperateEventListOperateDetailList = new ArrayList<>();
        listOperateEventListOperateDetailList.add(
            new EventDetailRequestInfo()
                .withAgentId("c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8")
                .withProcessPid(3123)
                .withFileHash("e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d")
                .withFilePath("/usr/test")
                .withFileAttr("33261")
                .withKeyword("file_path=/usr/test")
                .withHash("e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d")
                .withPrivateIp("127.0.0.2")
                .withLoginIp("127.0.0.1")
                .withLoginUserName("root")
                .withContainerId("containerid")
                .withContainerName("/test")
        );
        List<OperateEventRequestInfo> listbodyOperateEventList = new ArrayList<>();
        listbodyOperateEventList.add(
            new OperateEventRequestInfo()
                .withEventClassId("rootkit_0001")
                .withEventId("2a71e1e2-60f4-4d56-b314-2038fdc39de6")
                .withEventType(1010)
                .withOccurTime(1672046760353L)
                .withOperateDetailList(listOperateEventListOperateDetailList)
        );
        body.withOperateEventList(listbodyOperateEventList);
        body.withHandler("test");
        body.withOperateType("mark_as_handled");
        request.withBody(body);
        try {
            ChangeEventResponse response = client.changeEvent(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

手动处理告警事件类型为Rootkit、告警事件编号为2a71e1e2-60f4-4d56-b314-2038fdc39de6的入侵告警事件。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeEventRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.container_name = "<container_name>"
        request.container_id = "<container_id>"
        listOperateDetailListOperateEventList = [
            EventDetailRequestInfo(
                agent_id="c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8",
                process_pid=3123,
                file_hash="e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
                file_path="/usr/test",
                file_attr="33261",
                keyword="file_path=/usr/test",
                hash="e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
                private_ip="127.0.0.2",
                login_ip="127.0.0.1",
                login_user_name="root",
                container_id="containerid",
                container_name="/test"
            )
        ]
        listOperateEventListbody = [
            OperateEventRequestInfo(
                event_class_id="rootkit_0001",
                event_id="2a71e1e2-60f4-4d56-b314-2038fdc39de6",
                event_type=1010,
                occur_time=1672046760353,
                operate_detail_list=listOperateDetailListOperateEventList
            )
        ]
        request.body = ChangeEventRequestInfo(
            operate_event_list=listOperateEventListbody,
            handler="test",
            operate_type="mark_as_handled"
        )
        response = client.change_event(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

手动处理告警事件类型为Rootkit、告警事件编号为2a71e1e2-60f4-4d56-b314-2038fdc39de6的入侵告警事件。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeEventRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    containerNameRequest:= "<container_name>"
    request.ContainerName = &containerNameRequest
    containerIdRequest:= "<container_id>"
    request.ContainerId = &containerIdRequest
    agentIdOperateDetailList:= "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8"
    processPidOperateDetailList:= int32(3123)
    fileHashOperateDetailList:= "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d"
    filePathOperateDetailList:= "/usr/test"
    fileAttrOperateDetailList:= "33261"
    keywordOperateDetailList:= "file_path=/usr/test"
    hashOperateDetailList:= "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d"
    privateIpOperateDetailList:= "127.0.0.2"
    loginIpOperateDetailList:= "127.0.0.1"
    loginUserNameOperateDetailList:= "root"
    containerIdOperateDetailList:= "containerid"
    containerNameOperateDetailList:= "/test"
    var listOperateDetailListOperateEventList = []model.EventDetailRequestInfo{
        {
            AgentId: &agentIdOperateDetailList,
            ProcessPid: &processPidOperateDetailList,
            FileHash: &fileHashOperateDetailList,
            FilePath: &filePathOperateDetailList,
            FileAttr: &fileAttrOperateDetailList,
            Keyword: &keywordOperateDetailList,
            Hash: &hashOperateDetailList,
            PrivateIp: &privateIpOperateDetailList,
            LoginIp: &loginIpOperateDetailList,
            LoginUserName: &loginUserNameOperateDetailList,
            ContainerId: &containerIdOperateDetailList,
            ContainerName: &containerNameOperateDetailList,
        },
    }
    var listOperateEventListbody = []model.OperateEventRequestInfo{
        {
            EventClassId: "rootkit_0001",
            EventId: "2a71e1e2-60f4-4d56-b314-2038fdc39de6",
            EventType: int32(1010),
            OccurTime: int64(1672046760353),
            OperateDetailList: listOperateDetailListOperateEventList,
        },
    }
    handlerChangeEventRequestInfo:= "test"
    request.Body = &model.ChangeEventRequestInfo{
        OperateEventList: listOperateEventListbody,
```

```
        Handler: &handlerChangeEventRequestInfo,
        OperateType: "mark_as_handled",
    }
    response, err := client.ChangeEvent(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	success
400	参数非法
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.7.2 查入侵事件列表

功能介绍

查入侵事件列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/event/events

表 3-237 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-238 Query 参数

参数	是否必选	参数类型	描述
category	是	String	事件类别，包含如下: ● host : 主机安全事件 ● container : 容器安全事件 最小长度：**0** 最大长度：**32**
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
last_days	否	Integer	查询时间范围天数，与自定义查询时间begin_time，end_time互斥 最小值：**1** 最大值：**30**
host_name	否	String	服务器名称 最小长度：**1** 最大长度：**64**
host_id	否	String	主机ID 最小长度：**0** 最大长度：**64**
private_ip	否	String	服务器私有IP 最小长度：**1** 最大长度：**256**
public_ip	否	String	服务器公网IP 最小长度：**1** 最大长度：**256**
container_name	否	String	容器实例名称

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**1000** 缺省值：**10**

参数	是否必选	参数类型	描述
event_types	否	Array	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 : 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3026 : crontab提权 ● 3027 : Crontab可疑任务 ● 3029 : 系统安全防护被禁用 ● 3030 : 备份删除 ● 3031 : 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描

参数	是否必选	参数类型	描述
			● 6003 : 主机扫描 ● 13001 : Kubernetes事件删除 ● 13002 : Pod异常行为 ● 13003 : 枚举用户信息 ● 13004 : 绑定集群用户角色 最小值：**1000** 最大值：**30000** 数组长度：**1 - 500**
handle_status	否	String	处置状态，包含如下: ● unhandled ： 未处理 ● handled : 已处理 最小长度：**1** 最大长度：**32**
severity	否	String	威胁等级，包含如下: ● Security ： 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 ● Critical : 危急 最小长度：**1** 最大长度：**32**
begin_time	否	String	自定义查询时间，与查询时间范围天数互斥，查询时间段的起始时间，毫秒级时间戳，end_time减去begin_time小于等于2天，与查询时间范围天数互斥 最小长度：**13** 最大长度：**13**
end_time	否	String	自定义时间，查询时间段的终止时间，毫秒级时间戳，end_time减去begin_time小于等于2天，与查询时间范围天数互斥 最小长度：**13** 最大长度：**13**

参数	是否必选	参数类型	描述
event_class_ids	否	Array	事件标识，包含如下: ● container_1001：容器命名空间 ● container_1002：容器开放端口 ● container_1003：容器安全选项 ● container_1004：容器挂载目录 ● containerescape_0001：容器高危系统调用 ● containerescape_0002：Shocker攻击 ● containerescape_0003：DirtCow攻击 ● containerescape_0004：容器文件逃逸攻击 ● dockerfile_001：用户自定义容器保护文件被修改 ● dockerfile_002：容器文件系统可执行文件被修改 ● dockerproc_001：容器进程异常事件上报 ● fileprotect_0001：文件提权 ● fileprotect_0002：关键文件变更 ● fileprotect_0003：关键文件路径变更 ● fileprotect_0004：文件/目录变更 ● av_1002：病毒 ● av_1003：蠕虫 ● av_1004：木马 ● av_1005：僵尸网络 ● av_1006：后门 ● av_1007：间谍软件 ● av_1008：恶意广告软件 ● av_1009：钓鱼 ● av_1010：Rootkit ● av_1011：勒索软件 ● av_1012：黑客工具

参数	是否必选	参数类型	描述
			• av_1013 : 灰色软件 • av_1015 : Webshell • av_1016 : 挖矿软件 • login_0001 : 尝试暴力破解 • login_0002 : 爆破成功 • login_1001 : 登录成功 • login_1002 : 异地登录 • login_1003 : 弱口令 • malware_0001 : shell变更事件上报 • malware_0002 : 反弹shell事件上报 • malware_1001 : 恶意程序 • procdet_0001 : 进程异常行为检测 • procdet_0002 : 进程提权 • crontab_0001 : crontab脚本提权 • crontab_0002 : 恶意路径提权 • procreport_0001 : 危险命令 • user_1001 : 账号变更 • user_1002 : 风险账号 • vmescape_0001 : 虚拟机敏感命令执行 • vmescape_0002 : 虚拟化进程访问敏感文件 • vmescape_0003 : 虚拟机异常端口访问 • webshell_0001 : 网站后门 • network_1001 : 恶意挖矿 • network_1002 : 对外DDoS攻击 • network_1003 : 恶意扫描 • network_1004 : 敏感区域攻击 • ransomware_0001 : 勒索攻击 • ransomware_0002 : 勒索攻击

参数	是否必选	参数类型	描述
			• ransomware_0003：勒索攻击 • fileless_0001：进程注入 • fileless_0002：动态库注入进程 • fileless_0003：关键配置变更 • fileless_0004：环境变量变更 • fileless_0005：内存文件进程 • fileless_0006：vdso劫持 • crontab_1001：Crontab可疑任务 • vul_exploit_0001：Redis漏洞利用攻击 • vul_exploit_0002：Hadoop漏洞利用攻击 • vul_exploit_0003：MySQL漏洞利用攻击 • rootkit_0001：可疑rootkit文件 • rootkit_0002：可疑内核模块 • RASP_0004：上传Webshell • RASP_0018：无文件Webshell • blockexec_001：已知勒索攻击 • hips_0001：Windows Defender防护被禁用 • hips_0002：可疑的黑客工具 • hips_0003：可疑的勒索加密行为 • hips_0004：隐藏账号创建 • hips_0005：读取用户密码凭据 • hips_0006：可疑的SAM文件导出 • hips_0007：可疑shadow copy删除操作 • hips_0008：备份文件删除 • hips_0009：可疑勒索病毒操作注册表

参数	是否必选	参数类型	描述
			• hips_0010 : 可疑的异常进程行为 • hips_0011 : 可疑的扫描探测 • hips_0012 : 可疑的勒索病毒脚本运行 • hips_0013 : 可疑的挖矿命令执行 • hips_0014 : 可疑的禁用windows安全中心 • hips_0015 : 可疑的停止防火墙服务行为 • hips_0016 : 可疑的系统自动恢复禁用 • hips_0017 : Offies 创建可执行文件 • hips_0018 : 带宏Offies文件异常创建 • hips_0019 : 可疑的注册表操作 • hips_0020 : Confluence远程代码执行 • hips_0021 : MSDT远程代码执行 • portscan_0001 : 通用端口扫描 • portscan_0002 : 秘密端口扫描 • k8s_1001 : Kubernetes事件删除 • k8s_1002 : 创建特权Pod • k8s_1003 : Pod中使用交互式shell • k8s_1004 : 创建敏感目录Pod • k8s_1005 : 创建主机网络的Pod • k8s_1006 : 创建主机Pid空间的Pod • k8s_1007 : 普通pod访问APIserver认证失败 • k8s_1008 : 普通Pod通过Curl访问APIServer

参数	是否必选	参数类型	描述
			• k8s_1009：系统管理空间执行exec • k8s_1010：系统管理空间创建Pod • k8s_1011：创建静态Pod • k8s_1012：创建DaemonSet • k8s_1013：创建集群计划任务 • k8s_1014：Secrets操作 • k8s_1015：枚举用户可执行的操作 • k8s_1016：高权限RoleBinding或ClusterRoleBinding • k8s_1017：ServiceAccount创建 • k8s_1018：创建Cronjob • k8s_1019：Pod中exec使用交互式shell • k8s_1020：无权限访问Apiserver • k8s_1021：使用curl访问APIServer • k8s_1022：Ingress漏洞 • k8s_1023：中间人攻击 • k8s_1024：蠕虫挖矿木马 • k8s_1025：K8s事件删除 • k8s_1026：SelfSubjectRulesReview场景 • imgblock_0001：镜像白名单阻断 • imgblock_0002：镜像黑名单阻断 • imgblock_0003：镜像标签白名单阻断 • imgblock_0004：镜像标签黑名单阻断 • imgblock_0005：创建容器白名单阻断 • imgblock_0006：创建容器黑名单阻断

参数	是否必选	参数类型	描述
			• imgblock_0007 : 容器mount proc阻断 • imgblock_0008 : 容器 seccomp unconfined阻断 • imgblock_0009 : 容器特权阻断 • imgblock_0010 : 容器 capabilities阻断 数组长度：**1 - 200**
severity_list	否	Array	威胁等级，包含如下: • Security： 安全 • Low : 低危 • Medium : 中危 • High : 高危 • Critical : 危急 最小长度：**0** 最大长度：**32** 数组长度：**0 - 5**
attack_tag	否	String	攻击标识，包含如下: • attack_success : 攻击成功 • attack_attempt : 攻击尝试 • attack_blocked : 攻击被阻断 • abnormal_behavior : 异常行为 • collapsible_host : 主机失陷 • system_vulnerability : 系统脆弱性 最小长度：**0** 最大长度：**32**
asset_value	否	String	资产重要性，包含如下3种 • important： 重要资产 • common： 一般资产 • test： 测试资产 最小长度：**0** 最大长度：**128**

参数	是否必选	参数类型	描述
tag_list	否	Array	事件标签列表，例如:["热点事件"] 最小长度：**0** 最大长度：**10** 数组长度：**0 - 20**
att_ck	否	String	ATT&CK攻击阶，包含如下： ● Reconnaissance : 侦察 ● Initial Access : 初始访问 ● Execution : 执行 ● Persistence : 持久化 ● Privilege Escalation : 权限提升 ● Defense Evasion : 防御绕过 ● Credential Access : 凭据访问 ● Command and Control : 命令与控制 ● Impact : 影响破坏 最小长度：**0** 最大长度：**32**
event_name	否	String	告警名称 最小长度：**1** 最大长度：**128**

请求参数

表 3-239 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-240 响应 Body 参数

参数	参数类型	描述
total_num	Integer	告警事件总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **EventManagementResponseInfo** objects	事件列表详情 数组长度：**0 - 1000**

表 3-241 EventManagementResponseInfo

参数	参数类型	描述
event_id	String	事件编号

参数	参数类型	描述
event_class_id	String	事件分类，包含如下: ● container_1001：容器命名空间 ● container_1002：容器开放端口 ● container_1003：容器安全选项 ● container_1004：容器挂载目录 ● containerescape_0001：容器高危系统调用 ● containerescape_0002：Shocker攻击 ● containerescape_0003：DirtCow攻击 ● containerescape_0004：容器文件逃逸攻击 ● dockerfile_001：用户自定义容器保护文件被修改 ● dockerfile_002：容器文件系统可执行文件被修改 ● dockerproc_001：容器进程异常事件上报 ● fileprotect_0001：文件提权 ● fileprotect_0002：关键文件变更 ● fileprotect_0003：关键文件路径变更 ● fileprotect_0004：文件/目录变更 ● av_1002：病毒 ● av_1003：蠕虫 ● av_1004：木马 ● av_1005：僵尸网络 ● av_1006：后门 ● av_1007：间谍软件 ● av_1008：恶意广告软件 ● av_1009：钓鱼 ● av_1010：Rootkit ● av_1011：勒索软件 ● av_1012：黑客工具 ● av_1013：灰色软件 ● av_1015：Webshell ● av_1016：挖矿软件 ● login_0001：尝试暴力破解 ● login_0002：爆破成功 ● login_1001：登录成功 ● login_1002：异地登录 ● login_1003：弱口令 ● malware_0001：shell变更事件上报

参数	参数类型	描述
		• malware_0002：反弹shell事件上报 • malware_1001：恶意程序 • procdet_0001：进程异常行为检测 • procdet_0002：进程提权 • procreport_0001：危险命令 • user_1001：账号变更 • user_1002：风险账号 • vmescape_0001：虚拟机敏感命令执行 • vmescape_0002：虚拟化进程访问敏感文件 • vmescape_0003：虚拟机异常端口访问 • webshell_0001：网站后门 • network_1001：恶意挖矿 • network_1002：对外DDoS攻击 • network_1003：恶意扫描 • network_1004：敏感区域攻击 • ransomware_0001：勒索攻击 • ransomware_0002：勒索攻击 • ransomware_0003：勒索攻击 • fileless_0001：进程注入 • fileless_0002：动态库注入进程 • fileless_0003：关键配置变更 • fileless_0004：环境变量变更 • fileless_0005：内存文件进程 • fileless_0006：vdso劫持 • crontab_1001：Crontab可疑任务 • vul_exploit_0001：Redis漏洞利用攻击 • vul_exploit_0002：Hadoop漏洞利用攻击 • vul_exploit_0003：MySQL漏洞利用攻击 • rootkit_0001：可疑rootkit文件 • rootkit_0002：可疑内核模块 • RASP_0004：上传Webshell • RASP_0018：无文件Webshell • blockexec_001：已知勒索攻击 • hips_0001：Windows Defender防护被禁用 • hips_0002：可疑的黑客工具 • hips_0003：可疑的勒索加密行为 • hips_0004：隐藏账号创建

参数	参数类型	描述
		• hips_0005 : 读取用户密码凭据 • hips_0006 : 可疑的SAM文件导出 • hips_0007 : 可疑shadow copy删除操作 • hips_0008 : 备份文件删除 • hips_0009 : 可疑勒索病毒操作注册表 • hips_0010 : 可疑的异常进程行为 • hips_0011 : 可疑的扫描探测 • hips_0012 : 可疑的勒索病毒脚本运行 • hips_0013 : 可疑的挖矿命令执行 • hips_0014 : 可疑的禁用windows安全中心 • hips_0015 : 可疑的停止防火墙服务行为 • hips_0016 : 可疑的系统自动恢复禁用 • hips_0017 : Offies 创建可执行文件 • hips_0018 : 带宏Offies文件异常创建 • hips_0019 : 可疑的注册表操作 • hips_0020 : Confluence远程代码执行 • hips_0021 : MSDT远程代码执行 • portscan_0001 : 通用端口扫描 • portscan_0002 : 秘密端口扫描 • k8s_1001 : Kubernetes事件删除 • k8s_1002 : 创建特权Pod • k8s_1003 : Pod中使用交互式shell • k8s_1004 : 创建敏感目录Pod • k8s_1005 : 创建主机网络的Pod • k8s_1006 : 创建主机Pid空间的Pod • k8s_1007 : 普通pod访问APIserver认证失败 • k8s_1008 : 普通Pod通过Curl访问APIServer • k8s_1009 : 系统管理空间执行exec • k8s_1010 : 系统管理空间创建Pod • k8s_1011 : 创建静态Pod • k8s_1012 : 创建DaemonSet • k8s_1013 : 创建集群计划任务 • k8s_1014 : Secrets操作 • k8s_1015 : 枚举用户可执行的操作 • k8s_1016 : 高权限RoleBinding或ClusterRoleBinding • k8s_1017 : ServiceAccount创建

参数	参数类型	描述
		• k8s_1018 : 创建Cronjob • k8s_1019 : Pod中exec使用交互式shell • k8s_1020 : 无权限访问Apiserver • k8s_1021 : 使用curl访问APIServer • k8s_1022 : Ingress漏洞 • k8s_1023 : 中间人攻击 • k8s_1024 : 蠕虫挖矿木马 • k8s_1025 : K8s事件删除 • k8s_1026 : SelfSubjectRulesReview场景 • imgblock_0001 : 镜像白名单阻断 • imgblock_0002 : 镜像黑名单阻断 • imgblock_0003 : 镜像标签白名单阻断 • imgblock_0004 : 镜像标签黑名单阻断 • imgblock_0005 : 创建容器白名单阻断 • imgblock_0006 : 创建容器黑名单阻断 • imgblock_0007 : 容器mount proc阻断 • imgblock_0008 : 容器seccomp unconfined阻断 • imgblock_0009 : 容器特权阻断 • imgblock_0010 : 容器capabilities阻断

参数	参数类型	描述
event_type	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 : 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 : 系统安全防护被禁用 ● 3030 : 备份删除 ● 3031 : 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描 ● 6003 : 主机扫描

参数	参数类型	描述
		● 13001 : Kubernetes事件删除 ● 13002 : Pod异常行为 ● 13003 : 枚举用户信息 ● 13004 : 绑定集群用户角色
event_name	String	事件名称
severity	String	威胁等级，包含如下: ● Security : 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 ● Critical : 危急
container_name	String	容器实例名称，只有容器类型的告警有
image_name	String	镜像名称，只有容器类型的告警有
host_name	String	服务器名称
host_id	String	主机ID
private_ip	String	服务器私有IP
public_ip	String	弹性公网IP地址
os_type	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。
host_status	String	服务器状态，包含如下4种。 ● ACTIVE： 运行中。 ● SHUTOFF： 关机。 ● BUILDING： 创建中。 ● ERROR： 故障。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
agent_status	String	Agent状态，包含如下5种。 ● installed： 已安装。 ● not_installed： 未安装。 ● online： 在线。 ● offline： 离线。 ● install_failed： 安装失败。 ● installing： 安装中。 最小长度：**1** 最大长度：**32**
protect_status	String	防护状态，包含如下2种。 ● closed： 未防护。 ● opened： 防护中。 最小长度：**1** 最大长度：**32**
asset_value	String	资产重要性，包含如下4种 ● important： 重要资产 ● common： 一般资产 ● test： 测试资产 最小长度：**0** 最大长度：**128**
attack_phase	String	攻击阶段，包含如下： ● reconnaissance : 侦查跟踪 ● weaponization : 武器构建 ● delivery : 载荷投递 ● exploit : 漏洞利用 ● installation : 安装植入 ● command_and_control : 命令与控制 ● actions : 目标达成
attack_tag	String	攻击标识，包含如下： ● attack_success : 攻击成功 ● attack_attempt : 攻击尝试 ● attack_blocked : 攻击被阻断 ● abnormal_behavior : 异常行为 ● collapsible_host : 主机失陷 ● system_vulnerability : 系统脆弱性
occur_time	Integer	发生时间，毫秒

参数	参数类型	描述
handle_time	Integer	处理时间，毫秒，已处理的告警才有
handle_status	String	处理状态，包含如下: • unhandled ： 未处理 • handled : 已处理
handle_method	String	处理方式，已处理的告警才有，包含如下: • mark_as_handled : 手动处理 • ignore : 忽略 • add_to_alarm_whitelist : 加入告警白名单 • add_to_login_whitelist : 加入登录白名单 • isolate_and_kill : 隔离查杀
handler	String	备注信息，已处理的告警才有
operate_accept_list	Array of strings	支持的处理操作
operate_detail_list	Array of **EventDetailResponseInfo** objects	操作详情信息列表（页面不展示） 数组长度：**0 - 100**
forensic_info	Object	取证信息，json格式
resource_info	**EventResourceResponseInfo** object	资源信息
geo_info	Object	地理位置信息，json格式
malware_info	Object	恶意软件信息，json格式
network_info	Object	网络信息，json格式
app_info	Object	应用信息，json格式
system_info	Object	系统信息，json格式
extend_info	Object	事件扩展信息，json格式
recommendation	String	处置建议
description	String	告警说明 最小长度：**0** 最大长度：**1024**
event_abstract	String	告警摘要 最小长度：**0** 最大长度：**512**

参数	参数类型	描述
process_info_list	Array of **EventProcessResponseInfo** objects	进程信息列表 数组长度：**0 - 100**
user_info_list	Array of **EventUserResponseInfo** objects	用户信息列表 数组长度：**0 - 100**
file_info_list	Array of **EventFileResponseInfo** objects	文件信息列表 数组长度：**0 - 100**
event_details	String	事件信息的简述 最小长度：**0** 最大长度：**204800**
tag_list	Array of strings	标签列表 最小长度：**0** 最大长度：**10** 数组长度：**0 - 20**
event_count	Integer	事件发生次数 最小值：**0** 最大值：**2147483647**

表 3-242 EventDetailResponseInfo

参数	参数类型	描述
agent_id	String	Agent ID
process_pid	Integer	进程id
is_parent	Boolean	是否是父进程
file_hash	String	文件哈希
file_path	String	文件路径
file_attr	String	文件属性
private_ip	String	服务器私有IP
login_ip	String	登录源IP
login_user_name	String	登录用户名

参数	参数类型	描述
keyword	String	告警事件关键字，仅用于告警白名单
hash	String	告警事件hash，仅用于告警白名单

表 3-243 EventResourceResponseInfo

参数	参数类型	描述
domain_id	String	租户账号ID
project_id	String	项目ID
enterprise_project_id	String	企业项目ID
region_name	String	Region名称
vpc_id	String	VPC ID
cloud_id	String	云主机ID
vm_name	String	虚拟机名称
vm_uuid	String	虚拟机UUID，即主机ID
container_id	String	容器ID
container_status	String	容器状态
pod_uid	String	pod uid
pod_name	String	pod name
namespace	String	namespace
cluster_id	String	集群id
cluster_name	String	集群名称
image_id	String	镜像ID
image_name	String	镜像名称
host_attr	String	主机属性
service	String	业务服务
micro_service	String	微服务
sys_arch	String	系统CPU架构
os_bit	String	操作系统位数
os_type	String	操作系统类型

参数	参数类型	描述
os_name	String	操作系统名称
os_version	String	操作系统版本

表 3-244 EventProcessResponseInfo

参数	参数类型	描述
process_name	String	进程名称
process_path	String	进程文件路径
process_pid	Integer	进程id 最小值：**0** 最大值：**2147483647**
process_uid	Integer	进程用户id 最小值：**0** 最大值：**2147483647**
process_username	String	运行进程的用户名
process_cmdline	String	进程文件命令行
process_filename	String	进程文件名
process_start_time	Long	进程启动时间 最小值：**0** 最大值：**9223372036854775807**
process_gid	Integer	进程组ID 最小值：**0** 最大值：**2147483647**
process_egid	Integer	进程有效组ID 最小值：**0** 最大值：**2147483647**
process_euid	Integer	进程有效用户ID 最小值：**0** 最大值：**2147483647**
ancestor_process_path	String	祖父进程文件路径

参数	参数类型	描述
ancestor_process_pid	Integer	祖父进程id 最小值：**0** 最大值：**2147483647**
ancestor_process_cmdline	String	祖父进程文件命令行
parent_process_name	String	父进程名称
parent_process_path	String	父进程文件路径
parent_process_pid	Integer	父进程id 最小值：**0** 最大值：**2147483647**
parent_process_uid	Integer	父进程用户id 最小值：**0** 最大值：**2147483647**
parent_process_cmdline	String	父进程文件命令行
parent_process_filename	String	父进程文件名
parent_process_start_time	Long	父进程启动时间 最小值：**0** 最大值：**9223372036854775807**
parent_process_gid	Integer	父进程组ID 最小值：**0** 最大值：**2147483647**
parent_process_egid	Integer	父进程有效组ID 最小值：**0** 最大值：**2147483647**
parent_process_euid	Integer	父进程有效用户ID 最小值：**0** 最大值：**2147483647**
child_process_name	String	子进程名称
child_process_path	String	子进程文件路径

参数	参数类型	描述
child_process_pid	Integer	子进程id 最小值：**0** 最大值：**2147483647**
child_process_uid	Integer	子进程用户id 最小值：**0** 最大值：**2147483647**
child_process_cmdline	String	子进程文件命令行
child_process_filename	String	子进程文件名
child_process_start_time	Long	子进程启动时间 最小值：**0** 最大值：**9223372036854775807**
child_process_gid	Integer	子进程组ID 最小值：**0** 最大值：**2147483647**
child_process_egid	Integer	子进程有效组ID 最小值：**0** 最大值：**2147483647**
child_process_euid	Integer	子进程有效用户ID 最小值：**0** 最大值：**2147483647**
virt_cmd	String	虚拟化命令
virt_process_name	String	虚拟化进程名称
escape_mode	String	逃逸方式
escape_cmd	String	逃逸后后执行的命令
process_hash	String	进程启动文件hash
process_file_hash	String	进程文件hash
parent_process_file_hash	String	父进程文件hash
block	Integer	是否阻断成功，1阻断成功 0阻断失败 最小值：**0** 最大值：**1**

表 3-245 EventUserResponseInfo

参数	参数类型	描述
user_id	Integer	用户uid 最小值：**0** 最大值：**2147483647**
user_gid	Integer	用户gid 最小值：**0** 最大值：**2147483647**
user_name	String	用户名称
user_group_name	String	用户组名称
user_home_dir	String	用户home目录
login_ip	String	用户登录ip
service_type	String	服务类型，包含如下: ● system ● mysql ● redis
service_port	Integer	登录服务端口 最小值：**0** 最大值：**2147483647**
login_mode	Integer	登录方式 最小值：**0** 最大值：**2147483647**
login_last_time	Long	用户最后一次登录时间 最小值：**0** 最大值：**9223372036854775807**
login_fail_count	Integer	用户登录失败次数 最小值：**0** 最大值：**2147483647**
pwd_hash	String	口令hash
pwd_with_fuzzing	String	匿名化处理后的口令

参数	参数类型	描述
pwd_used_days	Integer	密码使用的天数 最小值：**0** 最大值：**2147483647**
pwd_min_days	Integer	口令的最短有效期限 最小值：**0** 最大值：**2147483647**
pwd_max_days	Integer	口令的最长有效期限 最小值：**0** 最大值：**2147483647**
pwd_warn_left_days	Integer	口令无效时提前告警天数 最小值：**0** 最大值：**2147483647**

表 3-246 EventFileResponseInfo

参数	参数类型	描述
file_path	String	文件路径
file_alias	String	文件别名
file_size	Integer	文件大小 最小值：**0** 最大值：**2147483647**
file_mtime	Long	文件最后一次修改时间 最小值：**0** 最大值：**9223372036854775807**
file_atime	Long	文件最后一次访问时间 最小值：**0** 最大值：**9223372036854775807**
file_ctime	Long	文件最后一次状态改变时间 最小值：**0** 最大值：**9223372036854775807**
file_hash	String	文件hash,当前为sha256
file_md5	String	文件md5
file_sha256	String	文件sha256
file_type	String	文件类型

参数	参数类型	描述
file_content	String	文件内容
file_attr	String	文件属性
file_operation	Integer	文件操作类型 最小值：**0** 最大值：**2147483647**
file_action	String	文件动作
file_change_attr	String	变更前后的属性
file_new_path	String	新文件路径
file_desc	String	文件描述
file_key_word	String	文件关键字
is_dir	Boolean	是否目录
fd_info	String	文件句柄信息
fd_count	Integer	文件句柄数量 最小值：**0** 最大值：**2147483647**

请求示例

查询前50条企业项目为xxx下未处理的主机事件信息

```
GET https://{endpoint}/v5/{project_id}/event/events?
offset=0&limit=50&handle_status=unhandled&category=host&enterprise_project_id=xxx
```

响应示例

状态码： 200

入侵事件列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "attack_phase" : "exploit",
    "attack_tag" : "abnormal_behavior",
    "event_class_id" : "lgin_1002",
    "event_id" : "d8a12cf7-6a43-4cd6-92b4-aabf1e917",
    "event_name" : "different locations",
    "event_type" : 4004,
    "forensic_info" : {
      "country" : "中国",
      "city" : "兰州市",
      "ip" : "127.0.0.1",
      "user" : "zhangsan",
      "sub_division" : "甘肃省",
      "city_id" : 3110
    },
```

```
  "handle_status" : "unhandled",
  "host_name" : "xxx",
  "occur_time" : 1661593036627,
  "operate_accept_list" : [ "ignore" ],
  "operate_detail_list" : [ {
    "agent_id" : "c9bed5397db449ebdfba15e85fcfc36accee125c68954daf5cab0528bab59bd8",
    "file_hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "file_path" : "/usr/test",
    "process_pid" : 3123,
    "file_attr" : 33261,
    "keyword" : "file_path=/usr/test",
    "hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "login_ip" : "127.0.0.1",
    "private_ip" : "127.0.0.2",
    "login_user_name" : "root",
    "is_parent" : false
  } ],
  "private_ip" : "127.0.0.1",
  "resource_info" : {
    "region_name" : "",
    "project_id" : "",
    "enterprise_project_id" : "0",
    "os_type" : "Linux",
    "os_version" : "2.5",
    "vm_name" : "",
    "vm_uuid" : "71a15ecc",
    "cloud_id" : "",
    "container_id" : "",
    "container_status" : "running / terminated",
    "image_id" : "",
    "pod_uid" : "",
    "pod_name" : "",
    "namespace" : "",
    "cluster_id" : "",
    "cluster_name" : ""
  },
  "severity" : "Medium",
  "extend_info" : "",
  "os_type" : "Linux",
  "agent_status" : "online",
  "asset_value" : "common",
  "protect_status" : "opened",
  "host_status" : "ACTIVE",
  "event_details" : "file_path:/root/test",
  "user_info_list" : [ {
    "login_ip" : "",
    "service_port" : 22,
    "service_type" : "ssh",
    "user_name" : "zhangsan",
    "login_mode" : 0,
    "login_last_time" : 1661593024,
    "login_fail_count" : 0
  } ],
  "process_info_list" : [ {
    "process_path" : "/root/test",
    "process_name" : "test",
    "process_cmdline" : "/bin/bash",
    "process_hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "process_filename" : "test",
    "process_file_hash" : "e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
    "process_username" : "root",
    "process_pid" : 372612,
    "process_uid" : 10000,
    "process_gid" : 10000,
    "process_egid" : 10000,
    "process_euid" : 10000,
    "process_start_time" : 1661593024,
    "block" : 0,
    "parent_process_path" : "/usr/bin/bash",
```

```
      "parent_process_name" : "test",
      "parent_process_cmdline" : "/bin/bash",
      "parent_process_filename" : "test",
      "parent_process_file_hash" :
"e8b50f0b91e3dce0885ccc5902846b139d28108a0a7976c9b8d43154c5dbc44d",
      "parent_process_pid" : 372612,
      "parent_process_uid" : 10000,
      "parent_process_gid" : 10000,
      "parent_process_egid" : 10000,
      "parent_process_euid" : 10000,
      "parent_process_start_time" : 1661593024,
      "child_process_path" : "/usr/bin/bash",
      "child_process_name" : "test",
      "child_process_cmdline" : "/bin/bash",
      "child_process_filename" : "test",
      "child_process_pid" : 372612,
      "child_process_uid" : 10000,
      "child_process_gid" : 10000,
      "child_process_egid" : 10000,
      "child_process_euid" : 10000,
      "child_process_start_time" : 1661593024,
      "virt_process_name" : "test",
      "virt_cmd" : "/bin/bash",
      "escape_cmd" : "/bin/bash",
      "escape_mode" : "0",
      "ancestor_process_pid" : 372612,
      "ancestor_process_cmdline" : "/bin/bash",
      "ancestor_process_path" : "/usr/bin/bash"
    } ],
    "description" : "",
    "event_abstract" : "",
    "tag_list" : [ "热点事件" ]
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ListSecurityEventsSolution {

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

        ICredential auth = new BasicCredentials()
                .withAk(ak)
```

```
            .withSk(sk);

    HssClient client = HssClient.newBuilder()
            .withCredential(auth)
            .withRegion(HssRegion.valueOf("<YOUR REGION>"))
            .build();
    ListSecurityEventsRequest request = new ListSecurityEventsRequest();
    request.withCategory("<category>");
    request.withEnterpriseProjectId("<enterprise_project_id>");
    request.withLastDays(<last_days>);
    request.withHostName("<host_name>");
    request.withHostId("<host_id>");
    request.withPrivateIp("<private_ip>");
    request.withPublicIp("<public_ip>");
    request.withContainerName("<container_name>");
    request.withOffset(<offset>);
    request.withLimit(<limit>);
    request.withEventTypes();
    request.withHandleStatus("<handle_status>");
    request.withSeverity("<severity>");
    request.withBeginTime("<begin_time>");
    request.withEndTime("<end_time>");
    request.withEventClassIds();
    request.withSeverityList();
    request.withAttackTag("<attack_tag>");
    request.withAssetValue("<asset_value>");
    request.withTagList();
    request.withAttCk("<att_ck>");
    request.withEventName("<event_name>");
    try {
        ListSecurityEventsResponse response = client.listSecurityEvents(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

    client = HssClient.new_builder() \
        .with_credentials(credentials) \
```

```
        .with_region(HssRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSecurityEventsRequest()
        request.category = "<category>"
        request.enterprise_project_id = "<enterprise_project_id>"
        request.last_days = <last_days>
        request.host_name = "<host_name>"
        request.host_id = "<host_id>"
        request.private_ip = "<private_ip>"
        request.public_ip = "<public_ip>"
        request.container_name = "<container_name>"
        request.offset = <offset>
        request.limit = <limit>
        request.event_types =
        request.handle_status = "<handle_status>"
        request.severity = "<severity>"
        request.begin_time = "<begin_time>"
        request.end_time = "<end_time>"
        request.event_class_ids =
        request.severity_list =
        request.attack_tag = "<attack_tag>"
        request.asset_value = "<asset_value>"
        request.tag_list =
        request.att_ck = "<att_ck>"
        request.event_name = "<event_name>"
        response = client.list_security_events(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListSecurityEventsRequest{}
    request.Category = "<category>"
```

```
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    lastDaysRequest:= int32(<last_days>)
    request.LastDays = &lastDaysRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
    publicIpRequest:= "<public_ip>"
    request.PublicIp = &publicIpRequest
    containerNameRequest:= "<container_name>"
    request.ContainerName = &containerNameRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    handleStatusRequest:= "<handle_status>"
    request.HandleStatus = &handleStatusRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    beginTimeRequest:= "<begin_time>"
    request.BeginTime = &beginTimeRequest
    endTimeRequest:= "<end_time>"
    request.EndTime = &endTimeRequest
    attackTagRequest:= "<attack_tag>"
    request.AttackTag = &attackTagRequest
    assetValueRequest:= "<asset_value>"
    request.AssetValue = &assetValueRequest
    attCkRequest:= "<att_ck>"
    request.AttCk = &attCkRequest
    eventNameRequest:= "<event_name>"
    request.EventName = &eventNameRequest
    response, err := client.ListSecurityEvents(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	入侵事件列表

错误码

请参见**错误码**。

3.7.3 查询告警白名单列表

功能介绍

查询告警白名单列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/event/white-list/alarm

表 3-247 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-248 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**64**
hash	否	String	事件白名单SHA256 最小长度：**64** 最大长度：**64**

参数	是否必选	参数类型	描述
event_type	否	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 : 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 : 系统安全防护被禁用 ● 3030 : 备份删除 ● 3031 : 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描 ● 6003 : 主机扫描

参数	是否必选	参数类型	描述
			• 13001：Kubernetes事件删除 • 13002：Pod异常行为 • 13003：枚举用户信息 • 13004：绑定集群用户角色 最小值：**1000** 最大值：**30000**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**1000** 缺省值：**10**

请求参数

表 3-249 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-250 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数
event_type_list	Array of integers	支持筛选的事件类型 最小值：**0** 最大值：**2147483647** 数组长度：**0 - 30000**
data_list	Array of **AlarmWhiteListResponseInfo** objects	告警白名单详情 数组长度：**0 - 100**

表 3-251 AlarmWhiteListResponseInfo

参数	参数类型	描述
enterprise_project_name	String	企业项目名称
hash	String	事件白名单SHA256
description	String	描述信息

参数	参数类型	描述
event_type	Integer	事件类型，包含如下: ● 1001 : 通用恶意软件 ● 1002 : 病毒 ● 1003 : 蠕虫 ● 1004 : 木马 ● 1005 : 僵尸网络 ● 1006 : 后门 ● 1010 : Rootkit ● 1011 : 勒索软件 ● 1012 ： 黑客工具 ● 1015 : Webshell ● 1016 : 挖矿 ● 1017 : 反弹Shell ● 2001 : 一般漏洞利用 ● 2012 : 远程代码执行 ● 2047 : Redis漏洞利用 ● 2048 : Hadoop漏洞利用 ● 2049 : MySQL漏洞利用 ● 3002 : 文件提权 ● 3003 : 进程提权 ● 3004 : 关键文件变更 ● 3005 : 文件/目录变更 ● 3007 : 进程异常行为 ● 3015 : 高危命令执行 ● 3018 : 异常Shell ● 3027 : Crontab可疑任务 ● 3029 ： 系统安全防护被禁用 ● 3030 ： 备份删除 ● 3031 ： 异常注册表操作 ● 3036 : 容器镜像阻断 ● 4002 : 暴力破解 ● 4004 : 异常登录 ● 4006 : 非法系统账号 ● 4014 : 用户账号添加 ● 4020 : 用户密码窃取 ● 6002 : 端口扫描 ● 6003 : 主机扫描

参数	参数类型	描述
		● 13001 : Kubernetes事件删除 ● 13002 : Pod异常行为 ● 13003 : 枚举用户信息 ● 13004 : 绑定集群用户角色
white_field	String	加白字段，包含如下: ● "file/process hash" # 进程/文件hash ● "file_path" # 文件路径 ● "process_path" # 进程路径 ● "login_ip" # 登录ip ● "reg_key" #注册表key ● "process_cmdline" # 进程命令行 ● "username" # 用户名 最小长度：**1** 最大长度：**20**
field_value	String	加白字段值 最小长度：**1** 最大长度：**128**
judge_type	String	通配符，包含如下: ● "equal" # 相等 ● "contain" # 包含 最小长度：**1** 最大长度：**10**
update_time	Long	事件白名单更新时间，毫秒 最小值：**0** 最大值：**9223372036854775807**

请求示例

查询前10条企业项目为xxx下的告警白名单列表

```
GET https://{endpoint}/v5/{project_id}/event/white-list/alarm?limit=10&offset=0&enterprise_project_id=xxx
```

响应示例

状态码： 200

告警白名单列表

```
{
  "data_list" : [ {
    "enterprise_project_name" : "所有项目",
    "event_type" : 1001,
```

```
    "hash" : "9ab079e5398cba3a368ccffbd478f54c5ec3edadf6284ec049a73c36419f1178",
    "description" : "/opt/cloud/3rdComponent/install/jre-8u201/bin/java",
    "update_time" : 1665715677307,
    "white_field" : "process/file hash",
    "judge_type" : "contain",
    "field_value" : "abcd12345612311112212323"
  } ],
  "event_type_list" : [ 1001 ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListAlarmWhiteListSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListAlarmWhiteListRequest request = new ListAlarmWhiteListRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHash("<hash>");
        request.withEventType(<event_type>);
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListAlarmWhiteListResponse response = client.listAlarmWhiteList(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListAlarmWhiteListRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.hash = "<hash>"
        request.event_type = <event_type>
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_alarm_white_list(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

```
    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAlarmWhiteListRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hashRequest:= "<hash>"
    request.Hash = &hashRequest
    eventTypeRequest:= int32(<event_type>)
    request.EventType = &eventTypeRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListAlarmWhiteList(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	告警白名单列表

错误码

请参见**错误码**。

3.8 主机管理

3.8.1 查询云服务器列表

功能介绍

查询云服务器列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/host-management/hosts

表 3-252 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-253 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
version	否	String	主机开通的版本，包含如下7种输入。 ● hss.version.null ： 无。 ● hss.version.basic ： 基础版。 ● hss.version.advanced ： 专业版。 ● hss.version.enterprise ： 企业版。 ● hss.version.premium ： 旗舰版。 ● hss.version.wtp ： 网页防篡改版。 ● hss.version.container.enterprise：容器版。 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
agent_status	否	String	Agent状态，包含如下6种。 ● installed： 已安装。 ● not_installed： 未安装。 ● online： 在线。 ● offline： 离线。 ● install_failed： 安装失败。 ● installing： 安装中。 ● not_online： 不在线的（除了在线以外的所有状态，仅作为查询条件）。 最小长度：**1** 最大长度：**20**
detect_result	否	String	检测结果，包含如下4种。 ● undetected： 未检测。 ● clean： 无风险。 ● risk： 有风险。 ● scanning： 检测中。 最小长度：**1** 最大长度：**32**
host_name	否	String	服务器名称
host_id	否	String	服务器ID
host_status	否	String	主机状态，包含如下4种。 ● ACTIVE： 正在运行。 ● SHUTOFF： 关机。 ● BUILDING： 创建中。 ● ERROR： 故障。 最小长度：**1** 最大长度：**32**
os_type	否	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。 最小长度：**0** 最大长度：**64**
private_ip	否	String	服务器私有IP
public_ip	否	String	服务器公网IP
ip_addr	否	String	公网或私网IP

参数	是否必选	参数类型	描述
protect_status	否	String	防护状态，包含如下2种。 ● closed： 关闭。 ● opened： 开启。 最小长度：**1** 最大长度：**32**
group_id	否	String	服务器组ID
group_name	否	String	服务器组名称 最小长度：**1** 最大长度：**64**
has_intrusion	否	Boolean	存在告警事件
policy_group_id	否	String	策略组ID 最小长度：**0** 最大长度：**128**
policy_group_name	否	String	策略组名称 最小长度：**0** 最大长度：**256**
charging_mode	否	String	收费模式，包含如下2种。 ● packet_cycle： 包年/包月。 ● on_demand： 按需。 最小长度：**1** 最大长度：**32**
refresh	否	Boolean	是否强制从ECS同步主机
above_version	否	Boolean	是否返回比当前版本高的所有版本
outside_host	否	Boolean	是否华为云主机
asset_value	否	String	资产重要性，包含如下4种 ● important： 重要资产 ● common： 一般资产 ● test： 测试资产 最小长度：**0** 最大长度：**128**
label	否	String	资产标签 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
server_group	否	String	资产服务器组 最小长度：**1** 最大长度：**64**
agent_upgradable	否	Boolean	agent是否可升级
limit	否	Integer	每页显示数量 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

请求参数

表 3-254 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-255 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2097152**
data_list	Array of **Host** objects	查询弹性云服务器状态列表 数组长度：**0 - 10241**

表 3-256 Host

参数	参数类型	描述
host_name	String	服务器名称 最小长度：**0** 最大长度：**128**
host_id	String	服务器ID 最小长度：**0** 最大长度：**128**
agent_id	String	Agent ID 最小长度：**0** 最大长度：**128**
private_ip	String	私有IP地址 最小长度：**0** 最大长度：**128**
public_ip	String	弹性公网IP地址 最小长度：**0** 最大长度：**128**
enterprise_project_id	String	企业项目ID 最小长度：**0** 最大长度：**256**
enterprise_project_name	String	所属企业项目名称 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
host_status	String	服务器状态，包含如下4种。 ● ACTIVE： 运行中。 ● SHUTOFF： 关机。 ● BUILDING： 创建中。 ● ERROR： 故障。 最小长度：**1** 最大长度：**32**
agent_status	String	Agent状态，包含如下5种。 ● installed： 已安装。 ● not_installed： 未安装。 ● online： 在线。 ● offline： 离线。 ● install_failed： 安装失败。 ● installing： 安装中。 最小长度：**1** 最大长度：**32**
install_result_code	String	安装结果，包含如下12种。 ● install_succeed： 安装成功。 ● network_access_timeout： 网络不通，访问超时。 ● invalid_port： 无效端口。 ● auth_failed： 认证错误，口令不正确。 ● permission_denied： 权限错误，被拒绝。 ● no_available_vpc： 没有相同VPC的agent在线虚拟机。 ● install_exception： 安装异常。 ● invalid_param： 参数错误。 ● install_failed： 安装失败。 ● package_unavailable： 安装包失效。 ● os_type_not_support： 系统类型错误。 ● os_arch_not_support： 架构类型错误。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
version	String	主机开通的版本，包含如下7种输入。 ● hss.version.null： 无。 ● hss.version.basic： 基础版。 ● hss.version.advanced： 专业版。 ● hss.version.enterprise： 企业版。 ● hss.version.premium： 旗舰版。 ● hss.version.wtp： 网页防篡改版。 ● hss.version.container.enterprise： 容器版。 最小长度：**1** 最大长度：**32**
protect_status	String	防护状态，包含如下2种。 ● closed： 未防护。 ● opened： 防护中。 最小长度：**1** 最大长度：**32**
os_image	String	系统镜像 最小长度：**0** 最大长度：**128**
os_type	String	操作系统类型，包含如下2种。 ● Linux： Linux。 ● Windows： Windows。 最小长度：**0** 最大长度：**128**
os_bit	String	操作系统位数 最小长度：**0** 最大长度：**128**
detect_result	String	云主机安全检测结果，包含如下4种。 ● undetected： 未检测。 ● clean： 无风险。 ● risk： 有风险。 ● scanning： 检测中。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
expire_time	Long	试用版到期时间（-1表示非试用版配额，当值不为-1时为试用版本过期时间） 最小值：**0** 最大值：**4824695185000**
charging_mode	String	收费模式，包含如下2种。 ● packet_cycle： 包年/包月。 ● on_demand： 按需。 最小长度：**1** 最大长度：**32**
resource_id	String	主机安全配额ID（UUID） 最小长度：**0** 最大长度：**128**
outside_host	Boolean	是否非华为云机器
group_id	String	服务器组ID 最小长度：**1** 最大长度：**128**
group_name	String	服务器组名称 最小长度：**1** 最大长度：**128**
policy_group_id	String	策略组ID 最小长度：**1** 最大长度：**128**
policy_group_name	String	策略组名称 最小长度：**1** 最大长度：**128**
asset	Integer	资产风险 最小值：**0** 最大值：**2097152**
vulnerability	Integer	漏洞风险总数，包含Linux软件漏洞、Windows系统漏洞、Web-CMS漏洞、应用漏洞 最小值：**0** 最大值：**2097152**
baseline	Integer	基线风险总数，包含配置风险、弱口令 最小值：**0** 最大值：**2097152**

参数	参数类型	描述
intrusion	Integer	入侵风险总数 最小值：**0** 最大值：**2097152**
asset_value	String	资产重要性，包含如下4种 ● important： 重要资产 ● common： 一般资产 ● test： 测试资产 最小长度：**0** 最大长度：**128**
labels	Array of strings	标签列表 最小长度：**0** 最大长度：**64** 数组长度：**0 - 100**
agent_create_time	Long	agent安装时间，采用时间戳，默认毫秒， 最小值：**0** 最大值：**4824695185000**
agent_update_time	Long	agent状态修改时间，采用时间戳，默认毫秒， 最小值：**0** 最大值：**4824695185000**
agent_version	String	agent版本 最小长度：**1** 最大长度：**32**
upgrade_status	String	升级状态，包含如下4种。 ● not_upgrade： 未升级，也就是默认状态，客户还没有给这台机器下发过升级。 ● upgrading： 正在升级中。 ● upgrade_failed： 升级失败。 ● upgrade_succeed： 升级成功。 最小长度：**1** 最大长度：**32**

参数	参数类型	描述
upgrade_result_code	String	升级失败原因，只有当 upgrade_status 为 upgrade_failed 时才显示，包含如下6种。 ● package_unavailable： 升级包解析失败，升级文件有错误。 ● network_access_timeout： 下载升级包失败，网络异常。 ● agent_offline： agent离线。 ● hostguard_abnormal： agent工作进程异常。 ● insufficient_disk_space： 磁盘空间不足。 ● failed_to_replace_file： 替换文件失败。 最小长度：**1** 最大长度：**32**
upgradable	Boolean	该服务器agent是否可升级
open_time	Long	开启防护时间，采用时间戳，默认毫秒， 最小值：**0** 最大值：**4824695185000**
protect_interrupt	Boolean	防护是否中断

请求示例

查询agent状态为在线的所有企业项目下的10台linux主机。

```
GET https://{endpoint}/v5/{project_id}/host-management/hosts?
limit=10&offset=0&agent_status=online&os_type=Linux&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

云服务器列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "agent_id" : "2758d2a61598fd9144cfa6b201049e7c0af8c3f1280cd24e3ec95a2f0811a2a2",
    "agent_status" : "online",
    "asset" : 0,
    "asset_value" : "common",
    "baseline" : 0,
    "charging_mode" : "packet_cycle",
    "detect_result" : "risk",
    "enterprise_project_id" : "all_granted_eps",
    "enterprise_project_name" : "default",
    "group_id" : "7c659ea3-006f-4687-9f1c-6d975d955f37",
    "group_name" : "default",
    "host_id" : "caa958ad-a481-4d46-b51e-6861b8864515",
    "host_name" : "ecs-r00431580-ubuntu",
```

```
    "host_status" : "ACTIVE",
    "intrusion" : 0,
    "expire_time" : -1,
    "os_bit" : "64",
    "os_type" : "Linux",
    "outside_host" : false,
    "policy_group_id" : "2758d2a61598fd9144cfa6b201049e7c0af8c3f1280cd24e3ec95a2f0811a2a2",
    "policy_group_name" : "wtp_ecs-r00431580-ubuntu(default)",
    "private_ip" : "192.168.0.182",
    "protect_status" : "opened",
    "protect_interrupt" : false,
    "public_ip" : "100.85.123.9",
    "resource_id" : "60f08ea4-c74e-4a45-be1c-3c057e373af2",
    "version" : "hss.version.wtp",
    "vulnerability" : 97,
    "labels" : [ "" ],
    "agent_create_time" : 0,
    "agent_update_time" : 0,
    "open_time" : 0
  } ]
}
```


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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListHostStatusSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListHostStatusRequest request = new ListHostStatusRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withVersion("<version>");
        request.withAgentStatus("<agent_status>");
        request.withDetectResult("<detect_result>");
        request.withHostName("<host_name>");
        request.withHostId("<host_id>");
        request.withHostStatus("<host_status>");
        request.withOsType("<os_type>");
```

```
        request.withPrivateIp("<private_ip>");
        request.withPublicIp("<public_ip>");
        request.withIpAddr("<ip_addr>");
        request.withProtectStatus("<protect_status>");
        request.withGroupId("<group_id>");
        request.withGroupName("<group_name>");
        request.withHasIntrusion(<has_intrusion>);
        request.withPolicyGroupId("<policy_group_id>");
        request.withPolicyGroupName("<policy_group_name>");
        request.withChargingMode("<charging_mode>");
        request.withRefresh(<refresh>);
        request.withAboveVersion(<above_version>);
        request.withOutsideHost(<outside_host>);
        request.withAssetValue("<asset_value>");
        request.withLabel("<label>");
        request.withServerGroup("<server_group>");
        request.withAgentUpgradable(<agent_upgradable>);
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListHostStatusResponse response = client.listHostStatus(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListHostStatusRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.version = "<version>"
        request.agent_status = "<agent_status>"
        request.detect_result = "<detect_result>"
        request.host_name = "<host_name>"
```

```
        request.host_id = "<host_id>"
        request.host_status = "<host_status>"
        request.os_type = "<os_type>"
        request.private_ip = "<private_ip>"
        request.public_ip = "<public_ip>"
        request.ip_addr = "<ip_addr>"
        request.protect_status = "<protect_status>"
        request.group_id = "<group_id>"
        request.group_name = "<group_name>"
        request.has_intrusion = <HasIntrusion>
        request.policy_group_id = "<policy_group_id>"
        request.policy_group_name = "<policy_group_name>"
        request.charging_mode = "<charging_mode>"
        request.refresh = <Refresh>
        request.above_version = <AboveVersion>
        request.outside_host = <OutsideHost>
        request.asset_value = "<asset_value>"
        request.label = "<label>"
        request.server_group = "<server_group>"
        request.agent_upgradable = <AgentUpgradable>
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_host_status(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListHostStatusRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    versionRequest:= "<version>"
    request.Version = &versionRequest
    agentStatusRequest:= "<agent_status>"
    request.AgentStatus = &agentStatusRequest
```

```
    detectResultRequest:= "<detect_result>"
    request.DetectResult = &detectResultRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    hostStatusRequest:= "<host_status>"
    request.HostStatus = &hostStatusRequest
    osTypeRequest:= "<os_type>"
    request.OsType = &osTypeRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
    publicIpRequest:= "<public_ip>"
    request.PublicIp = &publicIpRequest
    ipAddrRequest:= "<ip_addr>"
    request.IpAddr = &ipAddrRequest
    protectStatusRequest:= "<protect_status>"
    request.ProtectStatus = &protectStatusRequest
    groupIdRequest:= "<group_id>"
    request.GroupId = &groupIdRequest
    groupNameRequest:= "<group_name>"
    request.GroupName = &groupNameRequest
    hasIntrusionRequest:= <has_intrusion>
    request.HasIntrusion = &hasIntrusionRequest
    policyGroupIdRequest:= "<policy_group_id>"
    request.PolicyGroupId = &policyGroupIdRequest
    policyGroupNameRequest:= "<policy_group_name>"
    request.PolicyGroupName = &policyGroupNameRequest
    chargingModeRequest:= "<charging_mode>"
    request.ChargingMode = &chargingModeRequest
    refreshRequest:= <refresh>
    request.Refresh = &refreshRequest
    aboveVersionRequest:= <above_version>
    request.AboveVersion = &aboveVersionRequest
    outsideHostRequest:= <outside_host>
    request.OutsideHost = &outsideHostRequest
    assetValueRequest:= "<asset_value>"
    request.AssetValue = &assetValueRequest
    labelRequest:= "<label>"
    request.Label = &labelRequest
    serverGroupRequest:= "<server_group>"
    request.ServerGroup = &serverGroupRequest
    agentUpgradableRequest:= <agent_upgradable>
    request.AgentUpgradable = &agentUpgradableRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListHostStatus(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	云服务器列表

错误码

请参见**错误码**。

3.8.2 切换防护状态

功能介绍

切换防护状态

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/host-management/protection

表 3-257 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-258 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-259 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-260 请求 Body 参数

参数	是否必选	参数类型	描述
version	是	String	主机开通的版本，包含如下: ● hss.version.null： 无，代表关闭防护。 ● hss.version.basic： 基础版。 ● hss.version.advanced： 专业版。 ● hss.version.enterprise： 企业版。 ● hss.version.premium： 旗舰版。 ● hss.version.wtp： 网页防篡改版。 最小长度：**1** 最大长度：**128**
charging_mode	否	String	付费模式，当version不为“hss.version.null”时，则需必填该参数 ● packet_cycle : 包周期 ● on_demand : 按需 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
resource_id	否	String	HSS配额ID，不填该参数时，则随机选择对应版本配额 最小长度：**1** 最大长度：**128**
host_id_list	是	Array of strings	服务器列表 最小长度：**1** 最大长度：**128** 数组长度：**0 - 2097152**
tags	否	Array of **TagInfo** objects	资源标签列表 数组长度：**0 - 2097152**

表 3-261 TagInfo

参数	是否必选	参数类型	描述
key	否	String	键。最大长度128个unicode字符。 key不能为空 最小长度：**1** 最大长度：**128**
value	否	String	值。最大长度255个unicode字符。 最小长度：**1** 最大长度：**255**

响应参数

无

请求示例

切换ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器防护版本为企业版。

```
{
  "version" : "hss.version.enterprise",
  "charging_mode" : "packet_cycle",
  "resource_id" : "af4d08ad-2b60-4916-a5cf-8d6a23956dda",
  "host_id_list" : [ "71a15ecc-049f-4cca-bd28-5e90aca1817f" ],
  "tags" : [ {
    "key" : "服务",
    "value" : "hss"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

切换ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器防护版本为企业版。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class SwitchHostsProtectStatusSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        SwitchHostsProtectStatusRequest request = new SwitchHostsProtectStatusRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        SwitchHostsProtectStatusRequestInfo body = new SwitchHostsProtectStatusRequestInfo();
        List<TagInfo> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new TagInfo()
                .withKey("服务")
                .withValue("hss")
        );
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("71a15ecc-049f-4cca-bd28-5e90aca1817f");
        body.withTags(listbodyTags);
        body.withHostIdList(listbodyHostIdList);
        body.withResourceId("af4d08ad-2b60-4916-a5cf-8d6a23956dda");
        body.withChargingMode("packet_cycle");
        body.withVersion("hss.version.enterprise");
        request.withBody(body);
        try {
            SwitchHostsProtectStatusResponse response = client.switchHostsProtectStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
```

```
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

切换ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器防护版本为企业版。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = SwitchHostsProtectStatusRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listTagsbody = [
            TagInfo(
                key="服务",
                value="hss"
            )
        ]
        listHostIdListbody = [
            "71a15ecc-049f-4cca-bd28-5e90aca1817f"
        ]
        request.body = SwitchHostsProtectStatusRequestInfo(
            tags=listTagsbody,
            host_id_list=listHostIdListbody,
            resource_id="af4d08ad-2b60-4916-a5cf-8d6a23956dda",
            charging_mode="packet_cycle",
            version="hss.version.enterprise"
        )
        response = client.switch_hosts_protect_status(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

切换ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器防护版本为企业版。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.SwitchHostsProtectStatusRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    keyTags:= "服务"
    valueTags:= "hss"
    var listTagsbody = []model.TagInfo{
        {
            Key: &keyTags,
            Value: &valueTags,
        },
    }
    var listHostIdListbody = []string{
        "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    }
    resourceIdSwitchHostsProtectStatusRequestInfo:= "af4d08ad-2b60-4916-a5cf-8d6a23956dda"
    chargingModeSwitchHostsProtectStatusRequestInfo:= "packet_cycle"
    request.Body = &model.SwitchHostsProtectStatusRequestInfo{
        Tags: &listTagsbody,
        HostIdList: listHostIdListbody,
        ResourceId: &resourceIdSwitchHostsProtectStatusRequestInfo,
        ChargingMode: &chargingModeSwitchHostsProtectStatusRequestInfo,
        Version: "hss.version.enterprise",
    }
    response, err := client.SwitchHostsProtectStatus(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	successful response

错误码

请参见**错误码**。

3.8.3 查询服务器组列表

功能介绍

查询服务器组列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/host-management/groups

表 3-262 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-263 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**200** 缺省值：**10**
group_name	否	String	服务器组名称 最小长度：**1** 最大长度：**64**

请求参数

表 3-264 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-265 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数

参数	参数类型	描述
data_list	Array of HostGroupItem objects	服务器组列表 数组长度：**0 - 100**

表 3-266 HostGroupItem

参数	参数类型	描述
group_id	String	服务器组ID
group_name	String	服务器组名称
host_num	Integer	关联服务器数
risk_host_num	Integer	有风险服务器数
unprotect_host_num	Integer	未防护服务器数
host_id_list	Array of strings	服务器ID列表
is_outside	Boolean	是否是线下数据中心服务器组

请求示例

查询服务器组名称为test的服务器组。

```
GET https://{endpoint}/v5/{project_id}/host-management/groups?
offset=0&limit=200&enterprise_project_id=all_granted_eps&&group_name=test
```

响应示例

状态码： 200

服务器组列表

```
{
  "data_list" : [ {
    "group_id" : "36e59701-e2e7-4d56-b229-0db3bcf4e6e8",
    "group_name" : "test",
    "host_id_list" : [ "71a15ecc-049f-4cca-bd28-5e90aca1817f" ],
    "host_num" : 1,
    "risk_host_num" : 1,
    "unprotect_host_num" : 0
  } ],
  "total_num" : 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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListHostGroupsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListHostGroupsRequest request = new ListHostGroupsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withGroupName("<group_name>");
        try {
            ListHostGroupsResponse response = client.listHostGroups(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListHostGroupsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.offset = <offset>
        request.limit = <limit>
        request.group_name = "<group_name>"
        response = client.list_host_groups(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListHostGroupsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    groupNameRequest:= "<group_name>"
    request.GroupName = &groupNameRequest
    response, err := client.ListHostGroups(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	服务器组列表

错误码

请参见**错误码**。

3.8.4 创建服务器组

功能介绍

创建服务器组

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/host-management/groups

表 3-267 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-268 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-269 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**

表 3-270 请求 Body 参数

参数	是否必选	参数类型	描述
group_name	是	String	服务器组名称 最小长度：**1** 最大长度：**128**
host_id_list	是	Array of strings	服务器ID列表 最小长度：**1** 最大长度：**128** 数组长度：**1 - 10000**

响应参数

无

请求示例

创建名称为test的服务器组，服务器组中包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972。

```
POST https://{endpoint}/v5/{project_id}/host-management/groups

{
  "group_name" : "test",
  "host_id_list" : [ "15dac7fe-d81b-43bc-a4a7-4710fe673972" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

创建名称为test的服务器组，服务器组中包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class AddHostsGroupSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        AddHostsGroupRequest request = new AddHostsGroupRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
```

```
        AddHostsGroupRequestInfo body = new AddHostsGroupRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("15dac7fe-d81b-43bc-a4a7-4710fe673972");
        body.withHostIdList(listbodyHostIdList);
        body.withGroupName("test");
        request.withBody(body);
        try {
            AddHostsGroupResponse response = client.addHostsGroup(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

创建名称为test的服务器组，服务器组中包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = AddHostsGroupRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListbody = [
            "15dac7fe-d81b-43bc-a4a7-4710fe673972"
        ]
        request.body = AddHostsGroupRequestInfo(
            host_id_list=listHostIdListbody,
            group_name="test"
        )
        response = client.add_hosts_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建名称为test的服务器组，服务器组中包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddHostsGroupRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdListbody = []string{
        "15dac7fe-d81b-43bc-a4a7-4710fe673972",
    }
    request.Body = &model.AddHostsGroupRequestInfo{
        HostIdList: listHostIdListbody,
        GroupName: "test",
    }
    response, err := client.AddHostsGroup(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	success
400	参数非法

状态码	描述
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.8.5 编辑服务器组

功能介绍

编辑服务器组

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/host-management/groups

表 3-271 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-272 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-273 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**

表 3-274 请求 Body 参数

参数	是否必选	参数类型	描述
group_name	否	String	服务器组名称
group_id	是	String	服务器组ID
host_id_list	否	Array of strings	服务器ID列表

响应参数

无

请求示例

编辑名称为test的服务器组，服务器组ID为eca40dbe-27f7-4229-8f9d-a58213129fdc，服务器组包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972、21303c5b-36ad-4510-a1b0-cb4ac4c2875c。

```
PUT https://{endpoint}/v5/{project_id}/host-management/groups

{
  "group_id" : "eca40dbe-27f7-4229-8f9d-a58213129fdc",
  "group_name" : "test",
  "host_id_list" : [ "15dac7fe-d81b-43bc-a4a7-4710fe673972", "21303c5b-36ad-4510-a1b0-cb4ac4c2875c" ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

编辑名称为test的服务器组，服务器组ID为eca40dbe-27f7-4229-8f9d-a58213129fdc，服务器组包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972、21303c5b-36ad-4510-a1b0-cb4ac4c2875c。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeHostsGroupSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeHostsGroupRequest request = new ChangeHostsGroupRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ChangeHostsGroupRequestInfo body = new ChangeHostsGroupRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("15dac7fe-d81b-43bc-a4a7-4710fe673972");
        listbodyHostIdList.add("21303c5b-36ad-4510-a1b0-cb4ac4c2875c");
        body.withHostIdList(listbodyHostIdList);
        body.withGroupId("eca40dbe-27f7-4229-8f9d-a58213129fdc");
        body.withGroupName("test");
        request.withBody(body);
        try {
            ChangeHostsGroupResponse response = client.changeHostsGroup(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

编辑名称为test的服务器组，服务器组ID为eca40dbe-27f7-4229-8f9d-a58213129fdc，服务器组包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972、21303c5b-36ad-4510-a1b0-cb4ac4c2875c。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeHostsGroupRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListbody = [
            "15dac7fe-d81b-43bc-a4a7-4710fe673972",
            "21303c5b-36ad-4510-a1b0-cb4ac4c2875c"
        ]
        request.body = ChangeHostsGroupRequestInfo(
            host_id_list=listHostIdListbody,
            group_id="eca40dbe-27f7-4229-8f9d-a58213129fdc",
            group_name="test"
        )
        response = client.change_hosts_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

编辑名称为test的服务器组，服务器组ID为eca40dbe-27f7-4229-8f9d-a58213129fdc，服务器组包含的服务器ID为15dac7fe-d81b-43bc-a4a7-4710fe673972、21303c5b-36ad-4510-a1b0-cb4ac4c2875c。

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeHostsGroupRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdListbody = []string{
        "15dac7fe-d81b-43bc-a4a7-4710fe673972",
        "21303c5b-36ad-4510-a1b0-cb4ac4c2875c",
    }
    groupNameChangeHostsGroupRequestInfo:= "test"
    request.Body = &model.ChangeHostsGroupRequestInfo{
        HostIdList: &listHostIdListbody,
        GroupId: "eca40dbe-27f7-4229-8f9d-a58213129fdc",
        GroupName: &groupNameChangeHostsGroupRequestInfo,
    }
    response, err := client.ChangeHostsGroup(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	success
400	参数非法
401	鉴权失败
403	权限不足

状态码	描述
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.8.6 删除服务器组

功能介绍

删除服务器组

调用方法

请参见**如何调用API**。

URI

DELETE /v5/{project_id}/host-management/groups

表 3-275 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-276 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
group_id	是	String	服务器组ID

请求参数

表 3-277 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

无

请求示例

删除组id为34fcf861-402b-45c6-9b6a-13087791aae3的服务器组。

```
DELETE https://{endpoint}/v5/{project_id}/host-management/groups

{
  "group_id" : "34fcf861-402b-45c6-9b6a-13087791aae3"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

删除组id为34fcf861-402b-45c6-9b6a-13087791aae3的服务器组。

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

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;
```

```
public class DeleteHostsGroupSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteHostsGroupRequest request = new DeleteHostsGroupRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withGroupId("<group_id>");
        try {
            DeleteHostsGroupResponse response = client.deleteHostsGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

删除组id为34fcf861-402b-45c6-9b6a-13087791aae3的服务器组。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = DeleteHostsGroupRequest()
```

```
        request.enterprise_project_id = "<enterprise_project_id>"
        request.group_id = "<group_id>"
        response = client.delete_hosts_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

删除组id为34fcf861-402b-45c6-9b6a-13087791aae3的服务器组。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteHostsGroupRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.GroupId = "<group_id>"
    response, err := client.DeleteHostsGroup(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	success
400	参数非法
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.9 容器镜像

3.9.1 查询 swr 镜像仓库镜像列表

功能介绍

查询swr镜像仓库镜像列表,如果需要从swr同步最新镜像，需要先调用“从swr同步镜像”接口

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/swr-repository

表 3-278 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-279 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
namespace	否	String	组织名称 最小长度：**1** 最大长度：**256**
image_name	否	String	镜像名称 最小长度：**1** 最大长度：**128**
image_version	否	String	镜像版本 最小长度：**1** 最大长度：**64**
latest_version	否	Boolean	仅关注最新版本镜像 缺省值：**false**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 • local_image : 本地镜像 • instance_image : 企业镜像 最小长度：**1** 最大长度：**32**

参数	是否必选	参数类型	描述
scan_status	否	String	扫描状态，包含如下: ● unscan : 未扫描 ● success : 扫描完成 ● scanning : 扫描中 ● failed : 扫描失败 ● waiting_for_scan : 等待扫描 最小长度：**0** 最大长度：**32**
instance_name	否	String	企业镜像实例名称 最小长度：**0** 最大长度：**128**
image_size	否	Long	镜像大小 最小值：**0** 最大值：**2147483547** 缺省值：**0**
start_latest_update_time	否	Long	创建时间开始日期，时间单位毫秒（ms） 最小值：**0** 最大值：**4070880000000** 缺省值：**0**
end_latest_update_time	否	Long	创建时间结束日期，时间单位毫秒（ms） 最小值：**0** 最大值：**4070880000000** 缺省值：**0**
start_latest_scan_time	否	Long	最近一次扫描完成时间开始日期，时间单位 毫秒（ms） 最小值：**0** 最大值：**4070880000000** 缺省值：**0**
end_latest_scan_time	否	Long	最近一次扫描完成时间结束日期，时间单位 毫秒（ms） 最小值：**0** 最大值：**4070880000000** 缺省值：**0**
has_malicious_file	否	Boolean	是否存在恶意文件

参数	是否必选	参数类型	描述
has_unsafe_setting	否	Boolean	是否存在基线检查
has_vul	否	Boolean	是否存在软件漏洞
instance_id	否	String	企业仓库实例ID，swr共享版无需使用该参数 最小长度：**0** 最大长度：**128**

请求参数

表 3-280 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-281 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2147483547**
data_list	Array of **PrivateImageRepositoryInfo** objects	查询swr镜像仓库镜像列表 数组长度：**0 - 200**

表 3-282 PrivateImageRepositoryInfo

参数	参数类型	描述
id	Long	id 最小值：**0** 最大值：**2147483547**
namespace	String	命名空间 最小长度：**0** 最大长度：**64**
image_name	String	镜像名称 最小长度：**0** 最大长度：**128**
image_id	String	镜像id 最小长度：**0** 最大长度：**64**
image_digest	String	镜像digest 最小长度：**0** 最大长度：**128**
image_version	String	镜像版本 最小长度：**0** 最大长度：**64**
image_type	String	镜像类型，包含如下2种。 ● private_image： 私有镜像。 ● shared_image： 共享镜像。 最小长度：**0** 最大长度：**64**
latest_version	Boolean	是否是最新版本
scan_status	String	扫描状态，包含如下2种。 ● unscan： 未扫描。 ● success： 扫描完成。 ● scanning： 正在扫描。 ● failed： 扫描失败。 ● download_failed： 下载失败。 ● image_oversized： 镜像超大。 ● waiting_for_scan： 等待扫描。 最小长度：**0** 最大长度：**64**

参数	参数类型	描述
scan_failed_desc	String	扫描失败原因，包含如下14种。 ● "unknown_error" :未知错误 ● "authentication_failed":认证失败 ● "download_failed":镜像下载失败 ● "image_over_sized":镜像大小超限 ● "image_oversized":镜像超大 ● "failed_to_scan_vulnerability":漏洞扫描失败 ● "failed_to_scan_file":文件扫描失败 ● "failed_to_scan_software":软件扫描失败 ● "failed_to_check_sensitive_information":敏感信息核查失败 ● "failed_to_check_baseline":基线检查失败 ● "failed_to_check_software_compliance":软件合规检查失败 ● "failed_to_query_basic_image_information":基础镜像信息查询失败 ● "response_timed_out":响应超时 ● "database_error" : 数据库错误 ● "failed_to_send_the_scan_request" : 发送扫描请求失败 最小长度：**0** 最大长度：**64**
image_size	Long	镜像大小 最小值：**0** 最大值：**2147483547**
latest_update_time	Long	镜像版本最后更新时间，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**4070880000000**
latest_scan_time	Long	最近扫描时间，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**4070880000000**
vul_num	Integer	漏洞个数 最小值：**0** 最大值：**2147483647**
unsafe_setting_num	Integer	基线扫描未通过数 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
malicious_file_num	Integer	恶意文件数 最小值：**0** 最大值：**2147483647**
domain_name	String	拥有者（共享镜像参数） 最小长度：**0** 最大长度：**128**
shared_status	String	共享镜像状态，包含如下2种。 • expired： 已过期。 • effective： 有效。 最小长度：**1** 最大长度：**32**
scannable	Boolean	是否可扫描
instance_name	String	企业版镜像实例名称 最小长度：**0** 最大长度：**128**
instance_id	String	企业版镜像实例ID 最小长度：**0** 最大长度：**64**
instance_url	String	企业版镜像实例URL 最小长度：**0** 最大长度：**256**
association_images	Array of **AssociateImages** objects	多架构关联镜像信息 数组长度：**0 - 200**

表 3-283 AssociateImages

参数	参数类型	描述
image_name	String	镜像名称 最小长度：**0** 最大长度：**128**
image_version	String	镜像版本 最小长度：**0** 最大长度：**64**

参数	参数类型	描述
image_type	String	镜像类型 最小长度：**0** 最大长度：**64**
namespace	String	命名空间 最小长度：**0** 最大长度：**64**
image_digest	String	镜像digest 最小长度：**0** 最大长度：**128**
scan_status	String	扫描状态，包含如下2种。 ● unscan： 未扫描。 ● success： 扫描完成。 ● scanning： 正在扫描。 ● failed： 扫描失败。 ● download_failed： 下载失败。 ● image_oversized： 镜像超大。 ● waiting_for_scan： 等待扫描。 最小长度：**0** 最大长度：**32**

请求示例

查询镜像类型为私有镜像的swr镜像仓库镜像列表。

```
GET https://{endpoint}/v5/{project_id}/image/swr-repository?
offset=0&limit=50&image_type=private_image&latest_version=false&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

查询swr镜像仓库镜像列表,包括私有镜像列表和共享镜像列表（通过传参image_type控制）

```
{
  "total_num" : 3,
  "data_list" : [ {
    "id" : "111(私有镜像举例)",
    "image_digest" : "sha256:cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
    "image_id" : "cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
    "image_name" : "centos7",
    "image_size" : "1000 单位（Bytes）",
    "image_type" : "private_image",
    "image_version" : "common",
    "latest_scan_time" : 1691748641788,
    "latest_update_time" : 1687664346000,
    "latest_version" : false,
```

```
  "malicious_file_num" : 0,
  "namespace" : "aaa",
  "scan_status" : "success",
  "scannable" : true,
  "unsafe_setting_num" : 1,
  "vul_num" : 111,
  "instance_name" : "",
  "instance_id" : "",
  "instance_url" : ""
 }, {
  "id" : "222(共享镜像举例)",
  "domain_name" : "scc_cgs_XXX",
  "shared_status" : "effective",
  "image_digest" : "sha256:cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
  "image_id" : "cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
  "image_name" : "mysql",
  "image_size" : "1000 单位（Bytes）",
  "image_type" : "shared_image",
  "image_version" : "5.5",
  "latest_scan_time" : 1691748641788,
  "latest_update_time" : 1687664346000,
  "latest_version" : false,
  "malicious_file_num" : 0,
  "namespace" : "aaa",
  "scan_status" : "success",
  "scannable" : true,
  "unsafe_setting_num" : 1,
  "vul_num" : 111,
  "instance_name" : "",
  "instance_id" : "",
  "instance_url" : ""
 }, {
  "id" : "333(企业镜像举例)",
  "domain_name" : "scc_cgs_XXX",
  "shared_status" : "effective",
  "image_digest" : "sha256:cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
  "image_id" : "cebcdacde18091448a5040dc55bb1a9f6540b093db8XXXXXX",
  "image_name" : "mysql",
  "image_size" : "1000 单位（Bytes）",
  "image_type" : "shared_image",
  "image_version" : "5.5",
  "latest_scan_time" : 1691748641788,
  "latest_update_time" : 1687664346000,
  "latest_version" : false,
  "malicious_file_num" : 0,
  "namespace" : "aaa",
  "scan_status" : "success",
  "scannable" : true,
  "unsafe_setting_num" : 1,
  "vul_num" : 111,
  "instance_name" : "企业实例名称",
  "instance_id" : "",
  "instance_url" : ""
 } ]
}
```


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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListSwrImageRepositorySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListSwrImageRepositoryRequest request = new ListSwrImageRepositoryRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withNamespace("<namespace>");
        request.withImageName("<image_name>");
        request.withImageVersion("<image_version>");
        request.withLatestVersion(<latest_version>);
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withImageType("<image_type>");
        request.withScanStatus("<scan_status>");
        request.withInstanceName("<instance_name>");
        request.withImageSize(<image_size>L);
        request.withStartLatestUpdateTime(<start_latest_update_time>L);
        request.withEndLatestUpdateTime(<end_latest_update_time>L);
        request.withStartLatestScanTime(<start_latest_scan_time>L);
        request.withEndLatestScanTime(<end_latest_scan_time>L);
        request.withHasMaliciousFile(<has_malicious_file>);
        request.withHasUnsafeSetting(<has_unsafe_setting>);
        request.withHasVul(<has_vul>);
        request.withInstanceId("<instance_id>");
        try {
            ListSwrImageRepositoryResponse response = client.listSwrImageRepository(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListSwrImageRepositoryRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.namespace = "<namespace>"
        request.image_name = "<image_name>"
        request.image_version = "<image_version>"
        request.latest_version = <LatestVersion>
        request.offset = <offset>
        request.limit = <limit>
        request.image_type = "<image_type>"
        request.scan_status = "<scan_status>"
        request.instance_name = "<instance_name>"
        request.image_size = <image_size>
        request.start_latest_update_time = <start_latest_update_time>
        request.end_latest_update_time = <end_latest_update_time>
        request.start_latest_scan_time = <start_latest_scan_time>
        request.end_latest_scan_time = <end_latest_scan_time>
        request.has_malicious_file = <HasMaliciousFile>
        request.has_unsafe_setting = <HasUnsafeSetting>
        request.has_vul = <HasVul>
        request.instance_id = "<instance_id>"
        response = client.list_swr_image_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListSwrImageRepositoryRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    namespaceRequest:= "<namespace>"
    request.Namespace = &namespaceRequest
    imageNameRequest:= "<image_name>"
    request.ImageName = &imageNameRequest
    imageVersionRequest:= "<image_version>"
    request.ImageVersion = &imageVersionRequest
    latestVersionRequest:= <latest_version>
    request.LatestVersion = &latestVersionRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    request.ImageType = "<image_type>"
    scanStatusRequest:= "<scan_status>"
    request.ScanStatus = &scanStatusRequest
    instanceNameRequest:= "<instance_name>"
    request.InstanceName = &instanceNameRequest
    imageSizeRequest:= int64(<image_size>)
    request.ImageSize = &imageSizeRequest
    startLatestUpdateTimeRequest:= int64(<start_latest_update_time>)
    request.StartLatestUpdateTime = &startLatestUpdateTimeRequest
    endLatestUpdateTimeRequest:= int64(<end_latest_update_time>)
    request.EndLatestUpdateTime = &endLatestUpdateTimeRequest
    startLatestScanTimeRequest:= int64(<start_latest_scan_time>)
    request.StartLatestScanTime = &startLatestScanTimeRequest
    endLatestScanTimeRequest:= int64(<end_latest_scan_time>)
    request.EndLatestScanTime = &endLatestScanTimeRequest
    hasMaliciousFileRequest:= <has_malicious_file>
    request.HasMaliciousFile = &hasMaliciousFileRequest
    hasUnsafeSettingRequest:= <has_unsafe_setting>
    request.HasUnsafeSetting = &hasUnsafeSettingRequest
    hasVulRequest:= <has_vul>
    request.HasVul = &hasVulRequest
    instanceIdRequest:= "<instance_id>"
    request.InstanceId = &instanceIdRequest
    response, err := client.ListSwrImageRepository(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swr镜像仓库镜像列表,包括私有镜像列表和共享镜像列表（通过传参image_type控制）

错误码

请参见**错误码**。

3.9.2 镜像仓库镜像批量扫描

功能介绍

镜像仓库镜像批量扫描

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/image/batch-scan

表 3-284 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-285 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-286 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-287 请求 Body 参数

参数	是否必选	参数类型	描述
repo_type	否	String	仓库类型，现阶段接入了swr镜像仓库，包含如下: • SWR : SWR镜像仓库 最小长度：**1** 最大长度：**32**
image_info_list	否	Array of **BatchScanSwrImageInfo** objects	要扫描的镜像信息列表，operate_all参数为false时为必填 数组长度：**0 - 10241**
operate_all	否	Boolean	若为true全量查询，可筛选条件全部查询，若image_info_list为空，则必填
namespace	否	String	组织名称 最小长度：**1** 最大长度：**256**
image_name	否	String	镜像名称 最小长度：**1** 最大长度：**256**
image_version	否	String	镜像版本 最小长度：**1** 最大长度：**256**

参数	是否必选	参数类型	描述
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 最小长度：**1** 最大长度：**32**
scan_status	否	String	扫描状态，包含如下: • unscan : 未扫描 • success : 扫描完成 • scanning : 扫描中 • failed : 扫描失败 • download_failed : 下载失败 • image_oversized : 镜像超大 最小长度：**1** 最大长度：**32**
latest_version	否	Boolean	仅关注最新版本镜像
image_size	否	Long	镜像大小 最小值：**0** 最大值：**2147483547** 缺省值：**0**
start_latest_update_time	否	Long	创建时间开始日期，时间单位毫秒（ms） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
end_latest_update_time	否	Long	创建时间结束日期，时间单位毫秒（ms） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
start_latest_scan_time	否	Long	最近一次扫描完成时间开始日期，时间单位 毫秒（ms） 最小值：**0** 最大值：**2147483547** 缺省值：**0**

参数	是否必选	参数类型	描述
end_latest_scan_time	否	Long	最近一次扫描完成时间结束日期，时间单位 毫秒（ms） 最小值：**0** 最大值：**2147483547** 缺省值：**0**

表 3-288 BatchScanSwrImageInfo

参数	是否必选	参数类型	描述
namespace	否	String	命名空间 最小长度：**1** 最大长度：**64**
image_name	否	String	镜像名称 最小长度：**1** 最大长度：**128**
image_version	否	String	镜像版本 最小长度：**1** 最大长度：**128**
instance_id	否	String	企业实例ID 最小长度：**1** 最大长度：**128**
instance_url	否	String	下载企业镜像URL 最小长度：**0** 最大长度：**256**

响应参数

无

请求示例

- 类型为私有镜像的镜像进行批量扫描，body体传参镜像列表，operate_all没有传参，说明需要镜像列表批量扫描。

```
POST https://{endpoint}/v5/{project_id}/image/batch-scan

{
  "image_type" : "private_image",
  "image_info_list" : [ {
    "image_name" : "openjdk",
    "image_version" : "v8.8",
    "namespace" : "test"
  }, {
```

```
    "image_name" : "openjdk1",
    "image_version" : "v1.0",
    "namespace" : "test1"
  } ]
}
```

- 类型为私有镜像的镜像进行全量扫描，body体没有传参镜像列表，operate_all=true，说明需要镜像列表全量扫描。

```
POST https://{endpoint}/v5/{project_id}/image/batch-scan

{
  "image_type" : "private_image",
  "operate_all" : true
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

- 类型为私有镜像的镜像进行批量扫描，body体传参镜像列表，operate_all没有传参，说明需要镜像列表批量扫描。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class BatchScanSwrImageSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        BatchScanSwrImageRequest request = new BatchScanSwrImageRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        BatchScanPrivateImageRequestInfo body = new BatchScanPrivateImageRequestInfo();
        List<BatchScanSwrImageInfo> listbodyImageInfoList = new ArrayList<>();
```

```
        listbodyImageInfoList.add(
            new BatchScanSwrImageInfo()
                .withNamespace("test")
                .withImageName("openjdk")
                .withImageVersion("v8.8")
        );
        listbodyImageInfoList.add(
            new BatchScanSwrImageInfo()
                .withNamespace("test1")
                .withImageName("openjdk1")
                .withImageVersion("v1.0")
        );
        body.withImageType("private_image");
        body.withImageInfoList(listbodyImageInfoList);
        request.withBody(body);
        try {
            BatchScanSwrImageResponse response = client.batchScanSwrImage(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());
        }
    }
}
```

- 类型为私有镜像的镜像进行全量扫描，body体没有传参镜像列表，operate_all=true，说明需要镜像列表全量扫描。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class BatchScanSwrImageSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        BatchScanSwrImageRequest request = new BatchScanSwrImageRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        BatchScanPrivateImageRequestInfo body = new BatchScanPrivateImageRequestInfo();
```

```
        body.withImageType("private_image");
        body.withOperateAll(true);
        request.withBody(body);
        try {
            BatchScanSwrImageResponse response = client.batchScanSwrImage(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

- 类型为私有镜像的镜像进行批量扫描，body体传参镜像列表，operate_all没有传参，说明需要镜像列表批量扫描。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = BatchScanSwrImageRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listImageInfoListbody = [
            BatchScanSwrImageInfo(
                namespace="test",
                image_name="openjdk",
                image_version="v8.8"
            ),
            BatchScanSwrImageInfo(
                namespace="test1",
                image_name="openjdk1",
                image_version="v1.0"
            )
        ]
        request.body = BatchScanPrivateImageRequestInfo(
            image_type="private_image",
            image_info_list=listImageInfoListbody
        )
```

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

- 类型为私有镜像的镜像进行全量扫描，body体没有传参镜像列表，operate_all=true，说明需要镜像列表全量扫描。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = BatchScanSwrImageRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.body = BatchScanPrivateImageRequestInfo(
            image_type="private_image",
            operate_all=True
        )
        response = client.batch_scan_swr_image(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

- 类型为私有镜像的镜像进行批量扫描，body体传参镜像列表，operate_all没有传参，说明需要镜像列表批量扫描。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchScanSwrImageRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    namespaceImageInfoList:= "test"
    imageNameImageInfoList:= "openjdk"
    imageVersionImageInfoList:= "v8.8"
    namespaceImageInfoList1:= "test1"
    imageNameImageInfoList1:= "openjdk1"
    imageVersionImageInfoList1:= "v1.0"
    var listImageInfoListbody = []model.BatchScanSwrImageInfo{
        {
            Namespace: &namespaceImageInfoList,
            ImageName: &imageNameImageInfoList,
            ImageVersion: &imageVersionImageInfoList,
        },
        {
            Namespace: &namespaceImageInfoList1,
            ImageName: &imageNameImageInfoList1,
            ImageVersion: &imageVersionImageInfoList1,
        },
    }
    request.Body = &model.BatchScanPrivateImageRequestInfo{
        ImageType: "private_image",
        ImageInfoList: &listImageInfoListbody,
    }
    response, err := client.BatchScanSwrImage(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 类型为私有镜像的镜像进行全量扫描，body体没有传参镜像列表，operate_all=true，说明需要镜像列表全量扫描。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchScanSwrImageRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    operateAllBatchScanPrivateImageRequestInfo:= true
    request.Body = &model.BatchScanPrivateImageRequestInfo{
        ImageType: "private_image",
        OperateAll: &operateAllBatchScanPrivateImageRequestInfo,
    }
    response, err := client.BatchScanSwrImage(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	successful response

错误码

请参见**错误码**。

3.9.3 本地镜像列表查询

功能介绍

本地镜像列表查询

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/local-repositories

表 3-289 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-290 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
image_name	否	String	镜像名称 最小长度：**1** 最大长度：**128**
image_version	否	String	镜像版本 最小长度：**1** 最大长度：**64**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
scan_status	否	String	扫描状态，包含如下: ● unscan : 未扫描 ● success : 扫描完成 ● scanning : 扫描中 ● failed : 扫描失败 ● waiting_for_scan : 等待扫描 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
local_image_type	否	String	镜像类型，包含如下: ● other_image : 非SWR镜像 ● swr_image : SWR镜像 最小长度：**0** 最大长度：**64**
image_size	否	Long	镜像大小，单位字节 最小值：**0** 最大值：**2147483547** 缺省值：**0**
start_latest_update_time	否	Long	最近更新时间搜索开始日期，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
end_latest_update_time	否	Long	最近更新时间搜索结束日期，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
start_latest_scan_time	否	Long	最近一次扫描完成时间搜索开始日期，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
end_latest_scan_time	否	Long	最近一次扫描完成时间搜索结束日期，时间单位 毫秒（ ms ） 最小值：**0** 最大值：**2147483547** 缺省值：**0**
has_vul	否	Boolean	是否存在软件漏洞
host_name	否	String	本地镜像所关联服务器的名称 最小长度：**0** 最大长度：**128**
host_id	否	String	本地镜像所关联服务器的ID 最小长度：**0** 最大长度：**128**

参数	是否必选	参数类型	描述
host_ip	否	String	本地镜像所关联服务器的IP（公网或私网） 最小长度：**0** 最大长度：**128**
container_id	否	String	本地镜像所关联容器的ID 最小长度：**0** 最大长度：**128**
container_name	否	String	本地镜像所关联容器的名称 最小长度：**0** 最大长度：**128**
pod_id	否	String	本地镜像所关联Pod的ID 最小长度：**0** 最大长度：**128**
pod_name	否	String	本地镜像所关联Pod的名称 最小长度：**0** 最大长度：**128**
app_name	否	String	本地镜像所关联软件的名称 最小长度：**0** 最大长度：**128**

请求参数

表 3-291 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。 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-292 响应 Body 参数

参数	参数类型	描述
total_num	Integer	本地镜像总数 最小值：**0** 最大值：**2147483547**
data_list	Array of **ImageLocalInfo** objects	本地镜像数据列表 数组长度：**1 - 100000**

表 3-293 ImageLocalInfo

参数	参数类型	描述
image_name	String	镜像名称 最小长度：**0** 最大长度：**256**
image_id	String	镜像ID 最小长度：**0** 最大长度：**64**
image_digest	String	镜像digest 最小长度：**0** 最大长度：**128**
image_version	String	镜像版本 最小长度：**0** 最大长度：**256**
local_image_type	String	本地镜像类型 最小长度：**0** 最大长度：**256**
scan_status	String	扫描状态，包含如下： • unscan：未扫描 • success：扫描完成 • scanning：正在扫描 • failed：扫描失败 • waiting：等待扫描 最小长度：**0** 最大长度：**256**

参数	参数类型	描述
image_size	Long	镜像大小，单位字节 最小值：**0** 最大值：**9223372036854775807**
latest_update_time	Long	镜像版本最后更新时间，时间单位毫秒（ms） 最小值：**0** 最大值：**4070880000000**
latest_scan_time	Long	最近扫描时间，时间单位毫秒（ms） 最小值：**0** 最大值：**4070880000000**
vul_num	Long	漏洞个数 最小值：**0** 最大值：**9223372036854775807**
unsafe_setting_num	Long	基线扫描未通过数 最小值：**0** 最大值：**9223372036854775807**
malicious_file_num	Long	恶意文件数 最小值：**0** 最大值：**9223372036854775807**
host_num	Long	关联主机数 最小值：**0** 最大值：**9223372036854775807**
container_num	Long	关联容器数 最小值：**0** 最大值：**9223372036854775807**
component_num	Long	关联组件数 最小值：**0** 最大值：**9223372036854775807**

参数	参数类型	描述
scan_failed_desc	String	扫描失败原因，包含如下10种。 ● "unknown_error":未知错误 ● "failed_to_match_agent":对应主机未开启容器版防护或agent离线 ● "create_container_failed":创建容器失败 ● "get_container_info_failed":获取容器信息失败 ● "docker_offline":docker引擎不在线 ● "get_docker_root_failed":获取容器根文件系统失败 ● "image_not_exist_or_docker_api_fault":镜像不存在或docker接口错误 ● "huge_image":超大镜像 ● "docker_root_in_nfs":容器根目录位于网络挂载 ● "response_timed_out":响应超时 最小长度：**0** 最大长度：**64**

请求示例

查询前10条本地镜像信息列表

```
GET https://{endpoint}/v5/{project_id}/image/local-repositories?offset=0&limit=10
```

响应示例

状态码： 200

successful response

```
{
  "data_list" : [ {
    "image_id" : "f757deea-781e-45ec-90ec-f199249890df",
    "image_name" : "webshell-ljx",
    "image_version" : "v1",
    "image_digest" : "sha256:ce0b5d91b072730d0bc9518f11efd07eb7fdb9f43251e11a96cab5b1918b7044",
    "local_image_type" : "swr_image",
    "scan_status" : "success",
    "image_size" : 215304488,
    "latest_update_time" : 1697509433000,
    "latest_scan_time" : 1709973506292,
    "host_num" : 0,
    "container_num" : 5,
    "component_num" : 146,
    "vul_num" : 77,
    "host_name" : "myhost",
    "host_id" : "9ad79426-992c-4be4-a2d1-dfd3a75b7c14",
    "agent_id" : "1c1d073c5fc403eb0d9c3088bc49da4e015586fd4864513a2fd81afedce282d4",
    "severity_level" : "High"
  } ],
```

```
  "total_num" : 1
}
```

状态码

状态码	描述
200	successful response

错误码

请参见**错误码**。

3.9.4 查询镜像的漏洞信息

功能介绍

查询镜像的漏洞信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/{image_id}/vulnerabilities

表 3-294 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
image_id	是	String	镜像id 最小长度：**0** 最大长度：**128**

表 3-295 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 • local_image : 本地镜像 • instance_image : 企业镜像 最小长度：**1** 最大长度：**32**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
instance_id	否	String	企业仓库实例ID，swr共享版无需使用该参数 最小长度：**0** 最大长度：**128**
namespace	是	String	组织名称 最小长度：**0** 最大长度：**64**
image_name	是	String	镜像名称 最小长度：**0** 最大长度：**128**
tag_name	是	String	镜像版本 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
repair_necessity	否	String	危险程度，包含如下3种。 ● immediate_repair： 高危。 ● delay_repair： 中危。 ● not_needed_repair： 低危。 最小长度：**0** 最大长度：**32**
vul_id	否	String	漏洞ID（支持模糊查询） 最小长度：**0** 最大长度：**64**
app_name	否	String	软件名 最小长度：**0** 最大长度：**64**
type	否	String	漏洞类型，包含如下： -linux_vul : linux漏洞 -app_vul : 应用漏洞 最小长度：**0** 最大长度：**32**

请求参数

表 3-296 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-297 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2147483547**
data_list	Array of **ImageVulInfo** objects	镜像的漏洞列表 数组长度：**0 - 10241**

表 3-298 ImageVulInfo

参数	参数类型	描述
vul_id	String	漏洞id 最小长度：**0** 最大长度：**128**
repair_necessity	String	修复紧急度，包含如下3种。 • immediate_repair ： 高危。 • delay_repair ： 中危。 • not_needed_repair ： 低危。 最小长度：**0** 最大长度：**128**
description	String	漏洞描述 最小长度：**0** 最大长度：**128**
position	String	漏洞所在镜像层 最小长度：**0** 最大长度：**128**
app_name	String	漏洞的软件名称 最小长度：**0** 最大长度：**128**
app_path	String	应用软件的路径（只有应用漏洞有该字段） 最小长度：**1** 最大长度：**512**
version	String	软件版本 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
solution	String	解决方案 最小长度：**0** 最大长度：**256**
url	String	补丁地址 最小长度：**0** 最大长度：**128**

请求示例

查询私有镜像中命名空间为scc_hss_container，镜像名称为apptest，镜像版本为V1的漏洞信息。

```
GET https://{endpoint}/v5/{project_id}/image/{image_id}/vulnerabilities?
limit=10&offset=0&namespace=scc_hss_container&tag_name=v1&image_name=apptest&image_type=private
_image&type=linux_vul&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

镜像漏洞信息列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "app_name" : "xz-lib",
    "description" : "online",
    "position" : "sha256:74ddd0ec08fa43dXXXX",
    "repair_necessity" : "delay_repair",
    "solution" : "To upgrade the affected software",
    "url" : "https://access.redhat.com/errata/RHSAXXX",
    "version" : "5.2.4-3.el8",
    "vul_id" : "RHSA-2022:49XX"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListImageVulnerabilitiesSolution {
```

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListImageVulnerabilitiesRequest request = new ListImageVulnerabilitiesRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withImageType("<image_type>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withInstanceId("<instance_id>");
        request.withNamespace("<namespace>");
        request.withImageName("<image_name>");
        request.withTagName("<tag_name>");
        request.withRepairNecessity("<repair_necessity>");
        request.withVulId("<vul_id>");
        request.withAppName("<app_name>");
        request.withType("<type>");
        try {
            ListImageVulnerabilitiesResponse response = client.listImageVulnerabilities(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

    client = HssClient.new_builder() \
```

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

    try:
        request = ListImageVulnerabilitiesRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.image_type = "<image_type>"
        request.offset = <offset>
        request.limit = <limit>
        request.instance_id = "<instance_id>"
        request.namespace = "<namespace>"
        request.image_name = "<image_name>"
        request.tag_name = "<tag_name>"
        request.repair_necessity = "<repair_necessity>"
        request.vul_id = "<vul_id>"
        request.app_name = "<app_name>"
        request.type = "<type>"
        response = client.list_image_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)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListImageVulnerabilitiesRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.ImageType = "<image_type>"
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    instanceIdRequest:= "<instance_id>"
    request.InstanceId = &instanceIdRequest
    request.Namespace = "<namespace>"
```

```
    request.ImageName = "<image_name>"
    request.TagName = "<tag_name>"
    repairNecessityRequest:= "<repair_necessity>"
    request.RepairNecessity = &repairNecessityRequest
    vulIdRequest:= "<vul_id>"
    request.VulId = &vulIdRequest
    appNameRequest:= "<app_name>"
    request.AppName = &appNameRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    response, err := client.ListImageVulnerabilities(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	镜像漏洞信息列表

错误码

请参见**错误码**。

3.9.5 漏洞对应 cve 信息

功能介绍

漏洞对应cve信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/vulnerability/{vul_id}/cve

表 3-299 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

参数	是否必选	参数类型	描述
vul_id	是	String	漏洞ID 最小长度：**0** 最大长度：**64**

表 3-300 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**

请求参数

表 3-301 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-302 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2147483547**
data_list	Array of **ImageVulCveInfo** objects	列表 数组长度：**1 - 10000**

表 3-303 ImageVulCveInfo

参数	参数类型	描述
cve_id	String	cve id 最小长度：**0** 最大长度：**32**
cvss_score	Float	CVSS分数 最小值：**0** 最大值：**100**
publish_time	Long	公布时间，时间单位 毫秒（ms） 最小值：**0** 最大值：**2147483547**
description	String	cve描述 最小长度：**0** 最大长度：**65534**

请求示例

查询漏洞id为vul_id的漏洞对应cve信息。

```
GET https://{endpoint}/v5/{project_id}/image/vulnerability/{vul_id}/cve?
offset=0&limit=200&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

漏洞对应cve信息列表请求成功

```
{
  "total_num" : 1,
```

```
 "data_list" : [ {
   "cve_id" : "CVE-2021-45960",
   "cvss_score" : 8.8,
   "description" : "In Expat (aka libexpat) XXXX",
   "publish_time" : 1641035700000
 } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListVulnerabilityCveSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListVulnerabilityCveRequest request = new ListVulnerabilityCveRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListVulnerabilityCveResponse response = client.listVulnerabilityCve(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListVulnerabilityCveRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_vulnerability_cve(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
        Build())

    request := &model.ListVulnerabilityCveRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListVulnerabilityCve(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	漏洞对应cve信息列表请求成功

错误码

请参见**错误码**。

3.9.6 从 SWR 服务同步镜像列表

功能介绍

从SWR服务同步镜像列表

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/image/synchronize

表 3-304 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-305 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-306 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

表 3-307 请求 Body 参数

参数	是否必选	参数类型	描述
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 最小长度：**1** 最大长度：**32**

响应参数

状态码： 200

表 3-308 响应 Body 参数

参数	参数类型	描述
error_code	Integer	错误编码 最小值：**0** 最大值：**10**
error_description	String	错误描述 最小长度：**1** 最大长度：**128**

请求示例

从swr服务同步镜像，类型为私有镜像或者共享镜像。

```
POST https://{endpoint}/v5/{project_id}/image/synchronize

{
  "image_type" : "private_image"
}
```

响应示例

状态码： 200

请求成功

```
{
  "error_code" : 0,
  "error_description" : "success"
}
```

SDK 代码示例

SDK代码示例如下。

Java

从swr服务同步镜像，类型为私有镜像或者共享镜像。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class RunImageSynchronizeSolution {

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

```
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        RunImageSynchronizeRequest request = new RunImageSynchronizeRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        RunImageSynchronizeRequestInfo body = new RunImageSynchronizeRequestInfo();
        body.withImageType("private_image");
        request.withBody(body);
        try {
            RunImageSynchronizeResponse response = client.runImageSynchronize(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

从swr服务同步镜像，类型为私有镜像或者共享镜像。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = RunImageSynchronizeRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.body = RunImageSynchronizeRequestInfo(
            image_type="private_image"
        )
```

```
        response = client.run_image_synchronize(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

从swr服务同步镜像，类型为私有镜像或者共享镜像。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RunImageSynchronizeRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.Body = &model.RunImageSynchronizeRequestInfo{
        ImageType: "private_image",
    }
    response, err := client.RunImageSynchronize(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功

错误码

请参见**错误码**。

3.9.7 查询镜像安全配置检测结果列表

功能介绍

查询镜像安全配置检测结果列表,当前支持检测CentOS 7、Debian 10、EulerOS和Ubuntu16镜像的系统配置项、SSH应用配置项。

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/baseline/risk-configs

表 3-309 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-310 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

参数	是否必选	参数类型	描述
image_type	是	String	镜像类型，包含如下: ● private_image : 私有镜像仓库 ● shared_image : 共享镜像仓库 ● local_image : 本地镜像 ● instance_image : 企业镜像 最小长度：**1** 最大长度：**32**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
namespace	否	String	组织名称 最小长度：**0** 最大长度：**64**
image_name	否	String	镜像名称 最小长度：**0** 最大长度：**128**
image_version	否	String	镜像版本名称 最小长度：**0** 最大长度：**64**
check_name	否	String	基线名称 最小长度：**0** 最大长度：**256**
severity	否	String	风险等级，包含如下: ● Security : 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
standard	否	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 华为标准 ● qt_standard : 青腾标准 最小长度：**0** 最大长度：**256**
instance_id	否	String	企业仓库实例ID，swr共享版无需使用该参数 最小长度：**0** 最大长度：**128**

请求参数

表 3-311 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-312 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
data_list	Array of **ImageRiskConfigsInfoResponseInfo** objects	配置检测列表 数组长度：**0 - 2147483647**

表 3-313 ImageRiskConfigsInfoResponseInfo

参数	参数类型	描述
severity	String	风险等级，包含如下: • Security : 安全 • Low : 低危 • Medium : 中危 • High : 高危 最小长度：**1** 最大长度：**32**
check_name	String	基线名称 最小长度：**0** 最大长度：**256**
check_type	String	基线类型 最小长度：**0** 最大长度：**256**
standard	String	标准类型，包含如下: • cn_standard : 等保合规标准 • hw_standard : 华为标准 • qt_standard : 青腾标准 最小长度：**1** 最大长度：**16**
check_rule_num	Integer	检查项数量 最小值：**0** 最大值：**2097152**
failed_rule_num	Integer	风险项数量 最小值：**0** 最大值：**2097152**
check_type_desc	String	基线描述信息 最小长度：**0** 最大长度：**65534**

请求示例

查询私有镜像中命名空间为scc_hss_container，镜像名称为euleros，镜像版本为2.2的镜像安全配置检测结果列表。

```
GET https://{endpoint}/v5/{project_id}/image/baseline/risk-configs?
offset=0&limit=200&image_type=private_image&namespace=scc_hss_container&image_name=euleros/
test&image_version=2.2.6&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

镜像配置检测结果列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "check_name" : "CentOS 7",
    "check_rule_num" : 3,
    "check_type" : 3,
    "check_type_desc" : "本规范着重于从XXX",
    "failed_rule_num" : 0,
    "severity" : "Low",
    "standard" : "cn_standard"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListImageRiskConfigsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
```

```
            .withRegion(HssRegion.valueOf("<YOUR REGION>"))
            .build();
        ListImageRiskConfigsRequest request = new ListImageRiskConfigsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withImageType("<image_type>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withNamespace("<namespace>");
        request.withImageName("<image_name>");
        request.withImageVersion("<image_version>");
        request.withCheckName("<check_name>");
        request.withSeverity("<severity>");
        request.withStandard("<standard>");
        request.withInstanceId("<instance_id>");
        try {
            ListImageRiskConfigsResponse response = client.listImageRiskConfigs(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListImageRiskConfigsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.image_type = "<image_type>"
        request.offset = <offset>
        request.limit = <limit>
        request.namespace = "<namespace>"
        request.image_name = "<image_name>"
        request.image_version = "<image_version>"
        request.check_name = "<check_name>"
        request.severity = "<severity>"
        request.standard = "<standard>"
```

```
        request.instance_id = "<instance_id>"
        response = client.list_image_risk_configs(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListImageRiskConfigsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.ImageType = "<image_type>"
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    namespaceRequest:= "<namespace>"
    request.Namespace = &namespaceRequest
    imageNameRequest:= "<image_name>"
    request.ImageName = &imageNameRequest
    imageVersionRequest:= "<image_version>"
    request.ImageVersion = &imageVersionRequest
    checkNameRequest:= "<check_name>"
    request.CheckName = &checkNameRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    standardRequest:= "<standard>"
    request.Standard = &standardRequest
    instanceIdRequest:= "<instance_id>"
    request.InstanceId = &instanceIdRequest
    response, err := client.ListImageRiskConfigs(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	镜像配置检测结果列表

错误码

请参见**错误码**。

3.9.8 查询镜像指定安全配置项的检查项列表

功能介绍

查询镜像指定安全配置项的检查项列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/baseline/risk-configs/{check_name}/rules

表 3-314 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
check_name	是	String	基线名称 最小长度：**0** 最大长度：**128**

表 3-315 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 • local_image : 本地镜像 • instance_image : 企业镜像 最小长度：**1** 最大长度：**32**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
limit	否	Integer	每页显示数量 最小值：**10** 最大值：**200** 缺省值：**10**
namespace	否	String	组织名称（没有镜像相关信息时，表示查询所有镜像） 最小长度：**0** 最大长度：**64**
image_name	否	String	镜像名称 最小长度：**0** 最大长度：**128**
image_version	否	String	镜像版本名称 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
standard	是	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 华为标准 ● qt_standard : 青腾标准 最小长度：**0** 最大长度：**32**
result_type	否	String	结果类型，包含如下: ● pass： 已通过 ● failed : 未通过 缺省值：**unhandled** 最小长度：**0** 最大长度：**64**
check_rule_name	否	String	检查项名称，支持模糊匹配 最小长度：**0** 最大长度：**2048**
severity	否	String	风险等级，包含如下: ● Security : 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 ● Critical : 危急 最小长度：**0** 最大长度：**255**
instance_id	否	String	企业仓库实例ID，swr共享版无需使用该参数 最小长度：**0** 最大长度：**128**

请求参数

表 3-316 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） 最小长度：**1** 最大长度：**32768**
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-317 响应 Body 参数

参数	参数类型	描述
total_num	Integer	风险总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **ImageRiskConfigsCheckRulesResponseInfo** objects	数据列表 数组长度：**0 - 2147483647**

表 3-318 ImageRiskConfigsCheckRulesResponseInfo

参数	参数类型	描述
severity	String	风险等级，包含如下: • Security : 安全 • Low : 低危 • Medium : 中危 • High : 高危 最小长度：**0** 最大长度：**255**

参数	参数类型	描述
check_name	String	基线名称 最小长度：**0** 最大长度：**256**
check_type	String	基线类型 最小长度：**0** 最大长度：**256**
standard	String	标准类型，包含如下: ● cn_standard : 等保合规标准 ● hw_standard : 华为标准 ● qt_standard : 青腾标准 最小长度：**0** 最大长度：**16**
check_rule_name	String	检查项 最小长度：**0** 最大长度：**2048**
check_rule_id	String	检查项ID 最小长度：**0** 最大长度：**64**
scan_result	String	检测结果，包含如下: ● pass 通过 ● failed 未通过 最小长度：**0** 最大长度：**64**

请求示例

查询所属组织为aaa，镜像名称为centos7，镜像版本为common的私有镜像并且标准类型为华为规范的指定安全配置项的检查项列表。

```
GET https://{endpoint}/v5/{project_id}/image/baseline/risk-configs/{check_name}/rules?
offset=0&limit=200&image_type=private_image&namespace=aaa&image_name=centos7/
test&image_version=common&standard=hw_standard&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

指定安全配置项的检查项列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "check_rule_id" : "1.1",
```

```
    "check_rule_name" : "规则：口令锁定策略.",
    "check_name" : "CentOS 7",
    "check_type" : "CentOS 7",
    "standard" : "hw_standard",
    "scan_result" : "failed",
    "severity" : "High"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListImageRiskConfigRulesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListImageRiskConfigRulesRequest request = new ListImageRiskConfigRulesRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withImageType("<image_type>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withNamespace("<namespace>");
        request.withImageName("<image_name>");
        request.withImageVersion("<image_version>");
        request.withStandard("<standard>");
        request.withResultType("<result_type>");
        request.withCheckRuleName("<check_rule_name>");
        request.withSeverity("<severity>");
        request.withInstanceId("<instance_id>");
        try {
            ListImageRiskConfigRulesResponse response = client.listImageRiskConfigRules(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListImageRiskConfigRulesRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.image_type = "<image_type>"
        request.offset = <offset>
        request.limit = <limit>
        request.namespace = "<namespace>"
        request.image_name = "<image_name>"
        request.image_version = "<image_version>"
        request.standard = "<standard>"
        request.result_type = "<result_type>"
        request.check_rule_name = "<check_rule_name>"
        request.severity = "<severity>"
        request.instance_id = "<instance_id>"
        response = client.list_image_risk_config_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

func main() {
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListImageRiskConfigRulesRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.ImageType = "<image_type>"
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    namespaceRequest:= "<namespace>"
    request.Namespace = &namespaceRequest
    imageNameRequest:= "<image_name>"
    request.ImageName = &imageNameRequest
    imageVersionRequest:= "<image_version>"
    request.ImageVersion = &imageVersionRequest
    request.Standard = "<standard>"
    resultTypeRequest:= "<result_type>"
    request.ResultType = &resultTypeRequest
    checkRuleNameRequest:= "<check_rule_name>"
    request.CheckRuleName = &checkRuleNameRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    instanceIdRequest:= "<instance_id>"
    request.InstanceId = &instanceIdRequest
    response, err := client.ListImageRiskConfigRules(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	指定安全配置项的检查项列表

错误码

请参见**错误码**。

3.9.9 查询镜像配置检查项检测报告

功能介绍

查询镜像配置检查项检测报告

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/image/baseline/check-rule/detail

表 3-319 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-320 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
image_type	是	String	镜像类型，包含如下: • private_image : 私有镜像仓库 • shared_image : 共享镜像仓库 • local_image : 本地镜像 • instance_image : 企业镜像 最小长度：**1** 最大长度：**32**
namespace	否	String	组织名称（没有镜像相关信息时，表示查询所有镜像） 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
image_name	否	String	镜像名称 最小长度：**0** 最大长度：**128**
image_version	否	String	镜像版本名称 最小长度：**0** 最大长度：**64**
check_name	是	String	基线名称 最小长度：**0** 最大长度：**255**
check_type	是	String	基线类型 最小长度：**0** 最大长度：**255**
check_rule_id	是	String	检查项id 最小长度：**0** 最大长度：**255**
standard	是	String	标准类型，包含如下: • cn_standard : 等保合规标准 • hw_standard : 华为标准 • qt_standard : 青腾标准 最小长度：**0** 最大长度：**32**
instance_id	否	String	企业仓库实例ID，swr共享版无需使用该参数 最小长度：**0** 最大长度：**128**

请求参数

表 3-321 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） 最小长度：**1** 最大长度：**32768**

参数	是否必选	参数类型	描述
region	否	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-322 响应 Body 参数

参数	参数类型	描述
description	String	检查项描述 最小长度：**0** 最大长度：**2048**
reference	String	参考 最小长度：**0** 最大长度：**255**
audit	String	审计描述 最小长度：**0** 最大长度：**65534**
remediation	String	修改建议 最小长度：**0** 最大长度：**65534**
check_info_list	Array of **ImageCheckRuleCheckCaseResponseInfo** objects	检测用例信息 数组长度：**0 - 2147483647**

表 3-323 ImageCheckRuleCheckCaseResponseInfo

参数	参数类型	描述
check_description	String	检测用例描述 最小长度：**0** 最大长度：**65534**

参数	参数类型	描述
current_value	String	当前结果 最小长度：**0** 最大长度：**65534**
suggest_value	String	期待结果 最小长度：**0** 最大长度：**65534**

请求示例

查询所属组织为aaa，镜像名称为centos7，镜像版本为common的私有镜像、基线名称为SSH、检测项id为1.12并且标准类型为华为规范的配置检查项检测报告。

```
GET https://{endpoint}/v5/{project_id}/image/baseline/check-rule/detail?image_type=private_image&namespace=aaa&image_name=centos7&image_version=common&check_rule_id=1.12&standard=hw_standard&check_type=SSH&check_name=SSH&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

配置检查项检测报告

```
{
  "audit" : "检查配置文件/etc/pam.d/system",
  "check_info_list" : [ {
    "check_description" : "检查配置文件/etc/pam.d/system-auth"
  }, {
    "current_value" : ""
  }, {
    "suggest_value" : "每个文件都配置auth required "
  } ],
  "description" : "The two options ClientAliveInterval and ClientAliveCountMax control the timeout of SSH sessions. The ClientAliveInterval parameter sets a timeout interval in seconds after which if no data has been received from the client, sshd will send a message through the encrypted channel to request a response from the client. The ClientAliveCountMax parameter sets the number of client alive messages which may be sent without sshd receiving any messages back from the client. For example, if the ClientAliveInterval is set to 15s and the ClientAliveCountMax is set to 3, unresponsive SSH clients will be disconnected after approximately 45s.",
  "reference" : "",
  "remediation" : "Edit the /etc/ssh/sshd_config file to set the parameter as follows: \nClientAliveInterval 300 \nClientAliveCountMax 0"
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowImageCheckRuleDetailSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowImageCheckRuleDetailRequest request = new ShowImageCheckRuleDetailRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withImageType("<image_type>");
        request.withNamespace("<namespace>");
        request.withImageName("<image_name>");
        request.withImageVersion("<image_version>");
        request.withCheckName("<check_name>");
        request.withCheckType("<check_type>");
        request.withCheckRuleId("<check_rule_id>");
        request.withStandard("<standard>");
        request.withInstanceId("<instance_id>");
        try {
            ShowImageCheckRuleDetailResponse response = client.showImageCheckRuleDetail(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

```
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowImageCheckRuleDetailRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.image_type = "<image_type>"
        request.namespace = "<namespace>"
        request.image_name = "<image_name>"
        request.image_version = "<image_version>"
        request.check_name = "<check_name>"
        request.check_type = "<check_type>"
        request.check_rule_id = "<check_rule_id>"
        request.standard = "<standard>"
        request.instance_id = "<instance_id>"
        response = client.show_image_check_rule_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowImageCheckRuleDetailRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.ImageType = "<image_type>"
    namespaceRequest:= "<namespace>"
    request.Namespace = &namespaceRequest
    imageNameRequest:= "<image_name>"
    request.ImageName = &imageNameRequest
```

```
    imageVersionRequest:= "<image_version>"
    request.ImageVersion = &imageVersionRequest
    request.CheckName = "<check_name>"
    request.CheckType = "<check_type>"
    request.CheckRuleId = "<check_rule_id>"
    request.Standard = "<standard>"
    instanceIdRequest:= "<instance_id>"
    request.InstanceId = &instanceIdRequest
    response, err := client.ShowImageCheckRuleDetail(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	配置检查项检测报告

错误码

请参见**错误码**。

3.10 策略管理

3.10.1 查询策略组列表

功能介绍

查询策略组列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/policy/groups

表 3-324 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-325 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**
group_name	否	String	策略组名 最小长度：**1** 最大长度：**256**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**100000** 缺省值：**0**
limit	否	Integer	每页显示个数 最小值：**10** 最大值：**200** 缺省值：**10**
container_mode	否	Boolean	是否查询容器版策略

请求参数

表 3-326 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**

响应参数

状态码： 200

表 3-327 响应 Body 参数

参数	参数类型	描述
total_num	Integer	总数
data_list	Array of **PolicyGroupResponseInfo** objects	策略组列表 数组长度：**0 - 100**

表 3-328 PolicyGroupResponseInfo

参数	参数类型	描述
group_name	String	策略组名
group_id	String	策略组ID
description	String	策略组的描述信息 最小长度：**1** 最大长度：**64**
deletable	Boolean	是否允许删除该策略组
host_num	Integer	关联服务器数
default_group	Boolean	是否是默认策略组

参数	参数类型	描述
support_os	String	支持的操作系统，包含如下: ● Linux： 支持Linux系统 ● Windows : 支持Windows系统
support_version	String	支持的版本，包含如下: ● hss.version.basic： 基础版策略组 ● hss.version.advanced : 专业版策略组 ● hss.version.enterprise : 企业版策略组 ● hss.version.premium : 旗舰版策略组 ● hss.version.wtp : 网页防篡改版策略组 ● hss.version.container.enterprise : 容器版策略组

请求示例

查询所有企业项目下的策略组列表。

```
GET https://{endpoint}/v5/{project_id}/policy/groups?
offset=0&limit=100&enterprise_project_id=all_granted_eps
```

响应示例

状态码： 200

策略组列表

```
{
  "data_list" : [ {
    "default_group" : true,
    "deletable" : false,
    "description" : "container policy group for linux",
    "group_id" : "c831f177-226d-4b91-be0f-bcf98d04ef5d",
    "group_name" : "tenant_linux_container_default_policy_group ",
    "host_num" : 0,
    "support_version" : "hss.version.container.enterprise",
    "support_os" : "Linux"
  }, {
    "default_group" : true,
    "deletable" : false,
    "description" : "enterprise policy group for windows",
    "group_id" : "1ff54b90-1b3e-42a9-a1da-9883a83385ce",
    "group_name" : "tenant_windows_enterprise_default_policy_group ",
    "host_num" : 0,
    "support_version" : "hss.version.enterprise",
    "support_os" : "Windows"
  }, {
    "default_group" : true,
    "deletable" : false,
    "description" : "enterprise policy group for linux",
    "group_id" : "1069bcc0-c806-4ccd-a35d-f1f7456805e9",
    "group_name" : "tenant_linux_enterprise_default_policy_group ",
    "host_num" : 1,
    "support_version" : "hss.version.enterprise",
    "support_os" : "Linux"
  }, {
```

```
  "default_group" : true,
  "deletable" : false,
  "description" : "premium policy group for windows",
  "group_id" : "11216d24-9e91-4a05-9212-c4c1d646ee79",
  "group_name" : "tenant_windows_premium_default_policy_group ",
  "host_num" : 0,
  "support_version" : "hss.version.premium",
  "support_os" : "Linux"
 }, {
  "default_group" : true,
  "deletable" : false,
  "description" : "premium policy group for linux",
  "group_id" : "e6e1228a-7bb4-424f-a42b-755162234da7",
  "group_name" : "tenant_linux_premium_default_policy_group ",
  "host_num" : 0,
  "support_version" : "hss.version.premium",
  "support_os" : "Windows"
 } ],
 "total_num" : 5
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListPolicyGroupSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListPolicyGroupRequest request = new ListPolicyGroupRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withGroupName("<group_name>");
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withContainerMode(<container_mode>);
        try {
            ListPolicyGroupResponse response = client.listPolicyGroup(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListPolicyGroupRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.group_name = "<group_name>"
        request.offset = <offset>
        request.limit = <limit>
        request.container_mode = <ContainerMode>
        response = client.list_policy_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
variables and decrypted during use to ensure security.
   // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
   ak := os.Getenv("CLOUD_SDK_AK")
   sk := os.Getenv("CLOUD_SDK_SK")

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

   client := hss.NewHssClient(
      hss.HssClientBuilder().
         WithRegion(region.ValueOf("<YOUR REGION>")).
         WithCredential(auth).
         Build())

   request := &model.ListPolicyGroupRequest{}
   enterpriseProjectIdRequest:= "<enterprise_project_id>"
   request.EnterpriseProjectId = &enterpriseProjectIdRequest
   groupNameRequest:= "<group_name>"
   request.GroupName = &groupNameRequest
   offsetRequest:= int32(<offset>)
   request.Offset = &offsetRequest
   limitRequest:= int32(<limit>)
   request.Limit = &limitRequest
   containerModeRequest:= <container_mode>
   request.ContainerMode = &containerModeRequest
   response, err := client.ListPolicyGroup(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	策略组列表

错误码

请参见**错误码**。

3.10.2 部署策略组

功能介绍

部署策略组

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/policy/deploy

表 3-329 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-330 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**1** 最大长度：**256**

请求参数

表 3-331 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region ID 最小长度：**0** 最大长度：**128**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**

表 3-332 请求 Body 参数

参数	是否必选	参数类型	描述
target_policy_group_id	是	String	部署的目标策略组ID 最小长度：**36** 最大长度：**64**
operate_all	否	Boolean	是否要对全量主机部署策略，如果为true的话，不需填写host_id_list，如果为false的话，需要填写host_id_list
host_id_list	否	Array of strings	需要部署策略组的主机ID列表 最小长度：**1** 最大长度：**128** 数组长度：**0 - 10000**

响应参数

无

请求示例

部署服务器防护策略，目标服务器ID为15462c0e-32c6-4217-a869-bbd131a00ecf，目标策略ID为f671f7-2677-4705-a320-de1a62bff306。

```
POST https://{endpoint}/v5/{project_id}/policy/deploy

{
  "target_policy_group_id" : "1df671f7-2677-4705-a320-de1a62bff306",
  "host_id_list" : [ "15462c0e-32c6-4217-a869-bbd131a00ecf" ],
  "operate_all" : false
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

部署服务器防护策略，目标服务器ID为15462c0e-32c6-4217-a869-bbd131a00ecf，目标策略ID为f671f7-2677-4705-a320-de1a62bff306。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class AssociatePolicyGroupSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        AssociatePolicyGroupRequest request = new AssociatePolicyGroupRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        AssociatePolicyGroupRequestInfo body = new AssociatePolicyGroupRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("15462c0e-32c6-4217-a869-bbd131a00ecf");
        body.withHostIdList(listbodyHostIdList);
        body.withOperateAll(false);
        body.withTargetPolicyGroupId("1df671f7-2677-4705-a320-de1a62bff306");
        request.withBody(body);
        try {
            AssociatePolicyGroupResponse response = client.associatePolicyGroup(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

部署服务器防护策略，目标服务器ID为15462c0e-32c6-4217-a869-bbd131a00ecf，目标策略ID为f671f7-2677-4705-a320-de1a62bff306。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = AssociatePolicyGroupRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListbody = [
            "15462c0e-32c6-4217-a869-bbd131a00ecf"
        ]
        request.body = AssociatePolicyGroupRequestInfo(
            host_id_list=listHostIdListbody,
            operate_all=False,
            target_policy_group_id="1df671f7-2677-4705-a320-de1a62bff306"
        )
        response = client.associate_policy_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

部署服务器防护策略，目标服务器ID为15462c0e-32c6-4217-a869-bbd131a00ecf，目标策略ID为f671f7-2677-4705-a320-de1a62bff306。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AssociatePolicyGroupRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
```

```
    var listHostIdListbody = []string{
        "15462c0e-32c6-4217-a869-bbd131a00ecf",
    }
    operateAllAssociatePolicyGroupRequestInfo:= false
    request.Body = &model.AssociatePolicyGroupRequestInfo{
        HostIdList: &listHostIdListbody,
        OperateAll: &operateAllAssociatePolicyGroupRequestInfo,
        TargetPolicyGroupId: "1df671f7-2677-4705-a320-de1a62bff306",
    }
    response, err := client.AssociatePolicyGroup(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	Success
400	参数非法
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.11 漏洞管理

3.11.1 查询漏洞列表

功能介绍

查询漏洞列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/vulnerabilities

表 3-333 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-334 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**
type	否	String	漏洞类型，包含如下： - linux_vul : linux漏洞 - windows_vul : windows漏洞 - web_cms : Web-CMS漏洞 - app_vul : 应用漏洞 最小长度：**0** 最大长度：**32**
vul_id	否	String	漏洞ID 最小长度：**0** 最大长度：**256**
vul_name	否	String	漏洞名称 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页显示个数 最小值：**0** 最大值：**200** 缺省值：**10**

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
repair_priority	否	String	修复优先级 Critical 紧急 High 高 Medium 中 Low 低 最小长度：**1** 最大长度：**10**
handle_status	否	String	处置状态，包含如下: ● unhandled ： 未处理 ● handled : 已处理 缺省值：**unhandled** 最小长度：**1** 最大长度：**32**
cve_id	否	String	漏洞编号 最小长度：**0** 最大长度：**32**
label_list	否	String	漏洞标签 最小长度：**0** 最大长度：**128**
status	否	String	漏洞状态 最小长度：**0** 最大长度：**32**
asset_value	否	String	资产重要性 important common test 最小长度：**0** 最大长度：**32**
group_name	否	String	服务器组名称 最小长度：**0** 最大长度：**256**

请求参数

表 3-335 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-336 响应 Body 参数

参数	参数类型	描述
total_num	Long	漏洞总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **VulInfo** objects	软件漏洞列表 数组长度：**0 - 2147483647**

表 3-337 VulInfo

参数	参数类型	描述
vul_name	String	漏洞名称 最小长度：**0** 最大长度：**256**
vul_id	String	漏洞ID 最小长度：**0** 最大长度：**64**
label_list	Array of strings	漏洞标签 最小长度：**0** 最大长度：**65534** 数组长度：**0 - 2147483647**

参数	参数类型	描述
repair_necessity	String	修复必要性 ● Critical : 漏洞cvss评分大于等于9；对应控制台页面的高危 ● High : 漏洞cvss评分大于等于7，小于9；对应控制台页面的中危 ● Medium : 漏洞cvss评分大于等于4，小于7；对应控制台页面的中危 ● Low : 漏洞cvss评分小于4；对应控制台页面的低危 最小长度：**0** 最大长度：**64**
severity_level	String	漏洞级别 ● Critical : 漏洞cvss评分大于等于9；对应控制台页面的高危 ● High : 漏洞cvss评分大于等于7，小于9；对应控制台页面的中危 ● Medium : 漏洞cvss评分大于等于4，小于7；对应控制台页面的中危 ● Low : 漏洞cvss评分小于4；对应控制台页面的低危 最小长度：**0** 最大长度：**64**
host_num	Integer	受影响服务器台数 最小值：**0** 最大值：**2147483647**
unhandle_host_num	Integer	未处理主机台数，除已忽略和已修复的主机数量 最小值：**0** 最大值：**2147483647**
scan_time	Long	最近扫描时间，时间戳单位：毫秒 最小值：**0** 最大值：**9223372036854775807**
solution_detail	String	修复漏洞的指导意见 最小长度：**0** 最大长度：**65534**
url	String	URL链接 最小长度：**0** 最大长度：**2083**

参数	参数类型	描述
description	String	漏洞描述 最小长度：**0** 最大长度：**65534**
type	String	漏洞类型，包含如下： -linux_vul : linux漏洞 -windows_vul : windows漏洞 -web_cms : Web-CMS漏洞 -app_vul : 应用漏洞 最小长度：**0** 最大长度：**128**
host_id_list	Array of strings	可处置该漏洞的主机列表 最小长度：**0** 最大长度：**128** 数组长度：**0 - 2147483647**
cve_list	Array of **cve_list** objects	CVE列表 数组长度：**1 - 10000**
patch_url	String	补丁地址 最小长度：**0** 最大长度：**512**
repair_priority	String	修复优先级 Critical 紧急 High 高 Medium 中 Low 低 最小长度：**1** 最大长度：**32**
hosts_num	**VulnerabilityHostNumberInfo** object	影响主机
repair_success_num	Integer	修复成功次数 最小值：**0** 最大值：**1000000**
fixed_num	Long	修复数量 最小值：**0** 最大值：**1000000**
ignored_num	Long	忽略数量 最小值：**0** 最大值：**1000000**
verify_num	Integer	验证数量 最小值：**0** 最大值：**1000000**

参数	参数类型	描述
repair_priority_list	Array of RepairPriorityListInfo objects	修复优先级，每个修复优先级对应的主机数量 数组长度：**0 - 4**

表 3-338 cve_list

参数	参数类型	描述
cve_id	String	CVE ID 最小长度：**1** 最大长度：**32**
cvss	Float	CVSS分值 最小值：**0** 最大值：**10**

表 3-339 VulnerabilityHostNumberInfo

参数	参数类型	描述
important	Integer	重要主机数量 最小值：**0** 最大值：**10000**
common	Integer	一般主机数量 最小值：**0** 最大值：**10000**
test	Integer	测试主机数量 最小值：**0** 最大值：**10000**

表 3-340 RepairPriorityListInfo

参数	参数类型	描述
repair_priority	String	修复优先级 Critical 紧急 High 高 Medium 中 Low 低 最小长度：**1** 最大长度：**10**

参数	参数类型	描述
host_num	Integer	当前修复优先级对应的主机数量 最小值：**0** 最大值：**2147483647**

请求示例

查询project_id为2b31ed520xxxxxxebedb6e57xxxxxxxx的漏洞列表前10条数据。

```
GET https://{endpoint}/v5/2b31ed520xxxxxxebedb6e57xxxxxxxx/vulnerability/vulnerabilities?offset=0&limit=10
```

响应示例

状态码： 200

漏洞列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "description" : "It was discovered that FreeType did not correctly handle certain malformed font files. If a user were tricked into using a specially crafted font file, a remote attacker could cause FreeType to crash, or possibly execute arbitrary code.",
    "host_id_list" : [ "caa958ad-a481-4d46-b51e-6861b8864515" ],
    "host_num" : 1,
    "scan_time" : 1661752185836,
    "severity_level" : "Critical",
    "repair_necessity" : "Critical",
    "solution_detail" : "To upgrade the affected software",
    "type" : "linux_vul",
    "unhandle_host_num" : 0,
    "url" : "https://ubuntu.com/security/CVE-2022-27405",
    "vul_id" : "USN-5528-1",
    "vul_name" : "USN-5528-1: FreeType vulnerabilities",
    "repair_priority_list" : [ {
      "repair_priority" : "Critical",
      "host_num" : 0
    }, {
      "repair_priority" : "High",
      "host_num" : 0
    }, {
      "repair_priority" : "Medium",
      "host_num" : 1
    }, {
      "repair_priority" : "Low",
      "host_num" : 0
    } ]
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListVulnerabilitiesSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListVulnerabilitiesRequest request = new ListVulnerabilitiesRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withType("<type>");
        request.withVulId("<vul_id>");
        request.withVulName("<vul_name>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withRepairPriority("<repair_priority>");
        request.withHandleStatus("<handle_status>");
        request.withCveId("<cve_id>");
        request.withLabelList("<label_list>");
        request.withStatus("<status>");
        request.withAssetValue("<asset_value>");
        request.withGroupName("<group_name>");
        try {
            ListVulnerabilitiesResponse response = client.listVulnerabilities(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *
```

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListVulnerabilitiesRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.type = "<type>"
        request.vul_id = "<vul_id>"
        request.vul_name = "<vul_name>"
        request.limit = <limit>
        request.offset = <offset>
        request.repair_priority = "<repair_priority>"
        request.handle_status = "<handle_status>"
        request.cve_id = "<cve_id>"
        request.label_list = "<label_list>"
        request.status = "<status>"
        request.asset_value = "<asset_value>"
        request.group_name = "<group_name>"
        response = client.list_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)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
        Build())

    request := &model.ListVulnerabilitiesRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    vulIdRequest:= "<vul_id>"
    request.VulId = &vulIdRequest
    vulNameRequest:= "<vul_name>"
    request.VulName = &vulNameRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    repairPriorityRequest:= "<repair_priority>"
    request.RepairPriority = &repairPriorityRequest
    handleStatusRequest:= "<handle_status>"
    request.HandleStatus = &handleStatusRequest
    cveIdRequest:= "<cve_id>"
    request.CveId = &cveIdRequest
    labelListRequest:= "<label_list>"
    request.LabelList = &labelListRequest
    statusRequest:= "<status>"
    request.Status = &statusRequest
    assetValueRequest:= "<asset_value>"
    request.AssetValue = &assetValueRequest
    groupNameRequest:= "<group_name>"
    request.GroupName = &groupNameRequest
    response, err := client.ListVulnerabilities(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	漏洞列表

错误码

请参见**错误码**。

3.11.2 查询单个漏洞影响的云服务器信息

功能介绍

查询单个漏洞影响的云服务器信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/hosts

表 3-341 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-342 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**
vul_id	是	String	漏洞ID 最小长度：**0** 最大长度：**64**
type	是	String	漏洞类型 • linux_vul : 漏洞类型-linux漏洞 • windows_vul : 漏洞类型-windows漏洞 • web_cms : Web-CMS漏洞 • app_vul : 应用漏洞 • urgent_vul : 应急漏洞 最小长度：**0** 最大长度：**64**
host_name	否	String	受影响主机名称 最小长度：**0** 最大长度：**256**

参数	是否必选	参数类型	描述
host_ip	否	String	受影响主机ip 最小长度：**0** 最大长度：**128**
status	否	String	漏洞状态 ● vul_status_unfix : 未处理 ● vul_status_ignored : 已忽略 ● vul_status_verified : 验证中 ● vul_status_fixing : 修复中 ● vul_status_fixed : 修复成功 ● vul_status_reboot : 修复成功待重启 ● vul_status_failed : 修复失败 ● vul_status_fix_after_reboot : 请重启主机再次修复 最小长度：**0** 最大长度：**128**
limit	否	Integer	每页条数 最小值：**10** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
asset_value	否	String	资产重要性 important:重要 common：一般 test：测试 最小长度：**0** 最大长度：**32**
group_name	否	String	服务器组名称 最小长度：**0** 最大长度：**256**
handle_status	否	String	处置状态，包含如下: ● unhandled ： 未处理 ● handled : 已处理 最小长度：**1** 最大长度：**32**

参数	是否必选	参数类型	描述
severity_level	否	String	危险程度 ，Critical，High，Medium，Low 最小长度：**0** 最大长度：**32**
is_affect_business	否	Boolean	是否影响业务
repair_priority	否	String	修复优先级,包含如下 • Critical 紧急 • High 高 • Medium 中 • Low 低 最小长度：**1** 最大长度：**10**

请求参数

表 3-343 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-344 响应 Body 参数

参数	参数类型	描述
total_num	Integer	受影响的云服务器总数 最小值：**0** 最大值：**10000**

参数	参数类型	描述
data_list	Array of **VulHostInfo** objects	受影响的云服务器列表 数组长度：**1 - 10000**

表 3-345 VulHostInfo

参数	参数类型	描述
host_id	String	受漏洞影响的服务器id 最小长度：**1** 最大长度：**128**
severity_level	String	危险程度 • Critical : 漏洞cvss评分大于等于9；对应控制台页面的高危 • High : 漏洞cvss评分大于等于7，小于9；对应控制台页面的中危 • Medium : 漏洞cvss评分大于等于4，小于7；对应控制台页面的中危 • Low : 漏洞cvss评分小于4；对应控制台页面的低危 最小长度：**1** 最大长度：**128**
host_name	String	受影响主机名称 最小长度：**1** 最大长度：**256**
host_ip	String	受影响主机ip 最小长度：**1** 最大长度：**256**
agent_id	String	Agent ID 最小长度：**1** 最大长度：**128**
version	String	主机绑定的配额版本 最小长度：**1** 最大长度：**128**
cve_num	Integer	漏洞cve总数 最小值：**0** 最大值：**10000**

参数	参数类型	描述
cve_id_list	Array of strings	漏洞对应的cve id列表 最小长度：**1** 最大长度：**64** 数组长度：**1 - 10000**
status	String	漏洞状态 ● vul_status_unfix : 未处理 ● vul_status_ignored : 已忽略 ● vul_status_verified : 验证中 ● vul_status_fixing : 修复中 ● vul_status_fixed : 修复成功 ● vul_status_reboot : 修复成功待重启 ● vul_status_failed : 修复失败 ● vul_status_fix_after_reboot : 请重启主机再次修复 最小长度：**1** 最大长度：**128**
repair_cmd	String	修复漏洞需要执行的命令行（只有Linux漏洞有该字段） 最小长度：**1** 最大长度：**256**
app_path	String	应用软件的路径（只有应用漏洞有该字段） 最小长度：**1** 最大长度：**512**
region_name	String	地域 最小长度：**0** 最大长度：**128**
public_ip	String	服务器公网ip 最小长度：**0** 最大长度：**128**
private_ip	String	服务器私网ip 最小长度：**0** 最大长度：**128**
group_id	String	服务器组id 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
group_name	String	服务器组名称 最小长度：**0** 最大长度：**256**
os_type	String	操作系统 最小长度：**0** 最大长度：**32**
asset_value	String	资产重要性，包含如下3种 • important： 重要资产 • common： 一般资产 • test： 测试资产 最小长度：**0** 最大长度：**32**
is_affect_business	Boolean	是否影响业务
first_scan_time	Long	首次扫描时间 最小值：**0** 最大值：**9223372036854775807**
scan_time	Long	扫描时间，时间戳单位：毫秒 最小值：**0** 最大值：**9223372036854775807**
support_restore	Boolean	是否可以回滚到修复漏洞时创建的备份
disabled_operate_types	Array of **disabled_operate_types** objects	漏洞在当前主机上不可进行的操作类型列表 数组长度：**1 - 10000**
repair_priority	String	修复优先级,包含如下 • Critical 紧急 • High 高 • Medium 中 • Low 低 最小长度：**1** 最大长度：**10**

表 3-346 disabled_operate_types

参数	参数类型	描述
operate_type	String	操作类型 ● ignore : 忽略 ● not_ignore : 取消忽略 ● immediate_repair : 修复 ● manual_repair: 人工修复 ● verify : 验证 ● add_to_whitelist : 加入白名单 最小长度：**1** 最大长度：**64**
reason	String	不可进行操作的原因 最小长度：**0** 最大长度：**512**

请求示例

查询具有漏洞EulerOS-SA-2021-1894的服务器列表的前10条数据

```
GET https://{endpoint}/v5/2b31ed520xxxxxxebedb6e57xxxxxxxx/vulnerability/hosts?vul_id=EulerOS-SA-2021-1894&offset=0&limit=10
```

响应示例

状态码： 200

Vul host info list

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_id" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "severity_level" : "Low",
    "host_name" : "ecs",
    "host_ip" : "xxx.xxx.xxx.xxx",
    "agent_id" : "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
    "version" : "hss.version.enterprise",
    "cve_num" : 1,
    "cve_id_list" : [ "CVE-2022-1664" ],
    "status" : "vul_status_ignored",
    "repair_cmd" : "zypper update update-alternatives",
    "app_path" : "/root/apache-tomcat-8.5.15/bin/bootstrap.jar",
    "support_restore" : true,
    "disabled_operate_types" : [ {
      "operate_type" : "immediate_repair",
      "reason" : "cce机器的内核漏洞不支持自动修复"
    } ],
    "repair_priority" : "Critical"
  } ]
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListVulHostsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListVulHostsRequest request = new ListVulHostsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withVulId("<vul_id>");
        request.withType("<type>");
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withStatus("<status>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withAssetValue("<asset_value>");
        request.withGroupName("<group_name>");
        request.withHandleStatus("<handle_status>");
        request.withSeverityLevel("<severity_level>");
        request.withIsAffectBusiness(<is_affect_business>);
        request.withRepairPriority("<repair_priority>");
        try {
            ListVulHostsResponse response = client.listVulHosts(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListVulHostsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.vul_id = "<vul_id>"
        request.type = "<type>"
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.status = "<status>"
        request.limit = <limit>
        request.offset = <offset>
        request.asset_value = "<asset_value>"
        request.group_name = "<group_name>"
        request.handle_status = "<handle_status>"
        request.severity_level = "<severity_level>"
        request.is_affect_business = <IsAffectBusiness>
        request.repair_priority = "<repair_priority>"
        response = client.list_vul_hosts(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListVulHostsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    request.VulId = "<vul_id>"
    request.Type = "<type>"
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    statusRequest:= "<status>"
    request.Status = &statusRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    assetValueRequest:= "<asset_value>"
    request.AssetValue = &assetValueRequest
    groupNameRequest:= "<group_name>"
    request.GroupName = &groupNameRequest
    handleStatusRequest:= "<handle_status>"
    request.HandleStatus = &handleStatusRequest
    severityLevelRequest:= "<severity_level>"
    request.SeverityLevel = &severityLevelRequest
    isAffectBusinessRequest:= <is_affect_business>
    request.IsAffectBusiness = &isAffectBusinessRequest
    repairPriorityRequest:= "<repair_priority>"
    request.RepairPriority = &repairPriorityRequest
    response, err := client.ListVulHosts(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	Vul host info list

错误码

请参见**错误码**。

3.11.3 修改漏洞的状态

功能介绍

修改漏洞的状态

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/vulnerability/status

表 3-347 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-348 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 最小长度：**0** 最大长度：**128**

请求参数

表 3-349 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） 最小长度：**32** 最大长度：**4096**

参数	是否必选	参数类型	描述
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**

表 3-350 请求 Body 参数

参数	是否必选	参数类型	描述
operate_type	是	String	操作类型 ● ignore : 忽略 ● not_ignore : 取消忽略 ● immediate_repair : 修复 ● manual_repair: 人工修复 ● verify : 验证 ● add_to_whitelist : 加入白名单 最小长度：**1** 最大长度：**64**
remark	否	String	备注 最小长度：**0** 最大长度：**512**
select_type	否	String	选择全部漏洞类型 ● all_vul : 选择全部漏洞 ● all_host : 选择全部主机漏洞 最小长度：**1** 最大长度：**64**
type	否	String	漏洞类型，默认为linux_vul，包括如下： ● linux_vul : 漏洞类型-linux漏洞 ● windows_vul : 漏洞类型-windows漏洞 ● web_cms : Web-CMS漏洞 ● app_vul : 应用漏洞 ● urgent_vul : 应急漏洞 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
data_list	否	Array of **VulOperateInfo** objects	漏洞列表 数组长度：**1 - 500**
host_data_list	否	Array of **HostVulOperateInfo** objects	主机维度漏洞列表 数组长度：**1 - 500**
backup_info_id	否	String	本次漏洞处理的备份信息id，若不传该参数，则不进行备份 最小长度：**1** 最大长度：**128**
custom_backup_hosts	否	Array of **custom_backup_hosts** objects	自定义备份主机使用的存储库及备份名称；不在该列表中的主机备份时系统会自动选取剩余空间最大的存储库，并自动生成备份名称 数组长度：**1 - 50**

表 3-351 VulOperateInfo

参数	是否必选	参数类型	描述
vul_id	是	String	漏洞ID 最小长度：**1** 最大长度：**64**
host_id_list	是	Array of strings	主机列表 最小长度：**1** 最大长度：**64** 数组长度：**1 - 500**

表 3-352 HostVulOperateInfo

参数	是否必选	参数类型	描述
host_id	是	String	主机ID 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
vul_id_list	是	Array of strings	漏洞列表 最小长度：**1** 最大长度：**64** 数组长度：**1 - 500**

表 3-353 custom_backup_hosts

参数	是否必选	参数类型	描述
host_id	否	String	主机id 最小长度：**1** 最大长度：**128**
vault_id	否	String	存储库id 最小长度：**1** 最大长度：**128**
backup_name	否	String	备份名称 最小长度：**1** 最大长度：**64**

响应参数

无

请求示例

修改ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器的漏洞状态，将EulerOS-SA-2021-1894漏洞状态修改为忽略。

```
{
  "operate_type" : "ignore",
  "data_list" : [ {
    "vul_id" : "EulerOS-SA-2021-1894",
    "host_id_list" : [ "71a15ecc-049f-4cca-bd28-5e90aca1817f" ]
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器的漏洞状态，将EulerOS-SA-2021-1894漏洞状态修改为忽略。

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

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeVulStatusSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeVulStatusRequest request = new ChangeVulStatusRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ChangeVulStatusRequestInfo body = new ChangeVulStatusRequestInfo();
        List<String> listDataListHostIdList = new ArrayList<>();
        listDataListHostIdList.add("71a15ecc-049f-4cca-bd28-5e90aca1817f");
        List<VulOperateInfo> listbodyDataList = new ArrayList<>();
        listbodyDataList.add(
            new VulOperateInfo()
                .withVulId("EulerOS-SA-2021-1894")
                .withHostIdList(listDataListHostIdList)
        );
        body.withDataList(listbodyDataList);
        body.withOperateType("ignore");
        request.withBody(body);
        try {
            ChangeVulStatusResponse response = client.changeVulStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器的漏洞状态，将EulerOS-SA-2021-1894漏洞状态修改为忽略。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeVulStatusRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListDataList = [
            "71a15ecc-049f-4cca-bd28-5e90aca1817f"
        ]
        listDataListbody = [
            VulOperateInfo(
                vul_id="EulerOS-SA-2021-1894",
                host_id_list=listHostIdListDataList
            )
        ]
        request.body = ChangeVulStatusRequestInfo(
            data_list=listDataListbody,
            operate_type="ignore"
        )
        response = client.change_vul_status(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改ID为71a15ecc-049f-4cca-bd28-5e90aca1817f的服务器的漏洞状态，将EulerOS-SA-2021-1894漏洞状态修改为忽略。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ChangeVulStatusRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdListDataList = []string{
        "71a15ecc-049f-4cca-bd28-5e90aca1817f",
    }
    var listDataListbody = []model.VulOperateInfo{
        {
            VulId: "EulerOS-SA-2021-1894",
            HostIdList: listHostIdListDataList,
        },
    }
    request.Body = &model.ChangeVulStatusRequestInfo{
        DataList: &listDataListbody,
        OperateType: "ignore",
    }
    response, err := client.ChangeVulStatus(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	successful response

错误码

请参见**错误码**。

3.11.4 查询单台服务器漏洞信息

功能介绍

查询单台服务器漏洞信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/host/{host_id}

表 3-354 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
host_id	是	String	服务器id 最小长度：**1** 最大长度：**128**

表 3-355 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业租户ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**

参数	是否必选	参数类型	描述
type	否	String	漏洞类型，默认为linux_vul，包括如下： ● linux_vul : 漏洞类型-linux漏洞 ● windows_vul : 漏洞类型-windows漏洞 ● web_cms : Web-CMS漏洞 ● app_vul : 应用漏洞 ● urgent_vul : 应急漏洞 最小长度：**0** 最大长度：**64**
vul_name	否	String	漏洞名称 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页显示个数 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
handle_status	否	String	处置状态，包含如下: ● unhandled ： 未处理 ● handled : 已处理 最小长度：**1** 最大长度：**32**

参数	是否必选	参数类型	描述
status	否	String	漏洞状态，包含如下： ● vul_status_unfix : 未处理 ● vul_status_ignored : 已忽略 ● vul_status_verified : 验证中 ● vul_status_fixing : 修复中 ● vul_status_fixed : 修复成功 ● vul_status_reboot : 修复成功待重启 ● vul_status_failed : 修复失败 ● vul_status_fix_after_reboot : 请重启主机再次修复 最小长度：**1** 最大长度：**32**
repair_priority	否	String	修复优先级,包含如下 ● Critical 紧急 ● High 高 ● Medium 中 ● Low 低 最小长度：**1** 最大长度：**10**

请求参数

表 3-356 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-357 响应 Body 参数

参数	参数类型	描述
total_num	Long	总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **HostVulInfo** objects	服务器上的漏洞列表 数组长度：**0 - 2147483647**

表 3-358 HostVulInfo

参数	参数类型	描述
vul_name	String	漏洞名称 最小长度：**0** 最大长度：**256**
vul_id	String	漏洞ID 最小长度：**0** 最大长度：**64**
label_list	Array of strings	漏洞标签列表 最小长度：**0** 最大长度：**65534** 数组长度：**0 - 2147483647**
repair_necessity	String	修复紧急度，包括如下： • immediate_repair : 尽快修复 • delay_repair : 延后修复 • not_needed_repair : 暂可不修复 最小长度：**0** 最大长度：**64**
scan_time	Long	最近扫描时间 最小值：**0** 最大值：**9223372036854775807**
type	String	漏洞类型，包含如下： -linux_vul : linux漏洞 -windows_vul : windows漏洞 -web_cms : Web-CMS漏洞 -app_vul : 应用漏洞 最小长度：**0** 最大长度：**128**

参数	参数类型	描述
app_list	Array of **app_list** objects	服务器上受该漏洞影响的软件列表 数组长度：**0 - 2147483647**
severity_level	String	危险程度 • Critical：漏洞cvss评分大于等于9；对应控制台页面的高危 • High：漏洞cvss评分大于等于7，小于9；对应控制台页面的中危 • Medium：漏洞cvss评分大于等于4，小于7；对应控制台页面的中危 • Low：漏洞cvss评分小于4；对应控制台页面的低危 最小长度：**1** 最大长度：**128**
solution_detail	String	解决方案 最小长度：**0** 最大长度：**65534**
url	String	URL链接 最小长度：**0** 最大长度：**2083**
description	String	漏洞描述 最小长度：**0** 最大长度：**65534**
repair_cmd	String	修复命令行 最小长度：**1** 最大长度：**256**
status	String	漏洞状态 • vul_status_unfix：未处理 • vul_status_ignored：已忽略 • vul_status_verified：验证中 • vul_status_fixing：修复中 • vul_status_fixed：修复成功 • vul_status_reboot：修复成功待重启 • vul_status_failed：修复失败 • vul_status_fix_after_reboot：请重启主机再次修复 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
repair_success_num	Integer	HSS全网修复该漏洞的次数 最小值：**0** 最大值：**1000000**
cve_list	Array of **cve_list** objects	CVE列表 数组长度：**1 - 10000**
is_affect_business	Boolean	是否影响业务
first_scan_time	Long	首次扫描时间 最小值：**0** 最大值：**9223372036854775807**
app_name	String	软件名称 最小长度：**0** 最大长度：**256**
app_version	String	软件版本 最小长度：**0** 最大长度：**256**
app_path	String	软件路径 最小长度：**0** 最大长度：**512**
version	String	主机配额 最小长度：**0** 最大长度：**128**
support_restore	Boolean	是否可以回滚到修复漏洞时创建的备份
disabled_operate_types	Array of **disabled_operate_types** objects	该漏洞不可进行的操作类型列表 数组长度：**1 - 10000**
repair_priority	String	修复优先级,包含如下 • Critical 紧急 • High 高 • Medium 中 • Low 低 最小长度：**1** 最大长度：**10**

表 3-359 app_list

参数	参数类型	描述
app_name	String	软件名称 最小长度：**0** 最大长度：**256**
app_version	String	软件版本 最小长度：**0** 最大长度：**256**
upgrade_version	String	修复漏洞软件需要升级到的版本 最小长度：**0** 最大长度：**256**
app_path	String	应用软件的路径（只有应用漏洞有该字段） 最小长度：**1** 最大长度：**512**

表 3-360 cve_list

参数	参数类型	描述
cve_id	String	CVE ID 最小长度：**1** 最大长度：**32**
cvss	Float	CVSS分值 最小值：**0** 最大值：**10**

表 3-361 disabled_operate_types

参数	参数类型	描述
operate_type	String	操作类型 ● ignore : 忽略 ● not_ignore : 取消忽略 ● immediate_repair : 修复 ● manual_repair: 人工修复 ● verify : 验证 ● add_to_whitelist : 加入白名单 最小长度：**1** 最大长度：**64**
reason	String	不可进行操作的原因 最小长度：**0** 最大长度：**512**

请求示例

查询id为xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx的服务器上的漏洞列表前10条数据

```
GET https://{endpoint}/v5/2b31ed520xxxxxxebedb6e57xxxxxxxx/vulnerability/host/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx?offset=0&limit=10
```

响应示例

状态码： 200

服务器上的漏洞列表

```
{
  "data_list" : [ {
    "app_list" : [ {
      "app_name" : "Apache Log4j API(Apache Log4j API)",
      "app_version" : "2.8.2",
      "upgrade_version" : "2.8.3",
      "app_path" : "/CloudResetPwdUpdateAgent/lib/log4j-api-2.8.2.jar"
    }, {
      "app_name" : "Apache Log4j Core(Apache Log4j Core)",
      "app_version" : "2.8.2",
      "upgrade_version" : "2.8.3",
      "app_path" : "/CloudResetPwdUpdateAgent/lib/log4j-api-2.8.2.jar"
    } ],
    "app_name" : "Apache Log4j API(Apache Log4j API)",
    "app_path" : "/CloudResetPwdUpdateAgent/lib/log4j-api-2.8.2.jar",
    "app_version" : "2.8.2",
    "cve_list" : [ {
      "cve_id" : "CVE-2021-45046",
      "cvss" : 9
    } ],
    "description" : "发现在某些非默认配置中， Apache Log4j 2.15.0中针对CVE-2021-44228的修复不完整。当日志记录配置使用具有上下文查找（例如$${ctx:loginId}）或线程上下文映射模式（%X, %mdc或%MDC）使用JNDI查找模式构建恶意输入数据，从而在某些环境中导致信息泄漏和远程代码执行。Log4j 2.16.0 (Java 8)和2.12.2 (Java 7)通过删除对消息查找模式的支持并在默认情况下禁用JNDI功能来修复此问题。",
    "first_scan_time" : 1688956612533,
```

```
    "is_affect_business" : true,
    "label_list" : [ ],
    "repair_necessity" : "Critical",
    "scan_time" : 1690469489713,
    "severity_level" : "Critical",
    "repair_cmd" : "yum update tcpdump",
    "solution_detail" : "针对该漏洞的官方修复建议已发布，您可单击链接按照建议进行修复：\nhttps://
logging.apache.org/log4j/2.x/security.html\n针对该漏洞的补丁可参考：\nhttps://www.oracle.com/security-
alerts/cpujan2022.html\n针对该漏洞的非官方修复建议可参考：\nhttp://www.openwall.com/lists/oss-security/
2021/12/14/4\nhttps://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00646.html
\nhttps://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd
\nhttp://www.openwall.com/lists/oss-security/2021/12/15/3\nhttps://cert-portal.siemens.com/
productcert/pdf/ssa-661247.pdf\nhttps://www.kb.cert.org/vuls/id/930724\nhttps://cert-portal.siemens.com/
productcert/pdf/ssa-714170.pdf\nhttps://www.debian.org/security/2021/dsa-5022\nhttps://www.oracle.com/
security-alerts/alert-cve-2021-44228.html\nhttps://psirt.global.sonicwall.com/vuln-detail/
SNWLID-2021-0032\nhttp://www.openwall.com/lists/oss-security/2021/12/18/1\nhttps://cert-
portal.siemens.com/productcert/pdf/ssa-397453.pdf\nhttps://cert-portal.siemens.com/productcert/pdf/
ssa-479842.pdf\nhttps://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/
message/EOKPQGV24RRBBI4TBZUDQMM4MEH7MXCY/\nhttps://lists.fedoraproject.org/archives/list/
package-announce@lists.fedoraproject.org/message/SIG7FZULMNK2XF6FZRU4VWYDQXNMUGAJ/\n针对该
漏洞的漏洞利用/POC已曝光，可参考下方链接进行验证：\nhttps://github.com/X1pe0/Log4J-Scan-Win
\nhttps://github.com/cckuailong/Log4j_CVE-2021-45046\nhttps://github.com/BobTheShoplifter/
CVE-2021-45046-Info\nhttps://github.com/tejas-nagchandi/CVE-2021-45046\nhttps://github.com/pravin-pp/
log4j2-CVE-2021-45046\nhttps://github.com/mergebase/log4j-samples\nhttps://github.com/lukepasek/
log4jjndilookupremove\nhttps://github.com/ludy-dev/cve-2021-45046\nhttps://github.com/lijiejie/
log4j2_vul_local_scanner\nhttps://github.com/CaptanMoss/Log4Shell-Sandbox-Signature\nhttps://
github.com/taise-hub/log4j-poc",
    "status" : "vul_status_unfix",
    "type" : "app_vul",
    "url" : "[\"https://www.oracle.com/security-alerts/cpujan2022.html\"]",
    "version" : "hss.version.wtp",
    "vul_id" : "HCVD-APP-CVE-2021-45046",
    "vul_name" : "CVE-2021-45046",
    "repair_success_num" : 3,
    "support_restore" : true,
    "disabled_operate_types" : [ {
      "operate_type" : "immediate_repair",
      "reason" : "cce机器的内核漏洞不支持自动修复"
    } ]
  } ],
  "total_num" : 31
}
```


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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListHostVulsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
```

```
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListHostVulsRequest request = new ListHostVulsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withType("<type>");
        request.withVulName("<vul_name>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withHandleStatus("<handle_status>");
        request.withStatus("<status>");
        request.withRepairPriority("<repair_priority>");
        try {
            ListHostVulsResponse response = client.listHostVuls(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListHostVulsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.type = "<type>"
        request.vul_name = "<vul_name>"
```

```
        request.limit = <limit>
        request.offset = <offset>
        request.handle_status = "<handle_status>"
        request.status = "<status>"
        request.repair_priority = "<repair_priority>"
        response = client.list_host_vuls(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListHostVulsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    vulNameRequest:= "<vul_name>"
    request.VulName = &vulNameRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    handleStatusRequest:= "<handle_status>"
    request.HandleStatus = &handleStatusRequest
    statusRequest:= "<status>"
    request.Status = &statusRequest
    repairPriorityRequest:= "<repair_priority>"
    request.RepairPriority = &repairPriorityRequest
    response, err := client.ListHostVuls(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	服务器上的漏洞列表

错误码

请参见**错误码**。

3.11.5 创建漏洞扫描任务

功能介绍

创建漏洞扫描任务

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/vulnerability/scan-task

表 3-362 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**20** 最大长度：**64**

表 3-363 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	租户企业项目ID 最小长度：**0** 最大长度：**64**

请求参数

表 3-364 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） 最小长度：**32** 最大长度：**2097152**

表 3-365 请求 Body 参数

参数	是否必选	参数类型	描述
manual_scan_type	否	Array of strings	操作类型,包含如下： - linux_vul : linux漏洞 - windows_vul : windows漏洞 - web_cms : Web-CMS漏洞 - app_vul : 应用漏洞 - urgent_vul : 应急漏洞 数组长度：**1 - 200**
batch_flag	否	Boolean	是否是批量操作,为true时扫描所有支持的主机
range_type	否	String	扫描主机的范围，包含如下： - all_host : 扫描全部主机,此类型不需要填写agent_id_list - specific_host : 扫描指定主机 最小长度：**0** 最大长度：**32**
agent_id_list	否	Array of strings	主机列表 最小长度：**0** 最大长度：**32** 数组长度：**1 - 200**

参数	是否必选	参数类型	描述
urgent_vul_id_list	否	Array of strings	扫描的应急漏洞id列表，若为空则扫描所有应急漏洞 包含如下： "URGENT-CVE-2023-46604 Apache ActiveMQ远程代码执行漏洞" "URGENT-HSSVD-2020-1109 Elasticsearch 未授权访问漏洞" "URGENT-CVE-2022-26134 Atlassian Confluence OGNL 远程代码执行漏洞（CVE-2022-26134）" "URGENT-CVE-2023-22515 Atlassian Confluence Data Center and Server 权限提升漏洞 (CVE-2023-22515)" "URGENT-CVE-2023-22518 Atlassian Confluence Data Center & Server 授权机制不恰当漏洞 (CVE-2023-22518)" "URGENT-CVE-2023-28432 MinIO 信息泄露漏洞（CVE-2023-28432）" "URGENT-CVE-2023-37582 Apache RocketMQ 远程代码执行漏洞 (CVE-2023-37582)" "URGENT-CVE-2023-33246 Apache RocketMQ 远程代码执行漏洞 (CVE-2023-33246)" "URGENT-CNVD-2023-02709 禅道项目管理系统远程命令执行漏洞 (CNVD-2023-02709)" "URGENT-CVE-2022-36804 Atlassian Bitbucket Server 和 Data Center 命令注入漏洞 (CVE-2022-36804)" "URGENT-CVE-2022-22965 Spring Framework JDK >= 9 远程代码执行漏洞" "URGENT-CVE-2022-25845 fastjson <1.2.83 远程代码执行漏洞" "URGENT-CVE-2019-14439 Jackson-databind远程命令执行漏洞（CVE-2019-14439）" "URGENT-CVE-2020-13933 Apache Shiro身份验证绕过漏洞（CVE-2020-13933）" "URGENT-CVE-2020-26217 XStream < 1.4.14 远程代码执行漏洞

参数	是否必选	参数类型	描述
			（CVE-2020-26217）" "URGENT-CVE-2021-4034 Linux Polkit 权限提升漏洞预警（CVE-2021-4034）" "URGENT-CVE-2021-44228 Apache Log4j2 远程代码执行漏洞（CVE-2021-44228、CVE-2021-45046）" "URGENT-CVE-2022-0847 Dirty Pipe - Linux 内核本地提权漏洞（CVE-2022-0847）" 最小长度：**0** 最大长度：**32** 数组长度：**1 - 200**

响应参数

状态码： 200

表 3-366 响应 Body 参数

参数	参数类型	描述
task_id	String	检测任务id 最小长度：**0** 最大长度：**32**

请求示例

创建agent_id为0253edfd-30e7-439d-8f3f-17c54c997064，检测漏洞Id列表为urgent_vul_id_list的应急漏洞检测任务

```
POST https://{endpoint}/v5/{project_id}/vulnerability/scan-task?enterprise_project_id=XXX

{
  "manual_scan_type" : "urgent_vul",
  "batch_flag" : false,
  "range_type" : "specific_host",
  "agent_id_list" : [ "0253edfd-30e7-439d-8f3f-17c54c997064" ],
  "urgent_vul_id_list" : [ "URGENT-CVE-2023-46604", "URGENT-HSSVD-2020-1109", "URGENT-CVE-2022-26134", "URGENT-CVE-2023-22515", "URGENT-CVE-2023-22518", "URGENT-CVE-2023-28432", "URGENT-CVE-2023-37582", "URGENT-CVE-2023-33246", "URGENT-CNVD-2023-02709", "URGENT-CVE-2022-36804", "URGENT-CVE-2022-22965", "URGENT-CVE-2022-25845", "URGENT-CVE-2019-14439", "URGENT-CVE-2020-13933", "URGENT-CVE-2020-26217", "URGENT-CVE-2021-4034", "URGENT-CVE-2021-44228", "URGENT-CVE-2022-0847" ]
}
```

响应示例

状态码： 200

手动检测漏洞成功

```
{
  "task_id" : "d8a12cf7-6a43-4cd6-92b4-aabf1e917"
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建agent_id为0253edfd-30e7-439d-8f3f-17c54c997064，检测漏洞Id列表为urgent_vul_id_list的应急漏洞检测任务

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

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class CreateVulnerabilityScanTaskSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateVulnerabilityScanTaskRequest request = new CreateVulnerabilityScanTaskRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ManualVulScanRequestInfo body = new ManualVulScanRequestInfo();
        List<String> listbodyUrgentVulIdList = new ArrayList<>();
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-46604");
        listbodyUrgentVulIdList.add("URGENT-HSSVD-2020-1109");
        listbodyUrgentVulIdList.add("URGENT-CVE-2022-26134");
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-22515");
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-22518");
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-28432");
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-37582");
        listbodyUrgentVulIdList.add("URGENT-CVE-2023-33246");
        listbodyUrgentVulIdList.add("URGENT-CNVD-2023-02709");
        listbodyUrgentVulIdList.add("URGENT-CVE-2022-36804");
        listbodyUrgentVulIdList.add("URGENT-CVE-2022-22965");
        listbodyUrgentVulIdList.add("URGENT-CVE-2022-25845");
        listbodyUrgentVulIdList.add("URGENT-CVE-2019-14439");
        listbodyUrgentVulIdList.add("URGENT-CVE-2020-13933");
        listbodyUrgentVulIdList.add("URGENT-CVE-2020-26217");
```

```
        listbodyUrgentVulIdList.add("URGENT-CVE-2021-4034");
        listbodyUrgentVulIdList.add("URGENT-CVE-2021-44228");
        listbodyUrgentVulIdList.add("URGENT-CVE-2022-0847");
        List<String> listbodyAgentIdList = new ArrayList<>();
        listbodyAgentIdList.add("0253edfd-30e7-439d-8f3f-17c54c997064");
        body.withUrgentVulIdList(listbodyUrgentVulIdList);
        body.withAgentIdList(listbodyAgentIdList);
        body.withRangeType("specific_host");
        body.withBatchFlag(false);
        request.withBody(body);
        try {
            CreateVulnerabilityScanTaskResponse response = client.createVulnerabilityScanTask(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agent_id为0253edfd-30e7-439d-8f3f-17c54c997064，检测漏洞Id列表为urgent_vul_id_list的应急漏洞检测任务

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = CreateVulnerabilityScanTaskRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listUrgentVulIdListbody = [
            "URGENT-CVE-2023-46604",
            "URGENT-HSSVD-2020-1109",
            "URGENT-CVE-2022-26134",
            "URGENT-CVE-2023-22515",
            "URGENT-CVE-2023-22518",
            "URGENT-CVE-2023-28432",
            "URGENT-CVE-2023-37582",
            "URGENT-CVE-2023-33246",
            "URGENT-CNVD-2023-02709",
```

```
            "URGENT-CVE-2022-36804",
            "URGENT-CVE-2022-22965",
            "URGENT-CVE-2022-25845",
            "URGENT-CVE-2019-14439",
            "URGENT-CVE-2020-13933",
            "URGENT-CVE-2020-26217",
            "URGENT-CVE-2021-4034",
            "URGENT-CVE-2021-44228",
            "URGENT-CVE-2022-0847"
        ]
        listAgentIdListbody = [
            "0253edfd-30e7-439d-8f3f-17c54c997064"
        ]
        request.body = ManualVulScanRequestInfo(
            urgent_vul_id_list=listUrgentVulIdListbody,
            agent_id_list=listAgentIdListbody,
            range_type="specific_host",
            batch_flag=False
        )
        response = client.create_vulnerability_scan_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

创建agent_id为0253edfd-30e7-439d-8f3f-17c54c997064，检测漏洞Id列表为urgent_vul_id_list的应急漏洞检测任务

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateVulnerabilityScanTaskRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listUrgentVulIdListbody = []string{
        "URGENT-CVE-2023-46604",
        "URGENT-HSSVD-2020-1109",
        "URGENT-CVE-2022-26134",
```

```
        "URGENT-CVE-2023-22515",
        "URGENT-CVE-2023-22518",
        "URGENT-CVE-2023-28432",
        "URGENT-CVE-2023-37582",
        "URGENT-CVE-2023-33246",
        "URGENT-CNVD-2023-02709",
        "URGENT-CVE-2022-36804",
        "URGENT-CVE-2022-22965",
        "URGENT-CVE-2022-25845",
        "URGENT-CVE-2019-14439",
        "URGENT-CVE-2020-13933",
        "URGENT-CVE-2020-26217",
        "URGENT-CVE-2021-4034",
        "URGENT-CVE-2021-44228",
        "URGENT-CVE-2022-0847",
    }
    var listAgentIdListbody = []string{
        "0253edfd-30e7-439d-8f3f-17c54c997064",
    }
    rangeTypeManualVulScanRequestInfo:= "specific_host"
    batchFlagManualVulScanRequestInfo:= false
    request.Body = &model.ManualVulScanRequestInfo{
        UrgentVulIdList: &listUrgentVulIdListbody,
        AgentIdList: &listAgentIdListbody,
        RangeType: &rangeTypeManualVulScanRequestInfo,
        BatchFlag: &batchFlagManualVulScanRequestInfo,
    }
    response, err := client.CreateVulnerabilityScanTask(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	手动检测漏洞成功

错误码

请参见**错误码**。

3.11.6 查询漏洞扫描策略

功能介绍

查询漏洞扫描策略

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/scan-policy

表 3-367 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-368 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业租户ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**

请求参数

表 3-369 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-370 响应 Body 参数

参数	参数类型	描述
scan_period	String	扫描周期 ● one_day : 每天 ● three_day : 每三天 ● one_week : 每周 最小长度：**1** 最大长度：**32**
scan_vul_types	Array of strings	扫描的漏洞类型列表 最小长度：**0** 最大长度：**32** 数组长度：**0 - 2147483647**
scan_range_type	String	扫描主机的范围，包含如下： -all_host : 扫描全部主机 -specific_host : 扫描指定主机 最小长度：**0** 最大长度：**32**
host_ids	Array of strings	主机ID列表；当scan_range_type的值为specific_host时表示扫描的主机列表 最小长度：**1** 最大长度：**128** 数组长度：**0 - 20000**
total_host_num	Long	可进行漏洞扫描的主机总数 最小值：**0** 最大值：**20000**
status	String	扫描策略状态，包含如下： -open : 开启 -close : 关闭 最小长度：**0** 最大长度：**32**

请求示例

查询project_id为2b31ed520xxxxxxebedb6e57xxxxxxxx的漏洞扫描策略

```
GET https://{endpoint}/v5/2b31ed520xxxxxxebedb6e57xxxxxxxx/vulnerability/scan-policy
```

响应示例

状态码： 200

漏洞扫描策略

```
{
  "scan_period" : "one_day",
```

```
  "scan_vul_types" : [ "linux_vul" ],
  "scan_range_type" : "specific_host",
  "host_ids" : [ "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" ],
  "total_host_num" : 5,
  "status" : "open"
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowVulScanPolicySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowVulScanPolicyRequest request = new ShowVulScanPolicyRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        try {
            ShowVulScanPolicyResponse response = client.showVulScanPolicy(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowVulScanPolicyRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        response = client.show_vul_scan_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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowVulScanPolicyRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    response, err := client.ShowVulScanPolicy(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	漏洞扫描策略

错误码

请参见**错误码**。

3.11.7 修改漏洞扫描策略

功能介绍

修改漏洞扫描策略

调用方法

请参见**如何调用API**。

URI

PUT /v5/{project_id}/vulnerability/scan-policy

表 3-371 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-372 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业租户ID，注：修改漏洞扫描策略将影响租户账号下所有主机的漏洞扫描行为，因此开通了多企业项目的用户，该参数须填写“all_granted_eps”才能执行漏洞策略修改。 缺省值：**0** 最小长度：**0** 最大长度：**256**

请求参数

表 3-373 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） 最小长度：**1** 最大长度：**32768**

表 3-374 请求 Body 参数

参数	是否必选	参数类型	描述
scan_period	是	String	扫描周期 ● one_day : 每天 ● three_day : 每三天 ● one_week : 每周 最小长度：**1** 最大长度：**32**
scan_range_type	是	String	扫描主机的范围，包含如下： - all_host : 扫描全部主机 - specific_host : 扫描指定主机 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
host_ids	否	Array of strings	主机ID列表；当scan_range_type的值为specific_host时必填 最小长度：**1** 最大长度：**128** 数组长度：**0 - 20000**
scan_vul_types	否	Array of strings	扫描的漏洞类型列表 最小长度：**0** 最大长度：**32** 数组长度：**0 - 2147483647**
status	是	String	扫描策略状态，包含如下： - open : 开启 -close : 关闭 最小长度：**0** 最大长度：**32**

响应参数

无

请求示例

修改漏洞扫描策略，扫描周期为每天，扫描范围为指定主机，指定的主机id为xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx，策略状态为开启

```
PUT https://{endpoint}/v5/2b31ed520xxxxxebedb6e57xxxxxxxx/vulnerability/scan-policy?
enterprise_project_id=all_granted_eps

{
  "scan_period" : "one_day",
  "scan_range_type" : "specific_host",
  "host_ids" : [ "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" ],
  "status" : "open"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

修改漏洞扫描策略，扫描周期为每天，扫描范围为指定主机，指定的主机id为xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx，策略状态为开启

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class ChangeVulScanPolicySolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ChangeVulScanPolicyRequest request = new ChangeVulScanPolicyRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        ChangeVulScanPolicyRequestInfo body = new ChangeVulScanPolicyRequestInfo();
        List<String> listbodyHostIds = new ArrayList<>();
        listbodyHostIds.add("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx");
        body.withStatus("open");
        body.withHostIds(listbodyHostIds);
        body.withScanRangeType("specific_host");
        body.withScanPeriod("one_day");
        request.withBody(body);
        try {
            ChangeVulScanPolicyResponse response = client.changeVulScanPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改漏洞扫描策略，扫描周期为每天，扫描范围为指定主机，指定的主机id为xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx，策略状态为开启

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
```

```
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ChangeVulScanPolicyRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdsbody = [
            "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
        ]
        request.body = ChangeVulScanPolicyRequestInfo(
            status="open",
            host_ids=listHostIdsbody,
            scan_range_type="specific_host",
            scan_period="one_day"
        )
        response = client.change_vul_scan_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

修改漏洞扫描策略，扫描周期为每天，扫描范围为指定主机，指定的主机id为xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx，策略状态为开启

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
```

```
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

    request := &model.ChangeVulScanPolicyRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdsbody = []string{
        "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    }
    request.Body = &model.ChangeVulScanPolicyRequestInfo{
        Status: "open",
        HostIds: &listHostIdsbody,
        ScanRangeType: "specific_host",
        ScanPeriod: "one_day",
    }
    response, err := client.ChangeVulScanPolicy(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	successful response

错误码

请参见**错误码**。

3.11.8 查询漏洞扫描任务列表

功能介绍

查询漏洞扫描任务列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/scan-tasks

表 3-375 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-376 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	租户企业项目ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页显示个数 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**
scan_type	否	String	扫描任务的类型，包含如下： - manual : 手动扫描任务 - schedule : 定时扫描任务 最小长度：**0** 最大长度：**32**
task_id	否	String	扫描任务ID 最小长度：**0** 最大长度：**32**
min_start_time	否	Long	扫描任务开始时间的最小值 最小值：**0** 最大值：**9999999999999**
max_start_time	否	Long	扫描任务开始时间的最大值 最小值：**0** 最大值：**9999999999999**

请求参数

表 3-377 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-378 响应 Body 参数

参数	参数类型	描述
total_num	Long	总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **VulScanTaskInfo** objects	漏洞扫描任务列表 数组长度：**0 - 2147483647**

表 3-379 VulScanTaskInfo

参数	参数类型	描述
id	String	任务ID 最小长度：**1** 最大长度：**256**
scan_type	String	扫描任务的类型，包含如下： -manual : 手动扫描任务 -schedule : 定时扫描任务 最小长度：**0** 最大长度：**128**
start_time	Long	扫描任务开始的时间 最小值：**0** 最大值：**2147483647**

参数	参数类型	描述
end_time	Long	扫描任务结束的时间 最小值：**0** 最大值：**2147483647**
scan_vul_types	Array of strings	该任务扫描的漏洞类型列表 最小长度：**1** 最大长度：**32** 数组长度：**1 - 2147483647**
status	String	扫描任务的执行状态，包含如下： -running : 扫描中 -finished : 扫描完成 最小长度：**1** 最大长度：**32**
scanning_host_num	Integer	该任务处于扫描中状态的主机数量 最小值：**0** 最大值：**2147483647**
success_host_num	Integer	该任务已扫描成功的主机数量 最小值：**0** 最大值：**2147483647**
failed_host_num	Integer	该任务已扫描失败的主机数量 最小值：**0** 最大值：**2147483647**

请求示例

查询任务类型为手动扫描，task_id为195db604-2008-4e8b-a49e-389ab0175beb漏洞扫描任务信息，默认查询第一页10条

```
GET https://{endpoint}/v5/{project_id}/vulnerability/scan-tasks?offset=0&limit=10&enterprise_project_id=XXX
```

```
{
  "scan_type" : "manual",
  "task_id" : "195db604-2008-4e8b-a49e-389ab0175beb"
}
```

响应示例

状态码： 200

漏洞扫描任务列表

```
{
  "total_num" : 10,
  "data_list" : [ {
    "id" : "2b31ed520xxxxxxebedb6e57xxxxxxxx",
    "scan_type" : "manual",
    "start_time" : 1679042408195,
    "end_time" : 1679042408295,
```

```
  "scan_vul_types" : [ "linux_vul" ],
  "status" : "running",
  "scanning_host_num" : 1,
  "success_host_num" : 1,
  "failed_host_num" : 1
 } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询任务类型为手动扫描，task_id为195db604-2008-4e8b-a49e-389ab0175beb漏洞扫描任务信息，默认查询第一页10条

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListVulScanTaskSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListVulScanTaskRequest request = new ListVulScanTaskRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withScanType("<scan_type>");
        request.withTaskId("<task_id>");
        request.withMinStartTime(<min_start_time>L);
        request.withMaxStartTime(<max_start_time>L);
        try {
            ListVulScanTaskResponse response = client.listVulScanTask(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
```

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

Python

查询任务类型为手动扫描，task_id为195db604-2008-4e8b-a49e-389ab0175beb漏洞扫描任务信息，默认查询第一页10条

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListVulScanTaskRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.scan_type = "<scan_type>"
        request.task_id = "<task_id>"
        request.min_start_time = <min_start_time>
        request.max_start_time = <max_start_time>
        response = client.list_vul_scan_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

查询任务类型为手动扫描，task_id为195db604-2008-4e8b-a49e-389ab0175beb漏洞扫描任务信息，默认查询第一页10条

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)
```

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListVulScanTaskRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    scanTypeRequest:= "<scan_type>"
    request.ScanType = &scanTypeRequest
    taskIdRequest:= "<task_id>"
    request.TaskId = &taskIdRequest
    minStartTimeRequest:= int64(<min_start_time>)
    request.MinStartTime = &minStartTimeRequest
    maxStartTimeRequest:= int64(<max_start_time>)
    request.MaxStartTime = &maxStartTimeRequest
    response, err := client.ListVulScanTask(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	漏洞扫描任务列表

错误码

请参见**错误码**。

3.11.9 查询漏洞扫描任务对应的主机列表

功能介绍

查询漏洞扫描任务对应的主机列表

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/scan-task/{task_id}/hosts

表 3-380 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
task_id	是	String	任务ID 最小长度：**1** 最大长度：**256**

表 3-381 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业租户ID，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**
limit	否	Integer	每页显示个数 最小值：**0** 最大值：**200** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**2000000** 缺省值：**0**

参数	是否必选	参数类型	描述
scan_status	否	String	主机的扫描状态，包含如下： -scanning : 扫描中 -success : 扫描成功 -failed : 扫描失败 最小长度：**0** 最大长度：**32**

请求参数

表 3-382 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-383 响应 Body 参数

参数	参数类型	描述
total_num	Long	总数 最小值：**0** 最大值：**2147483647**
data_list	Array of **VulScanTaskHostInfo** objects	漏洞扫描任务对应的主机列表 数组长度：**0 - 2147483647**

表 3-384 VulScanTaskHostInfo

参数	参数类型	描述
host_id	String	主机ID 最小长度：**1** 最大长度：**128**

参数	参数类型	描述
host_name	String	主机名称 最小长度：**0** 最大长度：**128**
public_ip	String	弹性公网IP地址 最小长度：**0** 最大长度：**128**
private_ip	String	私有IP地址 最小长度：**0** 最大长度：**128**
asset_value	String	资产重要性，包含如下: • important： 重要资产 • common： 一般资产 • test： 测试资产 最小长度：**0** 最大长度：**128**
scan_status	String	主机的扫描状态，包含如下： -scanning : 扫描中 -success : 扫描成功 -failed : 扫描失败 最小长度：**0** 最大长度：**32**
failed_reasons	Array of **failed_reasons** objects	扫描失败的原因列表 数组长度：**0 - 2147483647**

表 3-385 failed_reasons

参数	参数类型	描述
vul_type	String	扫描失败的漏洞类型，包含如下： -linux_vul : linux漏洞 -windows_vul : windows漏洞 -web_cms : Web-CMS漏洞 -app_vul : 应用漏洞 -urgent_vul : 应急漏洞 最小长度：**1** 最大长度：**32**
failed_reason	String	扫描失败的原因 最小长度：**0** 最大长度：**128**

请求示例

查询漏洞扫描任务id为2b31ed520xxxxxxebedb6e57xxxxxxxx详情信息，展示失败的主机列表，包含失败原因，默认查询第一页10条

```
GET https://{endpoint}/v5/{project_id}/vulnerability/scan-task/{task_id}/hosts?
offset=0&limit=10&scan_status=failed&enterprise_project_id=XXX

{
  "scan_status" : "failed",
  "task_id" : "2b31ed520xxxxxxebedb6e57xxxxxxxx"
}
```

响应示例

状态码： 200

漏洞扫描任务对应的主机列表

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_id" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "host_name" : "ecs-ubuntu-abc123",
    "public_ip" : "112.10.10.3",
    "private_ip" : "192.168.10.1",
    "asset_value" : "important",
    "scan_status" : "failed",
    "failed_reasons" : [ {
      "vul_type" : "linux_vul",
      "failed_reason" : "this_is_failed_reason"
    } ]
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询漏洞扫描任务id为2b31ed520xxxxxxebedb6e57xxxxxxxx详情信息，展示失败的主机列表，包含失败原因，默认查询第一页10条

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListVulScanTaskHostSolution {

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

```
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListVulScanTaskHostRequest request = new ListVulScanTaskHostRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withScanStatus("<scan_status>");
        try {
            ListVulScanTaskHostResponse response = client.listVulScanTaskHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询漏洞扫描任务id为2b31ed520xxxxxxebedb6e57xxxxxxxx详情信息，展示失败的主机列表，包含失败原因，默认查询第一页10条

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListVulScanTaskHostRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.limit = <limit>
        request.offset = <offset>
        request.scan_status = "<scan_status>"
        response = client.list_vul_scan_task_host(request)
        print(response)
```

```
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询漏洞扫描任务id为2b31ed520xxxxxxebedb6e57xxxxxxxx详情信息，展示失败的主机列表，包含失败原因，默认查询第一页10条

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListVulScanTaskHostRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    scanStatusRequest:= "<scan_status>"
    request.ScanStatus = &scanStatusRequest
    response, err := client.ListVulScanTaskHost(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	漏洞扫描任务对应的主机列表

错误码

请参见**错误码**。

3.11.10 查询漏洞管理统计数据

功能介绍

查询漏洞管理统计数据

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/vulnerability/statistics

表 3-386 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**

表 3-387 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业租户ID，“0”表示默认企业项目，查询所有企业项目时填写：all_granted_eps 缺省值：**0** 最小长度：**0** 最大长度：**256**

请求参数

表 3-388 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） 最小长度：**1** 最大长度：**32768**

响应参数

状态码： 200

表 3-389 响应 Body 参数

参数	参数类型	描述
need_urgent_repair	Integer	需紧急修复的漏洞数 最小值：**0** 最大值：**2147483647**
unrepair	Integer	未完成修复的漏洞数 最小值：**0** 最大值：**2147483647**
existed_vul_hosts	Integer	存在漏洞的服务器数 最小值：**0** 最大值：**2147483647**
today_handle	Integer	今日处理漏洞数 最小值：**0** 最大值：**2147483647**
all_handle	Integer	累计处理漏洞数 最小值：**0** 最大值：**2147483647**
supported	Integer	已支持漏洞数 最小值：**0** 最大值：**2147483647**
vul_library_update_time	Long	漏洞库更新时间 最小值：**0** 最大值：**9223372036854775807**

请求示例

查询project_id为2b31ed520xxxxxxebedb6e57xxxxxxxx的漏洞统计数据

```
GET https://{endpoint}/v5/2b31ed520xxxxxxebedb6e57xxxxxxxx/vulnerability/statistics
```

响应示例

状态码： 200

漏洞数量统计数据

```
{
  "need_urgent_repair" : 22,
  "unrepair" : 23,
  "existed_vul_hosts" : 33,
  "today_handle" : 77,
  "all_handle" : 44,
  "supported" : 78,
  "vul_library_update_time" : 1692170925188
}
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ShowVulStaticsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowVulStaticsRequest request = new ShowVulStaticsRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        try {
            ShowVulStaticsResponse response = client.showVulStatics(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ShowVulStaticsRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        response = client.show_vul_statics(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"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowVulStaticsRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    response, err := client.ShowVulStatics(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	漏洞数量统计数据

错误码

请参见**错误码**。

3.12 网页防篡改

3.12.1 查询防护列表

功能介绍

查询防护列表：查询网页防篡改主机防护状态列表信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/webtamper/hosts

表 3-390 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**64**

表 3-391 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID 最小长度：**0** 最大长度：**64**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**256**
host_id	否	String	主机ID 最小长度：**0** 最大长度：**128**
public_ip	否	String	弹性公网IP 最小长度：**0** 最大长度：**128**
private_ip	否	String	私有IP 最小长度：**0** 最大长度：**128**
group_name	否	String	服务器组名称 最小长度：**0** 最大长度：**256**
os_type	否	String	操作系统类别（linux，windows） • linux : linux操作系统 • windows : windows操作系统 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
protect_status	否	String	防护状态 • closed : 未开启 • opened : 防护中 最小长度：**0** 最大长度：**32**
agent_status	否	String	客户端状态 • not_installed : agent未安装 • online : agent在线 • offline : agent不在线 最小长度：**0** 最大长度：**32**
limit	否	Integer	默认10 最小值：**10** 最大值：**100** 缺省值：**10**
offset	否	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**100** 缺省值：**0**

请求参数

表 3-392 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region Id 最小长度：**0** 最大长度：**32**

响应参数

状态码： 200

表 3-393 响应 Body 参数

参数	参数类型	描述
data_list	Array of **WtpProtectHostResponseInfo** objects	data list 数组长度：**0 - 200000**
total_num	Integer	total number of WTP protected servers 最小值：**0** 最大值：**65535**

表 3-394 WtpProtectHostResponseInfo

参数	参数类型	描述
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**
host_id	String	主机ID 最小长度：**0** 最大长度：**128**
public_ip	String	弹性公网IP 最小长度：**0** 最大长度：**128**
private_ip	String	私有IP 最小长度：**0** 最大长度：**128**
ipv6	String	私有IPv6地址 最小长度：**0** 最大长度：**256**
group_name	String	服务器组名称 最小长度：**0** 最大长度：**256**
os_bit	String	操作系统位数 最小长度：**0** 最大长度：**8**

参数	参数类型	描述
os_type	String	操作系统（linux，windows） 最小长度：**0** 最大长度：**32**
protect_status	String	防护状态 ● closed : 未开启 ● opened : 防护中 最小长度：**0** 最大长度：**32**
rasp_protect_status	String	动态网页防篡改状态 ● closed : 未开启 ● opened : 防护中 最小长度：**0** 最大长度：**32**
anti_tampering_times	Long	已防御篡改攻击次数 最小值：**0** 最大值：**2000000000**
detect_tampering_times	Long	已发现篡改攻击 最小值：**0** 最大值：**2000000000**
last_detect_time	Long	最近检测时间(ms) 最小值：**0** 最大值：**4070880000000**
scheduled_shutdown_status	String	定时关闭防护开关状态 ● opened : 开启 ● closed : 未开启 最小长度：**0** 最大长度：**32**
agent_status	String	Agent状态 ● not_installed : agent未安装 ● online : agent在线 ● offline : agent不在线 最小长度：**0** 最大长度：**32**

请求示例

查询防护状态为开启，企业项目ID为XX的网页防篡改主机防护状态列表信息，默认查询第一页10条

```
GET https://{endpoint}/v5/{project_id}/webtamper/hosts?
offset=XX&limit=XX&protect_status=opened&enterprise_project_id=XX

{
  "protect_status" : "opened"
}
```

响应示例

状态码： 200

OK

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_name" : "test",
    "host_id" : "000411f9-42a7-4acd-80e6-f7b9d3db895f",
    "public_ip" : "",
    "private_ip" : "192.168.0.70,fe80::f816:3eff:fed4:c4d7",
    "ipv6" : "fe80::f816:3eff:fed4:c4d7",
    "group_name" : "testGroup",
    "os_bit" : "64",
    "os_type" : "Linux",
    "protect_status" : "opened",
    "rasp_protect_status" : "opened",
    "anti_tampering_times" : 0,
    "detect_tampering_times" : 0,
    "last_detect_time" : 0,
    "agent_status" : "online"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询防护状态为开启，企业项目ID为XX的网页防篡改主机防护状态列表信息，默认查询第一页10条

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListWtpProtectHostSolution {

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

```
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListWtpProtectHostRequest request = new ListWtpProtectHostRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostName("<host_name>");
        request.withHostId("<host_id>");
        request.withPublicIp("<public_ip>");
        request.withPrivateIp("<private_ip>");
        request.withGroupName("<group_name>");
        request.withOsType("<os_type>");
        request.withProtectStatus("<protect_status>");
        request.withAgentStatus("<agent_status>");
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        try {
            ListWtpProtectHostResponse response = client.listWtpProtectHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询防护状态为开启，企业项目ID为XX的网页防篡改主机防护状态列表信息，默认查询第一页10条

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

    client = HssClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(HssRegion.value_of("<YOUR REGION>")) \
```

```
        .build()

    try:
        request = ListWtpProtectHostRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_name = "<host_name>"
        request.host_id = "<host_id>"
        request.public_ip = "<public_ip>"
        request.private_ip = "<private_ip>"
        request.group_name = "<group_name>"
        request.os_type = "<os_type>"
        request.protect_status = "<protect_status>"
        request.agent_status = "<agent_status>"
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_wtp_protect_host(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询防护状态为开启，企业项目ID为XX的网页防篡改主机防护状态列表信息，默认查询第一页10条

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListWtpProtectHostRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    publicIpRequest:= "<public_ip>"
    request.PublicIp = &publicIpRequest
    privateIpRequest:= "<private_ip>"
    request.PrivateIp = &privateIpRequest
```

```
    groupNameRequest:= "<group_name>"
    request.GroupName = &groupNameRequest
    osTypeRequest:= "<os_type>"
    request.OsType = &osTypeRequest
    protectStatusRequest:= "<protect_status>"
    request.ProtectStatus = &protectStatusRequest
    agentStatusRequest:= "<agent_status>"
    request.AgentStatus = &agentStatusRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListWtpProtectHost(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	OK

错误码

请参见**错误码**。

3.12.2 开启关闭网页防篡改防护

功能介绍

开启/关闭网页防篡改功能防护，下发/清空网页防篡改策略

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/webtamper/static/status

表 3-395 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**64**

表 3-396 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID 最小长度：**0** 最大长度：**64**

请求参数

表 3-397 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） 最小长度：**1** 最大长度：**32768**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**
region	是	String	Region Id 最小长度：**0** 最大长度：**32**

表 3-398 请求 Body 参数

参数	是否必选	参数类型	描述
status	是	Boolean	开启关闭状态，true表示enable， false表示disable

参数	是否必选	参数类型	描述
host_id_list	是	Array of strings	主机ID数组，不能为空 最小长度：**0** 最大长度：**128** 数组长度：**1 - 20000**
resource_id	否	String	资源ID 最小长度：**0** 最大长度：**64**
charging_mode	否	String	计费模式 • packet_cycle: 包周期 最小长度：**0** 最大长度：**32**

响应参数

无

请求示例

开启网页防篡改防护，目标服务器ID为a、b，包周期计费。

```
POST https://{endpoint}/v5/{project_id}/webtamper/static/status

{
  "status" : true,
  "host_id_list" : [ "a", "b" ],
  "resource_id" : "aaxxx",
  "charging_mode" : "packet_cycle"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启网页防篡改防护，目标服务器ID为a、b，包周期计费。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
```

```
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class SetWtpProtectionStatusInfoSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        SetWtpProtectionStatusInfoRequest request = new SetWtpProtectionStatusInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        SetWtpProtectionStatusRequestInfo body = new SetWtpProtectionStatusRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("a");
        listbodyHostIdList.add("b");
        body.withChargingMode("packet_cycle");
        body.withResourceId("aaxxx");
        body.withHostIdList(listbodyHostIdList);
        body.withStatus(true);
        request.withBody(body);
        try {
            SetWtpProtectionStatusInfoResponse response = client.setWtpProtectionStatusInfo(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启网页防篡改防护，目标服务器ID为a、b，包周期计费。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = SetWtpProtectionStatusInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        listHostIdListbody = [
            "a",
            "b"
        ]
        request.body = SetWtpProtectionStatusRequestInfo(
            charging_mode="packet_cycle",
            resource_id="aaxxx",
            host_id_list=listHostIdListbody,
            status=True
        )
        response = client.set_wtp_protection_status_info(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

开启网页防篡改防护，目标服务器ID为a、b，包周期计费。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.SetWtpProtectionStatusInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
```

```
    var listHostIdListbody = []string{
        "a",
        "b",
    }
    chargingModeSetWtpProtectionStatusRequestInfo:= "packet_cycle"
    resourceIdSetWtpProtectionStatusRequestInfo:= "aaxxx"
    request.Body = &model.SetWtpProtectionStatusRequestInfo{
        ChargingMode: &chargingModeSetWtpProtectionStatusRequestInfo,
        ResourceId: &resourceIdSetWtpProtectionStatusRequestInfo,
        HostIdList: listHostIdListbody,
        Status: true,
    }
    response, err := client.SetWtpProtectionStatusInfo(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	successful response

错误码

请参见**错误码**。

3.12.3 开启/关闭动态网页防篡改防护

功能介绍

开启/关闭动态网页防篡改防护，下发/清空动态网页防篡改策略

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/webtamper/rasp/status

表 3-399 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**64**

表 3-400 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID 最小长度：**0** 最大长度：**64**

请求参数

表 3-401 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） 最小长度：**1** 最大长度：**32768**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**
region	是	String	Region Id 最小长度：**0** 最大长度：**32**

表 3-402 请求 Body 参数

参数	是否必选	参数类型	描述
host_id_list	否	Array of strings	HostId list 最小长度：**0** 最大长度：**128** 数组长度：**0 - 20000**
status	否	Boolean	动态网页防篡改状态

响应参数

无

请求示例

开启动态网页防篡改防护，目标服务器为a、b。

```
POST https://{endpoint}/v5/{project_id}/webtamper/rasp/status

{
  "host_id_list" : [ "a", "b" ],
  "status" : true
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

开启动态网页防篡改防护，目标服务器为a、b。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class SetRaspSwitchSolution {

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

```
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        SetRaspSwitchRequest request = new SetRaspSwitchRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        SetRaspSwitchRequestInfo body = new SetRaspSwitchRequestInfo();
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("a");
        listbodyHostIdList.add("b");
        body.withStatus(true);
        body.withHostIdList(listbodyHostIdList);
        request.withBody(body);
        try {
            SetRaspSwitchResponse response = client.setRaspSwitch(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

开启动态网页防篡改防护，目标服务器为a、b。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = SetRaspSwitchRequest()
```

```
    request.enterprise_project_id = "<enterprise_project_id>"
    listHostIdListbody = [
        "a",
        "b"
    ]
    request.body = SetRaspSwitchRequestInfo(
        status=True,
        host_id_list=listHostIdListbody
    )
    response = client.set_rasp_switch(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

开启动态网页防篡改防护，目标服务器为a、b。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.SetRaspSwitchRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    var listHostIdListbody = []string{
        "a",
        "b",
    }
    statusSetRaspSwitchRequestInfo:= true
    request.Body = &model.SetRaspSwitchRequestInfo{
        Status: &statusSetRaspSwitchRequestInfo,
        HostIdList: &listHostIdListbody,
    }
    response, err := client.SetRaspSwitch(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	successful response

错误码

请参见**错误码**。

3.12.4 查询主机静态网页防篡改防护动态

功能介绍

查询主机静态网页防篡改防护动态：展示服务器名称、服务器ip、防护策略、检测时间、防护文件、事件描述信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/webtamper/static/protect-history

表 3-403 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**64**

表 3-404 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID 最小长度：**0** 最大长度：**64**

参数	是否必选	参数类型	描述
host_id	否	String	Host Id，为空时查所有主机 最小长度：**0** 最大长度：**128**
start_time	是	Long	起始时间(ms) 最小值：**0** 最大值：**4070880000000**
end_time	是	Long	终止时间(ms) 最小值：**0** 最大值：**4070880000000**
limit	是	Integer	limit 最小值：**0** 最大值：**100**
offset	是	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**100**
host_name	否	String	服务器名称 最小长度：**0** 最大长度：**128**
host_ip	否	String	服务器ip 最小长度：**0** 最大长度：**128**
file_path	否	String	防护文件 最小长度：**0** 最大长度：**128**
file_operation	否	String	文件操作类型 ● add: 新增 ● delete: 删除 ● modify: 修改内容 ● attribute: 修改属性 最小长度：**0** 最大长度：**128**

请求参数

表 3-405 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region Id 最小长度：**0** 最大长度：**32**

响应参数

状态码： 200

表 3-406 响应 Body 参数

参数	参数类型	描述
host_name	String	服务器名称 最小长度：**0** 最大长度：**256**
protect_status	String	防护状态 • close : 未开启 • opened : 防护中 最小长度：**0** 最大长度：**32**
total_num	Long	total number of static WTPs 最小值：**0** 最大值：**20000000**
data_list	Array of **HostProtectHistoryResponseInfo** objects	data list 数组长度：**0 - 20000**

表 3-407 HostProtectHistoryResponseInfo

参数	参数类型	描述
occr_time	Long	静态网页防篡改的检测时间(ms) 最小值：**0** 最大值：**4070880000000**
file_path	String	被篡改文件路径 最小长度：**0** 最大长度：**2000**
file_operation	String	文件操作类型 ● add: 新增 ● delete: 删除 ● modify: 修改内容 ● attribute: 修改属性 ● unknown: 未知 最小长度：**0** 最大长度：**32**
host_name	String	服务器名称 最小长度：**0** 最大长度：**64**
host_ip	String	服务器ip 最小长度：**0** 最大长度：**64**
process_id	String	进程ID 最小长度：**0** 最大长度：**8**
process_name	String	进程名称 最小长度：**0** 最大长度：**200**
process_cmd	String	进程命令行 最小长度：**0** 最大长度：**8191**

请求示例

查询主机静态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间1668563199000。

```
GET https://{endpoint}/v5/{project_id}/webtamper/static/protect-history
```

```
{
  "host_id" : "caa958ad-a481-4d46-b51e-6861b8864515",
  "start_time" : 1668563099000,
  "end_time" : 1668563199000,
  "limit" : 10,
  "offset" : 0
}
```

响应示例

状态码： 200

successful response

```
{
  "host_name" : "ecs-ubuntu",
  "protect_status" : "opened",
  "total_num" : 1,
  "data_list" : [ {
    "occr_time" : 1668156691000,
    "file_path" : "/root/test/tamper/test.xml",
    "host_name" : "hss-test",
    "host_ip" : "192.168.5.98",
    "file_operation" : "add",
    "process_id" : "18672",
    "process_name" : "program1",
    "process_cmd" : "del test.xml"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询主机静态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间1668563199000。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListHostProtectHistoryInfoSolution {

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

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

```
        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListHostProtectHistoryInfoRequest request = new ListHostProtectHistoryInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostId("<host_id>");
        request.withStartTime(<start_time>L);
        request.withEndTime(<end_time>L);
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withHostName("<host_name>");
        request.withHostIp("<host_ip>");
        request.withFilePath("<file_path>");
        request.withFileOperation("<file_operation>");
        try {
            ListHostProtectHistoryInfoResponse response = client.listHostProtectHistoryInfo(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询主机静态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间1668563199000。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListHostProtectHistoryInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_id = "<host_id>"
        request.start_time = <start_time>
```

```
        request.end_time = <end_time>
        request.limit = <limit>
        request.offset = <offset>
        request.host_name = "<host_name>"
        request.host_ip = "<host_ip>"
        request.file_path = "<file_path>"
        request.file_operation = "<file_operation>"
        response = client.list_host_protect_history_info(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询主机静态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间1668563199000。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListHostProtectHistoryInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    request.StartTime = int64(<start_time>)
    request.EndTime = int64(<end_time>)
    request.Limit = int32(<limit>)
    request.Offset = int32(<offset>)
    hostNameRequest:= "<host_name>"
    request.HostName = &hostNameRequest
    hostIpRequest:= "<host_ip>"
    request.HostIp = &hostIpRequest
    filePathRequest:= "<file_path>"
    request.FilePath = &filePathRequest
    fileOperationRequest:= "<file_operation>"
    request.FileOperation = &fileOperationRequest
```

```
    response, err := client.ListHostProtectHistoryInfo(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	successful response

错误码

请参见**错误码**。

3.12.5 查询主机动态网页防篡改防护动态

功能介绍

查询主机动态网页防篡改防护动态：包含告警级别、服务器ip、服务器名称、威胁类型、告警时间、攻击源ip、攻击源url信息

调用方法

请参见**如何调用API**。

URI

GET /v5/{project_id}/webtamper/rasp/protect-history

表 3-408 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**0** 最大长度：**64**

表 3-409 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目ID 最小长度：**0** 最大长度：**64**
host_id	否	String	Host Id，为空时查所有主机 最小长度：**0** 最大长度：**128**
start_time	是	Long	起始时间(ms) 最小值：**0** 最大值：**4070880000000**
end_time	是	Long	终止时间(ms) 最小值：**0** 最大值：**4070880000000**
limit	是	Integer	limit 最小值：**0** 最大值：**100**
offset	是	Integer	偏移量：指定返回记录的开始位置 最小值：**0** 最大值：**100**
alarm_level	否	Integer	告警级别 ● 1 : 低危 ● 2 : 中危 ● 3 : 高危 ● 4 : 严重 最小值：**0** 最大值：**100**
severity	否	String	威胁等级 ● Security : 安全 ● Low : 低危 ● Medium : 中危 ● High : 高危 ● Critical : 危急 最小长度：**0** 最大长度：**32**

参数	是否必选	参数类型	描述
protect_status	否	String	防护状态 • closed：未开启 • opened：防护中 最小长度：**0** 最大长度：**32**

请求参数

表 3-410 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） 最小长度：**1** 最大长度：**32768**
region	是	String	Region Id 最小长度：**0** 最大长度：**32**

响应参数

状态码： 200

表 3-411 响应 Body 参数

参数	参数类型	描述
total_num	Long	total number of dynamic WTPs 最小值：**0** 最大值：**200000**
data_list	Array of **HostRaspProtectHistoryResponseInfo** objects	data list 数组长度：**0 - 200000**

表 3-412 HostRaspProtectHistoryResponseInfo

参数	参数类型	描述
host_ip	String	服务器ip 最小长度：**0** 最大长度：**64**
host_name	String	服务器名称 最小长度：**0** 最大长度：**64**
alarm_time	Long	动态网页防篡改的告警时间(ms) 最小值：**0** 最大值：**4070880000000**
threat_type	String	威胁类型 最小长度：**0** 最大长度：**64**
alarm_level	Integer	告警级别 最小值：**0** 最大值：**100**
source_ip	String	攻击主机的源IP 最小长度：**0** 最大长度：**128**
attacked_url	String	攻击请求的URL 最小长度：**0** 最大长度：**2000**

请求示例

查询主机动态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间为1668563199000。

```
GET https://{endpoint}/v5/{project_id}/webtamper/rasp/protect-history

{
  "host_id" : "caa958ad-a481-4d46-b51e-6861b8864515",
  "start_time" : 1668563099000,
  "end_time" : 1668563199000,
  "limit" : 10,
  "offset" : 0
}
```

响应示例

状态码： 200

successful response

```
{
  "total_num" : 1,
  "data_list" : [ {
    "host_ip" : "192.168.5.98",
    "host_name" : "hss-test",
    "alarm_level" : 2,
    "alarm_time" : 1668394634000,
    "attacked_url" : "/vulns/001-dir-1.jsp",
    "source_ip" : "10.100.30.200",
    "threat_type" : "Path Traversal"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询主机动态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间为1668563199000。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class ListHostRaspProtectHistoryInfoSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        ListHostRaspProtectHistoryInfoRequest request = new ListHostRaspProtectHistoryInfoRequest();
        request.withEnterpriseProjectId("<enterprise_project_id>");
        request.withHostId("<host_id>");
        request.withStartTime(<start_time>L);
        request.withEndTime(<end_time>L);
        request.withLimit(<limit>);
        request.withOffset(<offset>);
        request.withAlarmLevel(<alarm_level>);
        request.withSeverity("<severity>");
        request.withProtectStatus("<protect_status>");
        try {
            ListHostRaspProtectHistoryInfoResponse response = client.listHostRaspProtectHistoryInfo(request);
            System.out.println(response.toString());
```

```
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询主机动态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间为1668563199000。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = ListHostRaspProtectHistoryInfoRequest()
        request.enterprise_project_id = "<enterprise_project_id>"
        request.host_id = "<host_id>"
        request.start_time = <start_time>
        request.end_time = <end_time>
        request.limit = <limit>
        request.offset = <offset>
        request.alarm_level = <alarm_level>
        request.severity = "<severity>"
        request.protect_status = "<protect_status>"
        response = client.list_host_rasp_protect_history_info(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询主机动态网页防篡改防护动态，目标主机ID为caa958ad-a481-4d46-b51e-6861b8864515，查询起始时间为1668563099000，查询终止时间为1668563199000。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListHostRaspProtectHistoryInfoRequest{}
    enterpriseProjectIdRequest:= "<enterprise_project_id>"
    request.EnterpriseProjectId = &enterpriseProjectIdRequest
    hostIdRequest:= "<host_id>"
    request.HostId = &hostIdRequest
    request.StartTime = int64(<start_time>)
    request.EndTime = int64(<end_time>)
    request.Limit = int32(<limit>)
    request.Offset = int32(<offset>)
    alarmLevelRequest:= int32(<alarm_level>)
    request.AlarmLevel = &alarmLevelRequest
    severityRequest:= "<severity>"
    request.Severity = &severityRequest
    protectStatusRequest:= "<protect_status>"
    request.ProtectStatus = &protectStatusRequest
    response, err := client.ListHostRaspProtectHistoryInfo(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	successful response

错误码

请参见**错误码**。

3.13 标签管理

3.13.1 批量创建标签

功能介绍

批量创建标签

调用方法

请参见**如何调用API**。

URI

POST /v5/{project_id}/{resource_type}/{resource_id}/tags/create

表 3-413 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
resource_type	是	String	由标签管理服务定义的资源类别，企业主机安全服务调用此接口时资源类别为hss 最小长度：**1** 最大长度：**64**
resource_id	是	String	由标签管理服务定义的资源id，企业主机安全服务调用此接口时资源id为配额ID 最小长度：**0** 最大长度：**128**

请求参数

表 3-414 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） 最小长度：**32** 最大长度：**512**
Content-Type	否	String	缺省值:application/json; charset=utf-8 最小长度：**0** 最大长度：**128**

表 3-415 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of **ResourceTagInfo** objects	标签对象列表 数组长度：**0 - 1024**

表 3-416 ResourceTagInfo

参数	是否必选	参数类型	描述
key	是	String	键。最大长度128个unicode字符。 key不能为空 最小长度：**1** 最大长度：**128**
value	是	String	值 最小长度：**1** 最大长度：**128**

响应参数

无

请求示例

创建标签键TESTKEY20220831190155（标签值为2）和标签键test（标签值为hss）。

```
POST https://{endpoint}/v5/05e1e8b7ba8010dd2f80c01070a8d4cd/hss/fbaa9aca-2b5f-11ee-8c64-fa163e139e02/tags/create

{
  "tags" : [ {
    "key" : "TESTKEY20220831190155",
    "value" : "2"
  }, {
    "key" : "test",
    "value" : "hss"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

创建标签键TESTKEY20220831190155（标签值为2）和标签键test（标签值为hss）。

```
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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;

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

public class BatchCreateTagsSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        BatchCreateTagsRequest request = new BatchCreateTagsRequest();
        BatchCreateTagsRequestInfo body = new BatchCreateTagsRequestInfo();
        List listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTagInfo()
                .withKey("TESTKEY20220831190155")
                .withValue("2")
        );
```

```
        listbodyTags.add(
            new ResourceTagInfo()
                .withKey("test")
                .withValue("hss")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            BatchCreateTagsResponse response = client.batchCreateTags(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

创建标签键TESTKEY20220831190155（标签值为2）和标签键test（标签值为hss）。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

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

    try:
        request = BatchCreateTagsRequest()
        listTagsbody = [
            ResourceTagInfo(
                key="TESTKEY20220831190155",
                value="2"
            ),
            ResourceTagInfo(
                key="test",
                value="hss"
            )
        ]
        request.body = BatchCreateTagsRequestInfo(
            tags=listTagsbody
        )
        response = client.batch_create_tags(request)
        print(response)
```

```
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建标签键TESTKEY20220831190155（标签值为2）和标签键test（标签值为hss）。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchCreateTagsRequest{}
    var listTagsbody = []model.ResourceTagInfo{
        {
            Key: "TESTKEY20220831190155",
            Value: "2",
        },
        {
            Key: "test",
            Value: "hss",
        },
    }
    request.Body = &model.BatchCreateTagsRequestInfo{
        Tags: listTagsbody,
    }
    response, err := client.BatchCreateTags(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	success
400	参数非法
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

3.13.2 删除资源标签

功能介绍

删除单个资源下的标签

调用方法

请参见**如何调用API**。

URI

DELETE /v5/{project_id}/{resource_type}/{resource_id}/tags/{key}

表 3-417 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID 最小长度：**1** 最大长度：**256**
resource_type	是	String	由标签管理服务定义的资源类别，企业主机安全服务调用此接口时资源类别为hss 最小长度：**1** 最大长度：**64**

参数	是否必选	参数类型	描述
resource_id	是	String	由标签管理服务定义的资源id，企业主机安全服务调用此接口时资源id为配额ID 最小长度：**0** 最大长度：**128**
key	是	String	待删除的key 最小长度：**1** 最大长度：**256**

请求参数

表 3-418 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） 最小长度：**32** 最大长度：**512**

响应参数

无

请求示例

删除project_id为94b5266c14ce489fa6549817f032dc61、resource_type为hss、resource_id为2acc46ee-34c2-40c2-8060-dc652e6c672a的key为abc的标签

```
DELETE https://{endpoint}/v5/94b5266c14ce489fa6549817f032dc61/hss/2acc46ee-34c2-40c2-8060-dc652e6c672a/tags/abc
```

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.hss.v5.region.HssRegion;
import com.huaweicloud.sdk.hss.v5.*;
import com.huaweicloud.sdk.hss.v5.model.*;


public class DeleteResourceInstanceTagSolution {

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

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

        HssClient client = HssClient.newBuilder()
                .withCredential(auth)
                .withRegion(HssRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteResourceInstanceTagRequest request = new DeleteResourceInstanceTagRequest();
        try {
            DeleteResourceInstanceTagResponse response = client.deleteResourceInstanceTag(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 huaweicloudsdkhss.v5.region.hss_region import HssRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkhss.v5 import *

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

    credentials = BasicCredentials(ak, sk)

    client = HssClient.new_builder() \
```

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

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

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    hss "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/hss/v5/region"
)

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

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

    client := hss.NewHssClient(
        hss.HssClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteResourceInstanceTagRequest{}
    response, err := client.DeleteResourceInstanceTag(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	Success

状态码	描述
400	参数非法
401	鉴权失败
403	权限不足
404	资源未找到
500	系统异常

错误码

请参见**错误码**。

A 附录

A.1 状态码

状态码	编码	状态说明
200	OK	请求已成功。
400	Bad Request	请求参数有误。
500	Internal Server Error	服务内部错误。

A.2 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见**API网关错误码**进行处理。

状态码	错误码	错误信息	描述	处理措施
400	HSS.0001	invalid param error	参数不合法	请检查参数是否合法
500	HSS.0041	Query host extend info error	查询信息出错	请检查参数是否合法