

Web 应用防火墙

API 参考

文档版本 41
发布日期 2025-01-20



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

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

商标声明



HUAWEI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。

目录

1 使用前必读	1
2 API 概览	3
3 如何调用 API	4
3.1 构造请求	4
3.2 认证鉴权	6
3.3 返回结果	8
4 API	10
4.1 云模式防护网站管理	10
4.1.1 查询云模式防护域名列表	10
4.1.2 创建云模式防护域名	19
4.1.3 根据防护域名 Id 查询云模式防护域名详细信息	32
4.1.4 更新云模式防护域名的配置	45
4.1.5 删除云模式防护域名	67
4.1.6 修改域名防护状态	72
4.1.7 获取云模式域名路由信息	79
4.2 独享模式防护网站管理	85
4.2.1 查询独享模式域名列表	85
4.2.2 创建独享模式域名	93
4.2.3 修改独享模式域名配置	108
4.2.4 查看独享模式域名配置	128
4.2.5 删除独享模式域名	140
4.2.6 修改独享模式域名防护状态	147
4.3 防护策略管理	153
4.3.1 查询防护策略列表	153
4.3.2 创建防护策略	162
4.3.3 根据 Id 查询防护策略	172
4.3.4 更新防护策略	180
4.3.5 删除防护策略	195
4.3.6 更新防护策略的域名	204
4.4 策略规则管理	212
4.4.1 修改单条规则的状态	212
4.4.2 查询 cc 规则列表	219

4.4.3 创建 cc 规则.....	230
4.4.4 根据 Id 查询 cc 防护规则.....	247
4.4.5 更新 cc 防护规则.....	258
4.4.6 删除 cc 防护规则.....	275
4.4.7 查询精准防护规则列表.....	286
4.4.8 创建精准防护规则.....	294
4.4.9 根据 Id 查询精准防护规则.....	306
4.4.10 更新精准防护规则.....	314
4.4.11 删除精准防护规则.....	326
4.4.12 创建全局白名单(原误报屏蔽)规则.....	334
4.4.13 查询全局白名单(原误报屏蔽)防护规则.....	346
4.4.14 更新全局白名单(原误报屏蔽)防护规则.....	354
4.4.15 删除全局白名单(原误报屏蔽)防护规则.....	366
4.4.16 查询黑白名单规则列表.....	374
4.4.17 创建黑白名单规则.....	381
4.4.18 查询黑白名单防护规则.....	390
4.4.19 更新黑白名单防护规则.....	396
4.4.20 查询全局白名单(原误报屏蔽)规则列表.....	404
4.4.21 删除黑白名单防护规则.....	414
4.4.22 查询 JS 脚本反爬虫规则列表.....	420
4.4.23 更新 JS 脚本反爬虫规则防护模式.....	428
4.4.24 创建 JS 脚本反爬虫规则.....	434
4.4.25 查询 JS 脚本反爬虫防护规则.....	443
4.4.26 更新 JS 脚本反爬虫防护规则.....	450
4.4.27 删除 JS 脚本反爬虫防护规则.....	459
4.4.28 查询隐私屏蔽防护规则列表.....	466
4.4.29 创建隐私屏蔽防护规则.....	472
4.4.30 查询隐私屏蔽防护规则.....	480
4.4.31 更新隐私屏蔽防护规则.....	487
4.4.32 删除隐私屏蔽防护规则.....	495
4.4.33 查询攻击惩罚规则列表.....	501
4.4.34 创建攻击惩罚规则.....	507
4.4.35 根据 Id 查询攻击惩罚防护规则.....	514
4.4.36 更新攻击惩罚规则.....	520
4.4.37 删除攻击惩罚规则.....	528
4.4.38 查询地理位置访问控制规则列表.....	534
4.4.39 创建地理位置控制规则.....	541
4.4.40 查询地理位置控制防护规则.....	549
4.4.41 更新地理位置控制防护规则.....	556
4.4.42 删除地理位置控制防护规则.....	565
4.4.43 查询防篡改规则列表.....	571
4.4.44 创建防篡改规则.....	578

4.4.45 查询防篡改防护规则.....	585
4.4.46 删除防篡改防护规则.....	591
4.4.47 网页防篡改规则更新缓存.....	597
4.4.48 查询防敏感信息泄露规则列表.....	603
4.4.49 创建防敏感信息泄露规则.....	610
4.4.50 查询防敏感信息泄露防护规则.....	617
4.4.51 更新防敏感信息泄露防护规则.....	623
4.4.52 删除防敏感信息泄露防护规则.....	631
4.4.53 查询引用表列表.....	637
4.4.54 创建引用表.....	644
4.4.55 查询引用表.....	651
4.4.56 修改引用表.....	657
4.4.57 删除引用表.....	664
4.5 地址组管理.....	669
4.5.1 查询地址组列表.....	670
4.5.2 创建 ip 地址组.....	677
4.5.3 查询 ip 地址组明细.....	683
4.5.4 修改 ip 地址组.....	690
4.5.5 删除 ip 地址组.....	697
4.6 证书管理.....	703
4.6.1 查询证书列表.....	703
4.6.2 创建证书.....	710
4.6.3 查询证书.....	720
4.6.4 修改证书.....	726
4.6.5 删除证书.....	733
4.6.6 绑定证书到域名.....	739
4.7 防护事件管理.....	746
4.7.1 查询指定事件 id 的防护事件详情.....	746
4.7.2 查询攻击事件列表.....	753
4.8 安全总览.....	763
4.8.1 查询安全统计 qps 次数.....	763
4.8.2 查询安全总览请求与攻击数量.....	770
4.8.3 查询安全统计带宽数据.....	777
4.8.4 查询业务异常数量.....	784
4.8.5 查询安全总览分类统计 top 信息.....	791
4.8.6 查询安全总览中请求次数时间线统计数据.....	799
4.9 局点支持特性查询.....	806
4.9.1 局点支持特性查询.....	807
4.10 独享实例管理.....	814
4.10.1 查询 WAF 独享引擎列表.....	814
4.10.2 创建 WAF 独享引擎实例.....	822
4.10.3 查询 WAF 独享引擎信息.....	831

4.10.4 重命名 WAF 独享引擎.....	838
4.10.5 删除 WAF 独享引擎信息.....	845
4.11 日志配置管理.....	852
4.11.1 配置全量日志 lts.....	852
4.11.2 查询 lts 配置信息.....	860
4.12 租户订购管理.....	866
4.12.1 购买包周期云模式 waf.....	866
4.12.2 变更包周期云模式 waf 规格.....	874
4.12.3 查询租户订购信息.....	881
4.13 租户域名查询.....	888
4.13.1 查询全部防护域名列表.....	889
4.13.2 根据 Id 查询防护域名.....	899
4.14 租户防护域名管理.....	908
4.14.1 按企业项目迁移防护域名.....	908
4.15 系统管理.....	915
4.15.1 查询 WAF 回源 IP 信息.....	915
4.16 告警管理.....	920
4.16.1 查询告警通知配置.....	920
4.16.2 更新告警通知配置.....	927
A 附录.....	936
A.1 状态码.....	936
A.2 错误码.....	937
A.3 获取项目 ID.....	947
A.4 地理位置编码.....	948

1 使用前必读

概述

Web应用防火墙（Web Application Firewall，WAF），通过对HTTP(S)请求进行检测，识别并阻断SQL注入、跨站脚本攻击、网页木马上传、命令/代码注入、文件包含、敏感文件访问、第三方应用漏洞攻击、CC攻击、恶意爬虫扫描、跨站请求伪造等攻击，保护Web服务安全稳定。

您可以使用本文档提供的API对WAF进行相关操作，如查询、更新等操作。

在调用WAF API之前，请确保已经充分了解WAF，有关WAF服务的详细介绍，请参见[Web应用防火墙](#)。

调用说明

WAF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](#)。

终端节点

终端节点（Endpoint）即调用API的[请求地址](#)，不同服务不同区域的终端节点不同，您可以从[地区和终端节点](#)中查询服务的终端节点。

基本概念

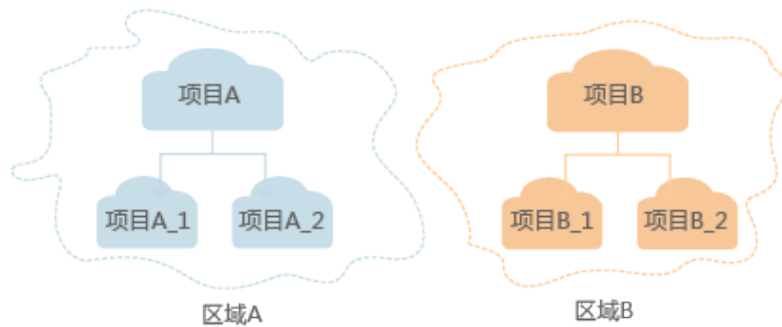
- **账号**
用户注册时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用他们进行日常管理工作。
- **用户**
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
在[我的凭证](#)下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- **区域（Region）**
从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属

Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。

详情请参见[区域和可用区](#)。

- 可用区（AZ，Availability Zone）
一个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



2 API 概览

通过使用WAF提供的接口，您可以完整地使用WAF的所有功能。

类型	说明
云模式防护网站API接口	云模式域名接口，包括创建、修改、查询以及删除域名等接口。
独享模式防护网站API接口	独享模式域名接口，包括创建、修改、查询以及删除域名等接口。
防护策略API接口	防护策略接口，包括创建防护策略、修改防护策略绑定的域名等接口。
策略规则API接口	防护规则接口，包括创建、更新、查询以及删除防护规则等接口。
证书API接口	包括创建、修改、查询证书等接口。
防护事件API接口	查询防护事件详细信息接口。
业务安全总览API接口	查询安全总览请求相关的接口。
局点支持特性查询API接口	局点支持特性查询的接口。
租户域名查询API接口	查看防护域名信息接口。

3 如何调用 API

3.1 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。

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

请求 URI

请求URI由如下部分组成。

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

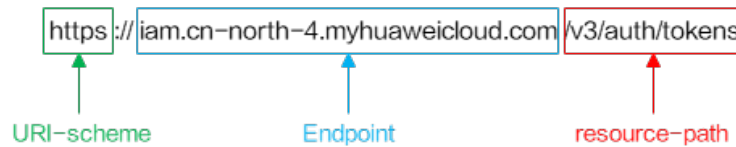
尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

- **URI-scheme:**
表示用于传输请求的协议，当前所有API均采用HTTPS协议。
- **Endpoint:**
指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从[地区和终端节点](#)获取。
例如IAM服务在“华北-北京四”区域的Endpoint为“iam.cn-north-4.myhuaweicloud.com”。
- **resource-path:**
资源路径，也即API访问路径。从具体API的URI模块获取，例如“获取用户Token”API的resource-path为“/v3/auth/tokens”。
- **query-string:**
查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“?”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。

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

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

图 3-1 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.cn-north-4.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.cn-north-4.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，***xxxxxxxxxxxxxxxxxxxx***为project的名称，如“cn-north-4”，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.cn-north-4.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

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

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。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- 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": "xxxxxxx"
    }
  }
}
```

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。

```
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

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

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。

3.3 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3-2](#)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 3-2 获取用户 Token 响应消息头

```
connection → keep-alive

content-type → application/json

date → Tue, 12 Feb 2019 06:52:13 GMT

server → Web Server

strict-transport-security → max-age=31536000; includeSubdomains;

transfer-encoding → chunked

via → proxy A

x-content-type-options → nosniff

x-download-options → noopen

x-frame-options → SAMEORIGIN

x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5

x-subject-token
→ MIIYXQYJKoZIhvcNAQcCoIIYtJCCEoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgg hacBIIIWmHsidG9rZW4iOnsiZlhwZXJlc19hdCI6ijlWMtktMDltMTNUMC
fj3Kjs6YgKnpVNRbW2eZ5eb785Z0kqJACgkIQ1wi4JlGzrpd18LGXK5tdf4q4qHCYb8P4NaY0NYejcAgz/VefFYtLWT1GSO0zxKZmlQHQj82HBqHdglZO9fuEbl5dMhdavj+33wEI
xHRCe9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXl1jipPEGA270g1FruooL6jqglFKNPQuFSOUs+uSstVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUbpuGv-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection → 1; mode=block;
```

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": "xxxxxxx",
            .....

```

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

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

其中，error_code表示错误码，error_msg表示错误描述信息。

4 API

4.1 云模式防护网站管理

4.1.1 查询云模式防护域名列表

功能介绍

查询云模式防护域名列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/instance

表 4-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-2 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

参数	是否必选	参数类型	描述
page	否	Integer	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。如果需要一次查全部域名，该参数值填-1。
hostname	否	String	要查询的防护域名，用于查询指定防护域名信息；可不传，查询用户云模式下所有防护域名
policyname	否	String	防护策略名，用于查询指定防护策略下的域名，可不传

请求参数

表 4-3 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-4 响应 Body 参数

参数	参数类型	描述
total	Integer	云模式防护域名的数量
items	Array of CloudWafHostItem objects	详细的云模式防护域名列表信息

表 4-5 CloudWafHostItem

参数	参数类型	描述
id	String	域名id
hostid	String	域名id
region	String	区域ID，控制台创建的域名会携带此参数，api调用创建的域名此参数为空，可以通过地区和终端节点文档查询区域ID对应的中文名称
description	String	域名描述信息，可选参数。
server	Array of CloudWafServer objects	防护域名的源站服务器配置信息
type	Integer	WAF部署模式，默认是1，目前仅支持反代模式，冗余参数
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">• false：不使用代理• true：使用代理
hostname	String	创建的云模式防护域名
access_code	String	cname前缀，cname后缀为.vip1.huaweicloudwaf.com
policyid	String	防护策略id
timestamp	Long	创建防护域名的时间
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">• -1: bypass，该域名的请求直接到达其后端服务器，不再经过WAF• 0: 暂停防护，WAF只转发该域名的请求，不做攻击检测• 1: 开启防护，WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态，0表示未接入，1表示已接入
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip，该参数为预留参数，用于后续功能扩展，用户可忽略 <ul style="list-style-type: none">• true：使用独享ip• false：不使用独享ip
paid_type	String	套餐付费模式，默认值为prePaid。 prePaid：包周期款模式；postPaid：按需模式。

参数	参数类型	描述
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
flag	Flag object	特殊标识，用于前端使用
enterprise_project_id	String	企业项目id

表 4-6 CloudWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

表 4-7 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">true: 开启false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">true: 开启false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">true: 支持false: 不支持

参数	参数类型	描述
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

状态码: 400

表 4-8 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-9 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-10 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-11 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-12 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-13 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

获取项目id为project_id的项目中的所有云模式防护域名

```
GET https://{Endpoint}/v1/{project_id}/waf/instance?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "items": [ {
    "id": "d0a4bc2f74e3407388a50243af700305",
    "hostid": "d0a4bc2f74e3407388a50243af700305",
    "description": "e",
    "type": 1,
    "proxy": false,
    "flag": {
      "pci_3ds": "false",
      "pci_dss": "false",
      "ipv6": "false",
      "cname": "new",
      "is_dual_az": "true"
    },
    "region": "cn-north-4",
    "hostname": "www.demo.com",
    "access_code": "7d06456ffaexxxxxxxxxx281bc13b",
    "policyid": "bb2124fabe6f42ff9fe4770eecb2670",
    "timestamp": 1642648030687,
    "protect_status": 1,
    "access_status": 0,
    "exclusive_ip": false,
    "web_tag": "iii",
    "paid_type": "prePaid"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListHostSolution {

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

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListHostRequest request = new ListHostRequest();
try {
    ListHostResponse response = client.listHost(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListHostRequest()
        response = client.list_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.2 创建云模式防护域名

功能介绍

创建云模式防护域名

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/instance

表 4-14 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-15 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-16 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)。
Content-Type	是	String	内容类型

表 4-17 请求 Body 参数

参数	是否必选	参数类型	描述
hostname	是	String	域名（域名只能由字母、数字、-、_和.组成，长度不能超过64个字符，如www.domain.com）
policyid	否	String	防护域名初始绑定的策略ID,可以通过策略名称调用查询防护策略列表（ListPolicy）接口查询到对应的策略id
server	是	Array of CloudWafServer objects	防护域名的源站服务器配置信息
certificateid	否	String	证书id，通过查询证书列表接口（ListCertificates）接口获取证书id <ul style="list-style-type: none">对外协议为HTTP时不需要填写对外协议HTTPS时为必填参数
certificatename	否	String	证书名 <ul style="list-style-type: none">对外协议为HTTP时不需要填写对外协议HTTPS时为必填参数
web_tag	否	String	网站名称，对应WAF控制台域名详情中的网站名称
exclusive_ip	否	Boolean	是否使用用户独享的引擎公网ip，该参数为预留参数，用于后续功能扩展，用户可忽略 <ul style="list-style-type: none">true：使用独享ipfalse：不使用独享ip
paid_type	否	String	套餐付费模式，默认值为prePaid。prePaid：包周期款模式；postPaid：按需模式。
proxy	是	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">false：不使用代理true：使用代理

参数	是否必选	参数类型	描述
lb_algorithm	否	String	<p>LB负载均衡，仅专业版（原企业版）和铂金版（原旗舰版）支持配置负载均衡算法</p> <ul style="list-style-type: none"> 源IP Hash：将某个IP的请求定向到同一个服务器 加权轮询：所有请求将按权重轮流分配给源站服务器 Session Hash：将某个Session标识的请求定向到同一个源站服务器，请确保在域名添加完毕后配置攻击惩罚的流量标识，否则Session Hash配置不生效
description	否	String	域名描述
forward_header_map	否	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> \$time_local \$request_id \$connection_requests \$tenant_id \$project_id \$remote_addr \$remote_port \$scheme \$request_method \$http_host -\$origin_uri \$request_length \$ssl_server_name \$ssl_protocol \$ssl_curves \$ssl_session_reused

表 4-18 CloudWafServer

参数	是否必选	参数类型	描述
front_protocol	是	String	客户端请求访问防护域名源站服务器的协议
back_protocol	是	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	否	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	是	String	客户端访问的源站服务器的IP地址
port	是	Integer	WAF转发客户端请求到源站服务的业务端口
type	是	String	源站地址为ipv4或ipv6

响应参数

状态码： 200

表 4-19 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	创建的云模式防护域名
policyid	String	策略id
access_code	String	cname前缀，cname后缀 为.vip1.huaweicloudwaf.com
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">-1: bypass，该域名的请求直接到达其后端服务器，不再经过WAF0: 暂停防护，WAF只转发该域名的请求，不做攻击检测1: 开启防护，WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态，0表示未接入，1表示已接入

参数	参数类型	描述
lb_algorithm	String	LB负载均衡，仅专业版（原企业版）和铂金版（原旗舰版）支持配置负载均衡算法 <ul style="list-style-type: none">源IP Hash：将某个IP的请求定向到同一个服务器加权轮询：所有请求将按权重轮流分配给源站服务器Session Hash：将某个Session标识的请求定向到同一个源站服务器，请确保在域名添加完毕后配置攻击惩罚的流量标识，否则Session Hash配置不生效
protocol	String	返回的客户端协议类型
certificateid	String	返回的证书id
certificatename	String	证书名称
server	Array of CloudWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">false：不使用代理true：使用代理
timestamp	Long	创建防护域名的时间
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip，该参数为预留参数，用于后续功能扩展，用户可忽略 <ul style="list-style-type: none">true：使用独享ipfalse：不使用独享ip
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
http2_enable	Boolean	是否支持http2 <ul style="list-style-type: none">true：表示支持http2false：表示不支持http2
block_page	BlockPage object	告警页面配置
flag	Flag object	特殊标识，用于前端使用
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。

参数	参数类型	描述
forward_header_map	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插入到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-20 CloudWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

表 4-21 BlockPage

参数	参数类型	描述
template	String	模板名称，默认页面请输入default，自定义告警页面时请输入custom，重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向”页面URL

表 4-22 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择text/html、text/xml和application/json三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-23 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

状态码： 400**表 4-24** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-25 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-26** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-27 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-28 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-29 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目下, 创建域名是“www.demo.com”的云模式防护域名, 该域名的客户端请求访问防护域名源站服务器的协议是“HTTPS”, WAF转发客户端请求到防护域名源站服务器的协议是“HTTP”, 源站地址是“ipv4”, 源站服务器的IP地址是“x.x.x.x”, WAF转发客户端请求到源站服务的业务端口是“7443”, 该域名不使用代理和独享ip, 套餐付费模式是包周期款模式, 域名描述是“dome”, 对应WAF控制台域名详情中的网站名称是“dome”, 对外协议的证书名称是“test6”, 证书id是“3ac1402300374a63a05be68c641e92c8”。

```
POST https://{Endpoint}/v1/{project_id}/waf/instance?enterprise_project_id=0
```

```
{
  "hostname": "www.demo.com",
  "server": [{
    "front_protocol": "HTTPS",
```

```
"back_protocol" : "HTTP",
"type" : "ipv4",
"address" : "x.x.x.x",
"port" : "7443"
}],
"proxy" : false,
"paid_type" : "prePaid",
"description" : "demo",
"web_tag" : "demo",
"certificateid" : "3ac1402300374a63a05be68c641e92c8",
"certificatename" : "test6",
"exclusive_ip" : false
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "31af669f567246c289771694f2112289",
  "hostname" : "www.demo.com",
  "protocol" : "HTTP",
  "server" : [ {
    "address" : "x.x.x.x",
    "port" : 80,
    "type" : "ipv4",
    "weight" : 1,
    "front_protocol" : "HTTP",
    "back_protocol" : "HTTP"
  } ],
  "proxy" : false,
  "timestamp" : 1650527546420,
  "flag" : {
    "pci_3ds" : "false",
    "pci_dss" : "false",
    "ipv6" : "false",
    "cname" : "new",
    "is_dual_az" : "true"
  },
  "policyid" : "41cba8aee2e94bcdbf57460874205494",
  "protect_status" : 1,
  "access_status" : 0,
  "access_code" : "1b18879b9d064f8bbcbf8abce7294cac",
  "block_page" : {
    "template" : "default"
  },
  "web_tag" : "",
  "exclusive_ip" : false,
  "http2_enable" : false
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目下，创建域名是“www.demo.com”的云模式防护域名，该域名的客户端请求访问防护域名源站服务器的协议是“HTTPS”，WAF转发客户端请求到防护域名源站服务器的协议是“HTTP”，源站地址是“ipv4”，源站服务器的IP地址是“x.x.x.x”，WAF转发客户端请求到源站服务的业务端口是“7443”，该域名不使用代理和独享ip，套餐付费模式是包周期款模式，域名描述是“dome”，对应WAF

控制台域名详情中的网站名称是“dome”，对外协议的证书名称是“test6”，证书id是“3ac1402300374a63a05be68c641e92c8”。

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

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

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

public class CreateHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateHostRequest request = new CreateHostRequest();
        CreateHostRequestBody body = new CreateHostRequestBody();
        List<CloudWafServer> listbodyServer = new ArrayList<>();
        listbodyServer.add(
            new CloudWafServer()
                .withFrontProtocol(CloudWafServer.FrontProtocolEnum.fromValue("HTTPS"))
                .withBackProtocol(CloudWafServer.BackProtocolEnum.fromValue("HTTP"))
                .withAddress("x.x.x.x")
                .withPort(7443)
                .withType(CloudWafServer.TypeEnum.fromValue("ipv4"))
        );
        body.withDescription("demo");
        body.withProxy(false);
        body.withPaidType("prePaid");
        body.withExclusiveIp(false);
        body.withWebTag("demo");
        body.withCertificatename("test6");
        body.withCertificateid("3ac1402300374a63a05be68c641e92c8");
        body.withServer(listbodyServer);
        body.withHostname("www.demo.com");
        request.withBody(body);
        try {
            CreateHostResponse response = client.createHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
        }
    }
}
```

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

Python

在项目id为project_id的项目下，创建域名是“www.demo.com”的云模式防护域名，该域名的客户端请求访问防护域名源站服务器的协议是“HTTPS”，WAF转发客户端请求到防护域名源站服务器的协议是“HTTP”，源站地址是“ipv4”，源站服务器的IP地址是“x.x.x.x”，WAF转发客户端请求到源站服务的业务端口是“7443”，该域名不使用代理和独享ip，套餐付费模式是包周期款模式，域名描述是“dome”，对应WAF控制台域名详情中的网站名称是“dome”，对外协议的证书名称是“test6”，证书id是“3ac1402300374a63a05be68c641e92c8”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateHostRequest()
        listServerbody = [
            CloudWafServer(
                front_protocol="HTTPS",
                back_protocol="HTTP",
                address="x.x.x.x",
                port=7443,
                type="ipv4"
            )
        ]
        request.body = CreateHostRequestBody(
            description="demo",
            proxy=False,
            paid_type="prePaid",
            exclusive_ip=False,
            web_tag="demo",
            certificatename="test6",
            certificateid="3ac1402300374a63a05be68c641e92c8",
            server=listServerbody,
            hostname="www.demo.com"
        )
        response = client.create_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

在项目id为project_id的项目下，创建域名是“www.demo.com”的云模式防护域名，该域名的客户端请求访问防护域名源站服务器的协议是“HTTPS”，WAF转发客户端请求到防护域名源站服务器的协议是“HTTP”，源站地址是“ipv4”，源站服务器的IP地址是“x.x.x.x”，WAF转发客户端请求到源站服务的业务端口是“7443”，该域名不使用代理和独享ip，套餐付费模式是包周期款模式，域名描述是“dome”，对应WAF控制台域名详情中的网站名称是“dome”，对外协议的证书名称是“test6”，证书id是“3ac1402300374a63a05be68c641e92c8”。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateHostRequest{}
    var listServerbody = []model.CloudWafServer{
        {
            FrontProtocol: model.GetCloudWafServerFrontProtocolEnum().HTTPS,
            BackProtocol: model.GetCloudWafServerBackProtocolEnum().HTTP,
            Address: "x.x.x.x",
            Port: int32(7443),
            Type: model.GetCloudWafServerTypeEnum().IPV4,
        },
    }
    descriptionCreateHostRequestBody:= "demo"
    paidTypeCreateHostRequestBody:= "prePaid"
    exclusivelpCreateHostRequestBody:= false
    webTagCreateHostRequestBody:= "demo"
    certificatenameCreateHostRequestBody:= "test6"
    certificateidCreateHostRequestBody:= "3ac1402300374a63a05be68c641e92c8"
    request.Body = &model.CreateHostRequestBody{
        Description: &descriptionCreateHostRequestBody,
        Proxy: false,
        PaidType: &paidTypeCreateHostRequestBody,
```

```
Exclusivelp: &exclusivelpCreateHostRequestBody,  
WebTag: &webTagCreateHostRequestBody,  
Certificatename: &certificatenameCreateHostRequestBody,  
Certificateid: &certificateidCreateHostRequestBody,  
Server: listServerbody,  
Hostname: "www.demo.com",  
}  
response, err := client.CreateHost(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.3 根据防护域名 Id 查询云模式防护域名详细信息

功能介绍

根据防护域名Id查询云模式防护域名详细信息

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/instance/{instance_id}

表 4-30 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	域名id, 您可以通过调用查询云模式防护域名列表 (ListHost) 获取域名id

表 4-31 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-32 请求 Header 参数

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

响应参数

状态码: 200

表 4-33 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	创建的云模式防护域名
policyid	String	防护域名的防护策略id
domainid	String	账号ID,对应控制台用户名->我的凭证->账号ID

参数	参数类型	描述
projectid	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
enterprise_project_id	String	企业项目ID, 对应控制台用户名->企业->项目管理->点击项目名称->ID
protocol	String	后端包含的协议类型: HTTPS、HTTP、HTTP&HTTPS
server	Array of CloudWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none"> • false: 不使用代理 • true: 使用代理
protect_status	Integer	域名防护状态: <ul style="list-style-type: none"> • -1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF • 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测 • 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
access_code	String	cname前缀, cname后缀 为.vip1.huaweicloudwaf.com
locked	Integer	预留参数, 用于后期设计冻结域名, 解锁域名功能, 目前暂不支持
timestamp	Long	创建防护域名的时间戳 (毫秒)
certificateid	String	https证书id
certificatename	String	证书名称
tls	String	配置的最低TLS版本 (TLS v1.0/TLS v1.1/TLS v1.2) ,默认为TLS v1.0版本, 对于低于最低TLS版本的请求, 将无法正常访问网站

参数	参数类型	描述
cipher	String	<p>加密套件 (cipher_1, cipher_2, cipher_3, cipher_4, cipher_5, cipher_6, cipher_default) :</p> <ul style="list-style-type: none"> • cipher_1: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2: 加密算法为EECDH+AESGCM:EDH+AESGCM • cipher_3: 加密算法为ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4: 加密算法为ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5: 加密算法为AES128-SHA:AES256-SHA:AES128-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default: 加密算法为ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
block_page	BlockPage object	告警页面配置
extend	Map<String,String>	扩展字段, 用于存放Web基础防护中一些开关配置等信息

参数	参数类型	描述
traffic_mark	TrafficMark object	流量标识
circuit_breaker	CircuitBreaker object	熔断配置
lb_algorithm	String	LB负载均衡，仅专业版（原企业版）和铂金版（原旗舰版）支持配置负载均衡算法 <ul style="list-style-type: none">源IP Hash：将某个IP的请求定向到同一个服务器加权轮询：所有请求将按权重轮流分配给源站服务器Session Hash：将某个Session标识的请求定向到同一个源站服务器，请确保在域名添加完毕后配置攻击惩罚的流量标识，否则Session Hash配置不生效
timeout_config	TimeoutConfig object	超时配置
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
flag	Flag object	特殊标识，用于前端使用
description	String	网站备注
http2_enable	Boolean	是否支持http2 <ul style="list-style-type: none">true：表示支持http2false：表示不支持http2
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip，该参数为预留参数，用于后续功能扩展，用户可忽略 <ul style="list-style-type: none">true：使用独享ipfalse：不使用独享ip
access_progress	Array of Access_progress objects	接入进度，仅用于新版console(前端)使用

参数	参数类型	描述
forward_header_map	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插入到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-34 CloudWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

表 4-35 BlockPage

参数	参数类型	描述
template	String	模板名称，默认页面请输入default，自定义告警页面时请输入custom，重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向”页面URL

表 4-36 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择text/html、text/xml和application/json三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-37 TrafficMark

参数	参数类型	描述
sip	Array of strings	IP标记，客户端最原始的IP地址的HTTP请求头字段。
cookie	String	Session标记，用于Cookie恶意请求的攻击惩罚功能。在选择Cookie拦截的攻击惩罚功能前，必须配置该标识
params	String	User标记，用于Params恶意请求的攻击惩罚功能。在选择Params拦截的攻击惩罚功能前，必须配置该标识。

表 4-38 CircuitBreaker

参数	参数类型	描述
switch	Boolean	熔断开关，是否开启连接保护 <ul style="list-style-type: none">• true: 开启连接保护• false: 关闭连接保护

参数	参数类型	描述
dead_num	Integer	502/504数量阈值，每30s累加的502/504数量阈值
dead_ratio	Number	502/504数量占比(%), 总请求数量中502/504数量占比达到所设定值, 并且与数量阈值同时满足时触发宕机保护
block_time	Integer	初次触发宕机的保护时间, 即WAF将停止转发用户请求的时间。
superposition_num	Integer	连续触发时, 保护时间延长最大倍数, 叠加周期为3600s。例如, “初次保护时间” 设置为180s, “连续触发叠加系数” 设置为3。 <ul style="list-style-type: none"> 当触发次数为2 (即小于3) 时, 保护时间为360s。 当次数大于等于3时, 保护时间为540s。 当累计保护时间超过1小时 (3600s), 叠加次数会从头计数。
suspend_num	Integer	读等待URL请求数量阈值, 读等待URL请求数量到达设定值即触发连接保护
sus_block_time	Integer	读等待URL请求数量超过阈值后的熔断时间, 达到数量阈值所触发的保护时间, 即WAF将停止转发用户请求的时间。

表 4-39 TimeoutConfig

参数	参数类型	描述
connect_timeout	Integer	WAF连接源站超时配置
send_timeout	Integer	WAF发送请求到源站超时配置
read_timeout	Integer	WAF接收源站响应超时配置

表 4-40 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启

参数	参数类型	描述
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

表 4-41 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none">• 1: 指回源IP加白• 2: 指本地验证• 3: 指修改DNS解析
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

状态码: 400

表 4-42 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-43 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-44 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-45 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-46 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-47 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

获取项目id为project_id, 防护域名Id为instance_id的云模式防护域名详细信息

```
GET https://{Endpoint}/v1/{project_id}/waf/instance/{instance_id}?enterprise_project_id=0
```

响应示例

状态码: 200

OK

```
{
  "id": "31af669f567246c289771694f2112289",
  "hostname": "www.demo.com",
  "protocol": "HTTP",
  "server": [
    {
      "address": "x.x.x.x",
      "port": 80,
      "type": "ipv4",
      "weight": 1,
      "front_protocol": "HTTP",
      "back_protocol": "HTTP"
    }
  ],
  "proxy": false,
  "locked": 0,
  "timestamp": 1650527546420,
  "flag": {
    "pci_3ds": "false",
    "pci_dss": "false",
    "ipv6": "false",
    "cname": "new",
    "is_dual_az": "true"
  },
  "description": "",
  "policyid": "41cba8aee2e94bcdbf57460874205494",
  "domainid": "d4ecb00b031941ce9171b7bc3386883f",
  "projectid": "0456cf04d6f64725ab02ed5bd2efdfa4",
}
```



```
"enterprise_project_id" : "0",
"protect_status" : 0,
"access_status" : 0,
"access_code" : "1b18879b9d064f8bbcbf8abce7294cac",
"block_page" : {
  "template" : "default"
},
"web_tag" : "",
"exclusive_ip" : false,
"http2_enable" : 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowHostRequest request = new ShowHostRequest();
        request.withInstanceId("{instance_id}");
        try {
            ShowHostResponse response = client.showHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = WafClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(WafRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ShowHostRequest()  
        request.instance_id = "{instance_id}"  
        response = client.show_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"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()
```

```
client := waf.NewWafClient(  
    waf.WafClientBuilder().  
        WithRegion(region.ValueOf("<YOUR REGION>")).  
        WithCredential(auth).  
        Build())  
  
request := &model.ShowHostRequest{}  
request.InstanceId = "{instance_id}"  
response, err := client.ShowHost(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.4 更新云模式防护域名的配置

功能介绍

更新云模式防护域名配置，在没有填入源站信息server的原始数据的情况下，则新的源站信息server会覆盖源站信息，而不是新增源站。此外，请求体可只传需要更新的部分。

调用方法

请参见[如何调用API](#)。

URI

PATCH /v1/{project_id}/waf/instance/{instance_id}

表 4-48 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	域名id，您可以通过调用查询云模式防护域名列表（ListHost）获取域名id

表 4-49 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-50 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)。
Content-Type	是	String	内容类型

表 4-51 请求 Body 参数

参数	是否必选	参数类型	描述
proxy	否	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">• false：不使用代理• true：使用代理

参数	是否必选	参数类型	描述
certificateid	否	String	证书id, 通过查询证书列表接口 (ListCertificates) 接口获取证书id <ul style="list-style-type: none">对外协议为HTTP时不需要填写对外协议HTTPS时为必填参数
certificatenam e	否	String	证书名 <ul style="list-style-type: none">对外协议为HTTP时不需要填写对外协议HTTPS时为必填参数
server	否	Array of CloudWafSer ver objects	防护域名的源站服务器配置信息
tls	否	String	配置的最低TLS版本 (TLS v1.0/TLS v1.1/TLS v1.2), 默认为TLS v1.0版本, 对于低于最低TLS版本的请求, 将无法正常访问网站

参数	是否必选	参数类型	描述
cipher	否	String	<p>加密套件 (cipher_1, cipher_2, cipher_3, cipher_4, cipher_5, cipher_6, cipher_default) :</p> <ul style="list-style-type: none"> • cipher_1: 加密算法为 ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2: 加密算法为 ECDH+AESGCM:EDH+AESGCM • cipher_3: 加密算法为 ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4: 加密算法为 ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5: 加密算法为 AES128-SHA:AES256-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6: 加密算法为 ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-

参数	是否必选	参数类型	描述
			AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default: 加密算法为 ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIG:MD5:aNULL:eNULL:NULL:DH:EDH:AESGCM
http2_enable	否	Boolean	是否支持http2 • true: 表示支持http2 • false: 表示不支持http2
ipv6_enable	否	Boolean	是否开启IPv6防护, 仅专业版(原企业版)和铂金版(原旗舰版)支持IPv6防护。 • true: 开启IPv6防护 • false: 关闭IPV6防护
web_tag	否	String	网站名称, 对应WAF控制台域名详情中的网站名称
exclusive_ip	否	Boolean	是否使用用户独享的引擎公网ip, 该参数为预留参数, 用于后续功能扩展, 用户可忽略 • true: 使用独享ip • false: 不使用独享ip
paid_type	否	String	套餐付费模式, 默认值为prePaid。prePaid: 包周期款模式; postPaid: 按需模式。
block_page	否	BlockPage object	告警页面配置
traffic_mark	否	TrafficMark object	流量标识
flag	否	Flag object	特殊标识, 用于前端使用
extend	否	Map<String,String>	扩展字段, 用于保存防护域名的一些配置信息。
circuit_breaker	否	CircuitBreaker object	熔断配置
timeout_config	否	TimeoutConfig object	超时配置

参数	是否必选	参数类型	描述
forward_header_map	否	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-52 CloudWafServer

参数	是否必选	参数类型	描述
front_protocol	是	String	客户端请求访问防护域名源站服务器的协议
back_protocol	是	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	否	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	是	String	客户端访问的源站服务器的IP地址
port	是	Integer	WAF转发客户端请求到源站服务的业务端口
type	是	String	源站地址为ipv4或ipv6

表 4-53 BlockPage

参数	是否必选	参数类型	描述
template	是	String	模板名称，默认页面请输入 default，自定义告警页面时请输入 custom，重定向请输入 redirect
custom_page	否	CustomPage object	自定义告警页面
redirect_url	否	String	“重定向”页面URL

表 4-54 CustomPage

参数	是否必选	参数类型	描述
status_code	是	String	返回状态码
content_type	是	String	“自定义”告警页面内容类型，可选择text/html、text/xml和 application/json三种类型
content	是	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-55 TrafficMark

参数	是否必选	参数类型	描述
sip	否	Array of strings	IP标记，客户端最原始的IP地址的HTTP请求头字段。
cookie	否	String	Session标记，用于Cookie恶意请求的攻击惩罚功能。在选择Cookie拦截的攻击惩罚功能前，必须配置该标识
params	否	String	User标记，用于Params恶意请求的攻击惩罚功能。在选择Params拦截的攻击惩罚功能前，必须配置该标识。

表 4-56 Flag

参数	是否必选	参数类型	描述
pci_3ds	否	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	否	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	否	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	否	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	否	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

表 4-57 CircuitBreaker

参数	是否必选	参数类型	描述
switch	否	Boolean	熔断开关, 是否开启连接保护 <ul style="list-style-type: none">• true: 开启连接保护• false: 关闭连接保护
dead_num	否	Integer	502/504数量阈值, 每30s累加的502/504数量阈值
dead_ratio	否	Number	502/504数量占比(%), 总请求数量中502/504数量占比达到所设定值, 并且与数量阈值同时满足时触发宕机保护
block_time	否	Integer	初次触发宕机的保护时间, 即WAF将停止转发用户请求的时间。

参数	是否必选	参数类型	描述
superposition_num	否	Integer	连续触发时，保护时间延长最大倍数，叠加周期为3600s。例如，“初次保护时间”设置为180s，“连续触发叠加系数”设置为3。 <ul style="list-style-type: none">当触发次数为2（即小于3）时，保护时间为360s。当次数大于等于3时，保护时间为540s。当累计保护时间超过1小时（3600s），叠加次数会从头计数。
suspend_num	否	Integer	读等待URL请求数量阈值，读等待URL请求数量到达设定值即触发连接保护
sus_block_time	否	Integer	读等待URL请求数量超过阈值后的熔断时间，达到数量阈值所触发的保护时间，即WAF将停止转发用户请求的时间。

表 4-58 TimeoutConfig

参数	是否必选	参数类型	描述
connect_timeout	否	Integer	WAF连接源站超时配置
send_timeout	否	Integer	WAF发送请求到源站超时配置
read_timeout	否	Integer	WAF接收源站响应超时配置

响应参数

状态码： 200

表 4-59 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	创建的云模式防护域名
policyid	String	防护域名的防护策略id

参数	参数类型	描述
domainid	String	账号ID,对应控制台用户名->我的凭证->账号ID
projectid	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
enterprise_project_id	String	企业项目ID, 对应控制台用户名->企业->项目管理->点击项目名称->ID
protocol	String	后端包含的协议类型: HTTPS、HTTP、HTTP&HTTPS
server	Array of CloudWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">• false: 不使用代理• true: 使用代理
protect_status	Integer	域名防护状态: <ul style="list-style-type: none">• -1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF• 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测• 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
access_code	String	cname前缀
locked	Integer	预留参数, 用于后期设计冻结域名, 解锁域名功能, 目前暂不支持
timestamp	Long	创建防护域名的时间戳(毫秒)
certificateid	String	https证书id
certificatename	String	证书名称
tls	String	配置的最低TLS版本(TLS v1.0/TLS v1.1/TLS v1.2),默认为TLS v1.0版本, 对于低于最低TLS版本的请求, 将无法正常访问网站

参数	参数类型	描述
cipher	String	<p>加密套件 (cipher_1, cipher_2, cipher_3, cipher_4, cipher_5, cipher_6, cipher_default) :</p> <ul style="list-style-type: none"> • cipher_1: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2: 加密算法为EECDH+AESGCM:EDH+AESGCM • cipher_3: 加密算法为ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4: 加密算法为ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5: 加密算法为AES128-SHA:AES256-SHA:AES128-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default: 加密算法为ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
block_page	BlockPage object	告警页面配置
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。

参数	参数类型	描述
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
traffic_mark	TrafficMark object	流量标识
circuit_breaker	CircuitBreaker object	熔断配置
lb_algorithm	String	LB负载均衡，仅专业版（原企业版）和铂金版（原旗舰版）支持配置负载均衡算法 <ul style="list-style-type: none">源IP Hash：将某个IP的请求定向到同一个服务器加权轮询：所有请求将按权重轮流分配给源站服务器Session Hash：将某个Session标识的请求定向到同一个源站服务器，请确保在域名添加完毕后配置攻击惩罚的流量标识，否则Session Hash配置不生效
timeout_config	TimeoutConfig object	超时配置
flag	Flag object	特殊标识，用于前端使用
description	String	网站备注
http2_enable	Boolean	是否支持http2 <ul style="list-style-type: none">true：表示支持http2false：表示不支持http2
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip，该参数为预留参数，用于后续功能扩展，用户可忽略 <ul style="list-style-type: none">true：使用独享ipfalse：不使用独享ip
access_progress	Array of Access_progress objects	接入进度，仅用于新版console(前端)使用

参数	参数类型	描述
forward_header_map	Map<String,String >	<p>字段转发配置，WAF会将添加的字段插入到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-60 CloudWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

表 4-61 BlockPage

参数	参数类型	描述
template	String	模板名称，默认页面请输入default，自定义告警页面时请输入custom，重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向”页面URL

表 4-62 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择text/html、text/xml和application/json三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-63 TrafficMark

参数	参数类型	描述
sip	Array of strings	IP标记，客户端最原始的IP地址的HTTP请求头字段。
cookie	String	Session标记，用于Cookie恶意请求的攻击惩罚功能。在选择Cookie拦截的攻击惩罚功能前，必须配置该标识
params	String	User标记，用于Params恶意请求的攻击惩罚功能。在选择Params拦截的攻击惩罚功能前，必须配置该标识。

表 4-64 CircuitBreaker

参数	参数类型	描述
switch	Boolean	熔断开关，是否开启连接保护 <ul style="list-style-type: none">• true: 开启连接保护• false: 关闭连接保护

参数	参数类型	描述
dead_num	Integer	502/504数量阈值，每30s累加的502/504数量阈值
dead_ratio	Number	502/504数量占比(%), 总请求数量中502/504数量占比达到所设定值, 并且与数量阈值同时满足时触发宕机保护
block_time	Integer	初次触发宕机的保护时间, 即WAF将停止转发用户请求的时间。
superposition_num	Integer	连续触发时, 保护时间延长最大倍数, 叠加周期为3600s。例如, “初次保护时间” 设置为180s, “连续触发叠加系数” 设置为3。 <ul style="list-style-type: none">当触发次数为2 (即小于3) 时, 保护时间为360s。当次数大于等于3时, 保护时间为540s。当累计保护时间超过1小时 (3600s), 叠加次数会从头计数。
suspend_num	Integer	读等待URL请求数量阈值, 读等待URL请求数量到达设定值即触发连接保护
sus_block_time	Integer	读等待URL请求数量超过阈值后的熔断时间, 达到数量阈值所触发的保护时间, 即WAF将停止转发用户请求的时间。

表 4-65 TimeoutConfig

参数	参数类型	描述
connect_timeout	Integer	WAF连接源站超时配置
send_timeout	Integer	WAF发送请求到源站超时配置
read_timeout	Integer	WAF接收源站响应超时配置

表 4-66 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">true: 开启false: 不开启

参数	参数类型	描述
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

表 4-67 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none">• 1: 指回源IP加白• 2: 指本地验证• 3: 指修改DNS解析
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

状态码: 400

表 4-68 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-69 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-70 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-71 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-72 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-73 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 域名id是instance_id的域名下的两个源站服务器, 第一个源站服务器的IP地址是x.x.x.x, 端口号是80, 源站地址是ipv4的, 源站权重是1, 客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议; 第二个源站服务器的IP地址是x.x.x.x, 端口号是80, 源站地址是ipv4的, 客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议; 域名关闭了ipv6防护。

```
PATCH https://{Endpoint}/v1/{project_id}/waf/instance/{instance_id}?enterprise_project_id=0
```

```
{
  "server": [ {
    "address": "x.x.x.x",
    "port": "80",
    "type": "ipv4",
    "weight": 1,
    "front_protocol": "HTTP",
    "back_protocol": "HTTP"
  }, {
    "front_protocol": "HTTP",
    "back_protocol": "HTTP",
    "type": "ipv4",
    "address": "x.x.x.x",
    "port": "80"
  } ],
  "ipv6_enable": false
}
```

响应示例

状态码: 200

OK

```
{
  "id": "e91ad96e379b4bea84f8fcda3d153370",
```

```
"hostname": "www.demo.com",
"protocol": "HTTP",
"server": [ {
  "address": "x.x.x.x",
  "port": 80,
  "type": "ipv4",
  "weight": 1,
  "front_protocol": "HTTP",
  "back_protocol": "HTTP"
}, {
  "address": "1.1.1.4",
  "port": 80,
  "type": "ipv4",
  "weight": 1,
  "front_protocol": "HTTP",
  "back_protocol": "HTTP"
} ],
"proxy": false,
"locked": 0,
"timestamp": 1650423573577,
"flag": {
  "pci_3ds": "false",
  "pci_dss": "false",
  "ipv6": "false",
  "cname": "new",
  "is_dual_az": "true"
},
"description": "",
"policyid": "f385eceedf7c4c34a4d1def19eafbe85",
"domainid": "d4ecb00b031941ce9171b7bc3386883f",
"projectid": "0456cf04d6f64725ab02ed5bd2efdfa4",
"enterprise_project_id": "0",
"protect_status": 1,
"access_status": 0,
"access_code": "4f5372610cdc44f7970759fcca138c81",
"block_page": {
  "template": "default"
},
"web_tag": "we",
"exclusive_ip": false,
"http2_enable": false
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，域名id是instance_id的域名下的两个源站服务器，第一个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，源站权重是1，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；第二个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；域名关闭了ipv6防护。

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

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

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

public class UpdateHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateHostRequest request = new UpdateHostRequest();
        request.withInstanceId("{instance_id}");
        UpdateHostRequestBody body = new UpdateHostRequestBody();
        List<CloudWafServer> listbodyServer = new ArrayList<>();
        listbodyServer.add(
            new CloudWafServer()
                .withFrontProtocol(CloudWafServer.FrontProtocolEnum.fromValue("HTTP"))
                .withBackProtocol(CloudWafServer.BackProtocolEnum.fromValue("HTTP"))
                .withWeight(1)
                .withAddress("x.x.x.x")
                .withPort(80)
                .withType(CloudWafServer.TypeEnum.fromValue("ipv4"))
        );
        listbodyServer.add(
            new CloudWafServer()
                .withFrontProtocol(CloudWafServer.FrontProtocolEnum.fromValue("HTTP"))
                .withBackProtocol(CloudWafServer.BackProtocolEnum.fromValue("HTTP"))
                .withAddress("x.x.x.x")
                .withPort(80)
                .withType(CloudWafServer.TypeEnum.fromValue("ipv4"))
        );
        body.withIpv6Enable(false);
        body.withServer(listbodyServer);
        request.withBody(body);
        try {
            UpdateHostResponse response = client.updateHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，域名id是instance_id的域名下的两个源站服务器，第一个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，源站权重是1，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；第二个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；域名关闭了ipv6防护。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateHostRequest()
        request.instance_id = "{instance_id}"
        listServerbody = [
            CloudWafServer(
                front_protocol="HTTP",
                back_protocol="HTTP",
                weight=1,
                address="x.x.x.x",
                port=80,
                type="ipv4"
            ),
            CloudWafServer(
                front_protocol="HTTP",
                back_protocol="HTTP",
                address="x.x.x.x",
                port=80,
                type="ipv4"
            )
        ]
        request.body = UpdateHostRequestBody(
            ipv6_enable=False,
            server=listServerbody
        )
        response = client.update_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

更新项目id为project_id，域名id是instance_id的域名下的两个源站服务器，第一个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，源站权重是1，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；第二个源站服务器的IP地址是x.x.x.x，端口号是80，源站地址是ipv4的，客户端请求访问防护域名源站服务器的协议和WAF转发客户端请求到防护域名源站服务器的协议都是HTTP协议；域名关闭了ipv6防护。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateHostRequest{}
    request.InstanceId = "{instance_id}"
    weightServer := int32(1)
    var listServerbody = []model.CloudWafServer{
        {
            FrontProtocol: model.GetCloudWafServerFrontProtocolEnum().HTTP,
            BackProtocol: model.GetCloudWafServerBackProtocolEnum().HTTP,
            Weight: &weightServer,
            Address: "x.x.x.x",
            Port: int32(80),
            Type: model.GetCloudWafServerTypeEnum().IPV4,
        },
        {
            FrontProtocol: model.GetCloudWafServerFrontProtocolEnum().HTTP,
            BackProtocol: model.GetCloudWafServerBackProtocolEnum().HTTP,
            Address: "x.x.x.x",
            Port: int32(80),
            Type: model.GetCloudWafServerTypeEnum().IPV4,
        },
    }
    ipv6EnableUpdateHostRequestBody := false
    request.Body = &model.UpdateHostRequestBody{
        Ipv6Enable: &ipv6EnableUpdateHostRequestBody,
        Server: &listServerbody,
    }
    response, err := client.UpdateHost(request)
    if err == nil {
```



```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.5 删除云模式防护域名

功能介绍

删除云模式防护域名

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/instance/{instance_id}

表 4-74 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	域名id，您可以通过调用查询云模式防护域名列表（ListHost）获取域名id

表 4-75 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-76 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-77 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostid	String	域名id
description	String	描述信息
type	Integer	WAF部署模式，默认是1，目前仅支持反代模式
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">false：不使用代理true：使用代理
flag	Flag object	特殊标识，用于前端使用
hostname	String	创建的云模式防护域名
access_code	String	cname后缀
policyid	String	策略id
timestamp	Long	创建防护域名的时间

参数	参数类型	描述
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">-1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	接入状态, 0: 未接入, 1: 已接入
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip, 该参数为预留参数, 用于后续功能扩展, 用户可忽略 <ul style="list-style-type: none">true: 使用独享ipfalse: 不使用独享ip
paid_type	String	套餐付费模式, 默认值为prePaid。 prePaid: 包周期款模式; postPaid: 按需模式。
web_tag	String	网站名称, 对应WAF控制台域名详情中的网站名称

表 4-78 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">true: 开启false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">true: 开启false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">true: 支持false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">true: 支持false: 不支持

状态码： 400**表 4-79** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-80 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-81** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-82 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-83 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-84 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

```
DELETE https://{Endpoint}/v1/{project_id}/waf/instance/{instance_id}?enterprise_project_id=0
```

删除项目id为project_id, 域名id的为instance_id的云模式域名。

响应示例

状态码： 200

OK

```
{  
  "id": "e91ad96e379b4bea84f8fcd3d153370",  
  "hostid": "e91ad96e379b4bea84f8fcd3d153370",  
  "description": "",  
  "type": 1,  
}
```

```
"proxy": true,
"flag": {
  "pci_3ds": "false",
  "pci_dss": "false",
  "ipv6": "true",
  "cname": "new",
  "is_dual_az": "true"
},
"region": "cn-north-4",
"hostname": "www.demo.com",
"access_code": "4f5372610cdc44f7970759fcca138c81",
"policyid": "f385eceedf7c4c34a4d1def19eafbe85",
"timestamp": 1650423573650,
"protect_status": 1,
"access_status": 0,
"exclusive_ip": false,
"web_tag": "we",
"paid_type": "prePaid"
}
```

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.6 修改域名防护状态

功能介绍

修改域名防护状态

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/instance/{instance_id}/protect-status

表 4-85 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

参数	是否必选	参数类型	描述
instance_id	是	String	域名id, 您可以通过调用查询云模式防护域名列表 (ListHost) 获取域名id

表 4-86 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-87 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)。
Content-Type	是	String	内容类型

表 4-88 请求 Body 参数

参数	是否必选	参数类型	描述
protect_status	是	Integer	域名防护状态: <ul style="list-style-type: none">• -1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF• 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测• 1: 开启防护, WAF根据您的配置的策略进行攻击检测

响应参数

状态码: 200

表 4-89 响应 Body 参数

参数	参数类型	描述
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">-1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测1: 开启防护, WAF根据您配置的策略进行攻击检测

状态码: 400

表 4-90 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-91 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-92 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-93 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-94 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-95 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id，域名id是instance_id的域名防护状态为暂停防护

```
PUT https://{Endpoint}/v1/{project_id}/waf/instance/{instance_id}/protect-status?enterprise_project_id=0
{
  "protect_status" : 0
}
```

响应示例

状态码： 200

请求成功

```
{
  "protect_status" : 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改项目id为project_id，域名id是instance_id的域名防护状态为暂停防护

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

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

public class UpdateHostProtectStatusSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateHostProtectStatusRequest request = new UpdateHostProtectStatusRequest();
        request.withInstanceId("{instance_id}");
        UpdateHostProtectStatusRequestBody body = new UpdateHostProtectStatusRequestBody();
        body.withProtectStatus(0);
        request.withBody(body);
    }
}
```

```
try {
    UpdateHostProtectStatusResponse response = client.updateHostProtectStatus(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

修改项目id为project_id，域名id是instance_id的域名防护状态为暂停防护

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateHostProtectStatusRequest()
        request.instance_id = "{instance_id}"
        request.body = UpdateHostProtectStatusRequestBody(
            protect_status=0
        )
        response = client.update_host_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

修改项目id为project_id，域名id是instance_id的域名防护状态为暂停防护

```
package main

import (
    "fmt"
```

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateHostProtectStatusRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.UpdateHostProtectStatusRequestBody{
        ProtectStatus: int32(0),
    }
    response, err := client.UpdateHostProtectStatus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.1.7 获取云模式域名路由信息

功能介绍

返回路由信息。

📖 说明

该API局点受限使用，后续将下线。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/instance/{instance_id}/route

表 4-96 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	域名id，您可以通过调用查询云模式防护域名列表（ListHost）获取域名id

表 4-97 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-98 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-99 响应 Body 参数

参数	参数类型	描述
total	Integer	路由线路数量
items	Array of RouteBody objects	路由线路列表

表 4-100 RouteBody

参数	参数类型	描述
cname	String	WAF集群的cname后缀
name	String	WAF集群名称
servers	Array of RouteServerBody objects	防护域名源站服务器信息列表

表 4-101 RouteServerBody

参数	参数类型	描述
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口

状态码： 400

表 4-102 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-103 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-104 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-105 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-106 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-107 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

获取项目id为project_id、域名id为instance_id的云模式域名路由信息

```
GET https://{Endpoint}/v1/{project_id}/waf/instance/{instance_id}/route?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{  
  "total": 1,  
  "items": [{  
    "cname": "e9d30fda6xxxxxxxxxxxx5b3d3286",
```



```
"name" : "Beijing",
"servers" : [ {
  "back_protocol" : "HTTP",
  "address" : "x.x.x.x",
  "port" : 80
} ]
} ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListHostRouteSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListHostRouteRequest request = new ListHostRouteRequest();
        request.withInstanceId("{instance_id}");
        try {
            ListHostRouteResponse response = client.listHostRoute(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListHostRouteRequest()
        request.instance_id = "{instance_id}"
        response = client.list_host_route(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2 独享模式防护网站管理

4.2.1 查询独享模式域名列表

功能介绍

查询独享模式域名列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/premium-waf/host

表 4-108 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID

表 4-109 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id。默认值为0，表示默认企业项目。 缺省值：0
page	否	String	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	String	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。如果需要一次查全部域名，该参数值填-1。
hostname	否	String	域名
policyname	否	String	策略名称
protect_status	否	Integer	域名防护状态： <ul style="list-style-type: none">0：暂停防护，WAF只转发该域名的请求，不做攻击检测1：开启防护，WAF根据您配置的策略进行攻击检测

请求参数

表 4-110 请求 Header 参数

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

响应参数

状态码： 200

表 4-111 响应 Body 参数

参数	参数类型	描述
total	Integer	全部防护域名的数量
items	Array of SimplePremiumWafHost objects	详细的防护域名信息数组

表 4-112 SimplePremiumWafHost

参数	参数类型	描述
id	String	域名id
hostname	String	域名
extend	Map<String,String >	扩展字段，用于保存防护域名的一些配置信息。
region	String	区域ID，控制台创建的域名会携带此参数，api调用创建的域名此参数为空，可以通过地区和终端节点文档查询区域ID对应的中文名称
flag	Flag object	特殊标识，用于前端使用
description	String	域名描述
policyid	String	防护域名初始绑定的防护策略ID,可以通过策略名称调用查询防护策略列表（ListPolicy）接口查询到对应的策略id
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">0：暂停防护，WAF只转发该域名的请求，不做攻击检测1：开启防护，WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态，0表示未接入，1表示已接入
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
hostid	String	域名id，和id的值是一样的，属于冗余字段

参数	参数类型	描述
enterprise_project_id	String	企业项目id

表 4-113 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

状态码: 400

表 4-114 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-115 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-116 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-117 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-118 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-119 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

获取项目id是project_id的独享模式域名列表

```
GET https://{Endpoint}/v1/{project_id}/premium-waf/host?enterprise_project_id=0
```

响应示例

状态码: 200

OK

```
{
  "total": 1,
  "items": [ {
    "id": "ee896796e1a84f3f85865ae0853d8974",
    "hostname": "www.demo.com",
    "extend": { },
    "region": "cn-north-4",
    "flag": {
      "pci_3ds": "false",
      "pci_dss": "false"
    },
    "description": "",
    "policyid": "df15d0eb84194950a8fdc615b6c012dc",
    "protect_status": 1,
    "access_status": 0,
    "hostid": "ee896796e1a84f3f85865ae0853d8974"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListPremiumHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPremiumHostRequest request = new ListPremiumHostRequest();
        try {
            ListPremiumHostResponse response = client.listPremiumHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = ListPremiumHostRequest()
    response = client.list_premium_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2.2 创建独享模式域名

功能介绍

创建独享模式域名

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/premium-waf/host

表 4-120 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-121 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-122 请求 Header 参数

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

表 4-123 请求 Body 参数

参数	是否必选	参数类型	描述
certificateid	否	String	证书id，通过查询证书列表接口（ListCertificates）接口获取证书id <ul style="list-style-type: none"> 对外协议为HTTP时不需要填写 对外协议HTTPS时为必填参数
certificatename	否	String	证书名 <ul style="list-style-type: none"> 对外协议为HTTP时不需要填写 对外协议HTTPS时为必填参数
hostname	是	String	防护域名或IP（可带端口）
proxy	是	Boolean	防护域名是否使用代理 <ul style="list-style-type: none"> false：不使用代理 true：使用代理
policyid	否	String	防护域名初始绑定的防护策略ID,可以通过策略名称调用查询防护策略列表（ListPolicy）接口查询到对应的策略id
server	是	Array of PremiumWaf Server objects	防护域名的源站服务器配置信息
block_page	否	BlockPage object	告警页面配置，该参数为非必填参数。当需要配置自定义页面时，该参数的子字段都为必填参数

参数	是否必选	参数类型	描述
forward_header_map	否	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused
mode	否	String	添加云模式elb接入域名时，请输入elb-shared，否则不输入
loadbalancer_id	否	String	负载均衡器（ELB）id,可以在ELB侧查询其id，添加云模式elb接入域名时，此为必须输入的值
listener_id	否	String	监听器id，可在ELB侧监听器页签下查询其id；不输入时，负载均衡器（ELB）下的所有监听器都将接入WAF防护，包括该ELB下未来新增的符合条件的监听器，添加云模式elb接入域名时，可考虑输入此项id
protocol_port	否	Integer	业务端口，添加云模式elb接入域名时，此为必须输入的值（0 - 65535）
description	否	String	防护域名备注

表 4-124 PremiumWafServer

参数	是否必选	参数类型	描述
front_protocol	是	String	客户端请求访问防护域名源站服务器的协议
back_protocol	是	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	否	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	是	String	客户端访问的源站服务器的IP地址
port	是	Integer	WAF转发客户端请求到源站服务的业务端口
type	是	String	源站地址为ipv4或ipv6
vpc_id	是	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none"> 1.找到独享引擎所在的虚拟私有云名称，VPC\子网这一列就是VPC的名称：登录WAF的控制台->单击系统管理->独享引擎->VPC\子网 2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

表 4-125 BlockPage

参数	是否必选	参数类型	描述
template	是	String	模板名称，默认页面请输入default，自定义告警页面时请输入custom，重定向请输入redirect
custom_page	否	CustomPage object	自定义告警页面
redirect_url	否	String	“重定向”页面URL

表 4-126 CustomPage

参数	是否必选	参数类型	描述
status_code	是	String	返回状态码
content_type	是	String	“自定义”告警页面内容类型，可选择text/html、text/xml和application/json三种类型
content	是	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

响应参数

状态码： 200

表 4-127 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	防护域名
protocol	String	对外协议，客户端（例如浏览器）请求访问网站的协议类型
server	Array of PremiumWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	是否使用代理 <ul style="list-style-type: none">true：代表使用代理false：代表未使用代理
locked	Integer	域名冻结状态，0表示未冻结，1表示为冻结，冗余参数
timestamp	Long	创建域名的时间，13位毫秒时间戳
tls	String	配置的最低TLS版本（TLS v1.0/TLS v1.1/TLS v1.2），默认为TLS v1.0版本，对于低于最低TLS版本的请求，将无法访问网站

参数	参数类型	描述
cipher	String	<p>对外协议为https时才有cipher参数，加密套件（cipher_1，cipher_2，cipher_3，cipher_4，cipher_5，cipher_6，cipher_default）：</p> <ul style="list-style-type: none"> • cipher_1：加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2：加密算法为EECDH+AESGCM:EDH+AESGCM • cipher_3：加密算法为ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4：加密算法为ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5：加密算法为AES128-SHA:AES256-SHA:AES128-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6：加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default：加密算法为ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。
flag	Flag object	特殊标识，用于前端使用

参数	参数类型	描述
mode	String	云模式elb接入域名返回此字段: elb-shared
loadbalancer_id	String	云模式elb接入域名返回此字段, 表示负载均衡器 (ELB) id
listener_id	String	云模式elb接入域名返回此字段, 表示监听器id
protocol_port	Integer	云模式elb接入域名返回此字段, 表示业务端口
description	String	域名描述
policyid	String	防护域名初始绑定的防护策略ID,可以通过策略名称调用查询防护策略列表 (ListPolicy) 接口查询到对应的策略id
domainid	String	账号ID,对应控制台用户名->我的凭证->账号ID
projectid	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
enterprise_project_id	String	企业项目ID, 对应控制台用户名->企业->项目管理->点击项目名称->ID
protect_status	Integer	域名防护状态: <ul style="list-style-type: none">● 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测● 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
block_page	BlockPage object	告警页面配置

参数	参数类型	描述
forward_header_map	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插入到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-128 PremiumWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

参数	参数类型	描述
vpc_id	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none"> 1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网 2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

表 4-129 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none"> true: 支持 false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none"> true: 支持 false: 不支持

表 4-130 BlockPage

参数	参数类型	描述
template	String	模板名称, 默认页面请输入default, 自定义告警页面时请输入custom, 重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向” 页面URL

表 4-131 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择 text/html、text/xml 和 application/json 三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

状态码：400

表 4-132 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码(加密)后的详细拒绝原因，用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码，可参考 STS5 联调自验证。IAM5 鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5 鉴权错误时才会返回此字段。

表 4-133 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码：401

表 4-134 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-135 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-136 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-137 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中创建创建独享模式域名，域名为"www.demo.com"，防护域名的源站服务器配置中，客户端请求访问防护域名源站服务器的协议和

WAF转发客户端请求到防护域名源站服务器的协议是HTTP协议，独享引擎所在的虚拟私有云id是"cf6dbace-b36a-4d51-ae04-52a3319ae247"，源站地址为ipv4，源站IP地址是x.x.x.x，WAF转发客户端请求到源站服务的业务端口是80。

```
POST https://{Endpoint}/v1/{project_id}/premium-waf/host?enterprise_project_id=0

{
  "hostname": "www.demo.com",
  "server": [ {
    "front_protocol": "HTTP",
    "back_protocol": "HTTP",
    "vpc_id": "cf6dbace-b36a-4d51-ae04-52a3319ae247",
    "type": "ipv4",
    "address": "x.x.x.x",
    "port": 80
  } ],
  "proxy": false,
  "description": ""
}
```

响应示例

状态码： 200

OK

```
{
  "id": "51a5649e52d341a9bb802044950969dc",
  "hostname": "www.demo.com",
  "protocol": "HTTP",
  "server": [ {
    "address": "x.x.x.x",
    "port": 80,
    "type": "ipv4",
    "weight": 1,
    "front_protocol": "HTTP",
    "back_protocol": "HTTP",
    "vpc_id": "cf6dbace-b36a-4d51-ae04-52a3319ae247"
  } ],
  "proxy": false,
  "locked": 0,
  "timestamp": 1650596007113,
  "flag": {
    "pci_3ds": "false",
    "pci_dss": "false"
  },
  "description": "",
  "policyid": "1607df035bc847b582ce9c838c083b88",
  "domainid": "d4ecb00b031941ce9171b7bc3386883f",
  "enterprise_project_id": "0",
  "protect_status": 1,
  "access_status": 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建创建独享模式域名，域名为"www.demo.com"，防护域名的源站服务器配置中，客户端请求访问防护域名源站服务器的协议和

WAF转发客户端请求到防护域名源站服务器的协议是HTTP协议，独享引擎所在的虚拟私有云id是"cf6dbace-b36a-4d51-ae04-52a3319ae247"，源站地址为ipv4，源站IP地址是x.x.x.x，WAF转发客户端请求到源站服务的业务端口是80。

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

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

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

public class CreatePremiumHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        CreatePremiumHostRequest request = new CreatePremiumHostRequest();
        CreatePremiumHostRequestBody body = new CreatePremiumHostRequestBody();
        List<PremiumWafServer> listbodyServer = new ArrayList<>();
        listbodyServer.add(
            new PremiumWafServer()
                .withFrontProtocol(PremiumWafServer.FrontProtocolEnum.fromValue("HTTP"))
                .withBackProtocol(PremiumWafServer.BackProtocolEnum.fromValue("HTTP"))
                .withAddress("x.x.x.x")
                .withPort(80)
                .withType(PremiumWafServer.TypeEnum.fromValue("ipv4"))
                .withVpcId("cf6dbace-b36a-4d51-ae04-52a3319ae247")
        );
        body.withDescription("");
        body.withServer(listbodyServer);
        body.withProxy(false);
        body.withHostname("www.demo.com");
        request.withBody(body);
        try {
            CreatePremiumHostResponse response = client.createPremiumHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

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

Python

在项目id为project_id的项目中创建独享模式域名，域名为"www.demo.com"，防护域名的源站服务器配置中，客户端请求访问防护域名源站服务器的协议和

WAF转发客户端请求到防护域名源站服务器的协议是HTTP协议，独享引擎所在的虚拟私有云id是"cf6dbace-b36a-4d51-ae04-52a3319ae247"，源站地址为ipv4，源站IP地址是x.x.x.x，WAF转发客户端请求到源站服务的业务端口是80。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreatePremiumHostRequest()
        listServerbody = [
            PremiumWafServer(
                front_protocol="HTTP",
                back_protocol="HTTP",
                address="x.x.x.x",
                port=80,
                type="ipv4",
                vpc_id="cf6dbace-b36a-4d51-ae04-52a3319ae247"
            )
        ]
        request.body = CreatePremiumHostRequestBody(
            description="",
            server=listServerbody,
            proxy=False,
            hostname="www.demo.com"
        )
        response = client.create_premium_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

在项目id为project_id的项目中创建创建独享模式域名，域名为"www.demo.com"，防护域名的源站服务器配置中，客户端请求访问防护域名源站服务器的协议和

WAF转发客户端请求到防护域名源站服务器的协议是HTTP协议，独享引擎所在的虚拟私有云id是"cf6dbace-b36a-4d51-ae04-52a3319ae247"，源站地址为ipv4，源站IP地址是x.x.x.x，WAF转发客户端请求到源站服务的业务端口是80。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreatePremiumHostRequest{}
    var listServerbody = []model.PremiumWafServer{
        {
            FrontProtocol: model.GetPremiumWafServerFrontProtocolEnum().HTTP,
            BackProtocol: model.GetPremiumWafServerBackProtocolEnum().HTTP,
            Address: "x.x.x.x",
            Port: int32(80),
            Type: model.GetPremiumWafServerTypeEnum().IPV4,
            Vpclid: "cf6dbace-b36a-4d51-ae04-52a3319ae247",
        },
    }
    descriptionCreatePremiumHostRequestBody := ""
    request.Body = &model.CreatePremiumHostRequestBody{
        Description: &descriptionCreatePremiumHostRequestBody,
        Server: listServerbody,
        Proxy: false,
        Hostname: "www.demo.com",
    }
    response, err := client.CreatePremiumHost(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

更多

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

状态码

状态码	描述
200	OK
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2.3 修改独享模式域名配置

功能介绍

修改独享模式域名配置，在没有填入源站信息server的原始数据的情况下，则新的源站信息server会覆盖源站信息，而不是新增源站。此外，请求体可只传需要更新的部分。

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/premium-waf/host/{host_id}

表 4-138 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
host_id	是	String	独享模式域名ID

表 4-139 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-140 请求 Header 参数

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

表 4-141 请求 Body 参数

参数	是否必选	参数类型	描述
proxy	否	Boolean	防护域名是否使用代理 <ul style="list-style-type: none"> • false: 不使用代理 • true: 使用代理
certificateid	否	String	证书id，通过查询证书列表接口（ListCertificates）接口获取证书id <ul style="list-style-type: none"> • 对外协议为HTTP时不需要填写 • 对外协议HTTPS时为必填参数
certificatename	否	String	证书名 <ul style="list-style-type: none"> • 对外协议为HTTP时不需要填写 • 对外协议HTTPS时为必填参数
server	否	Array of PremiumWaf Server objects	防护域名的源站服务器配置信息

参数	是否必选	参数类型	描述
tls	否	String	配置的最低TLS版本（TLS v1.0/TLS v1.1/TLS v1.2），默认为TLS v1.0版本，对于低于最低TLS版本的请求，将无法正常访问网站

参数	是否必选	参数类型	描述
cipher	否	String	<p>加密套件 (cipher_1, cipher_2, cipher_3, cipher_4, cipher_5, cipher_6, cipher_default) :</p> <ul style="list-style-type: none"> • cipher_1: 加密算法为 ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2: 加密算法为 ECDH+AESGCM:EDH+AESGCM • cipher_3: 加密算法为 ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4: 加密算法为 ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5: 加密算法为 AES128-SHA:AES256-SHA:AES256-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6: 加密算法为 ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-

参数	是否必选	参数类型	描述
			AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default: 加密算法为 ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
mode	否	String	独享模式特殊域名模式（仅特殊模式需要，如elb）
locked	否	Integer	预留参数，用于后期设计冻结域名，解锁域名功能，目前暂不支持
protect_status	否	Integer	域名防护状态： • 0：暂停防护，WAF只转发该域名的请求，不做攻击检测 • 1：开启防护，WAF根据您配置的策略进行攻击检测
access_status	否	Integer	域名接入状态，0表示未接入，1表示已接入
timestamp	否	Integer	时间戳
pool_ids	否	Array of strings	特殊模式域名所属独享引擎组（仅特殊模式需要，如elb）
block_page	否	BlockPage object	告警页面配置，该参数为非必填参数。当需要配置自定义页面时，该参数的子字段都为必填参数
traffic_mark	否	TrafficMark object	流量标识
circuit_breaker	否	CircuitBreaker object	熔断配置
timeout_config	否	TimeoutConfig object	超时配置
flag	否	HostFlag object	配置独享防护域名合规认证开关

参数	是否必选	参数类型	描述
forward_header_map	否	Map<String,String>	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host • \$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused

表 4-142 PremiumWafServer

参数	是否必选	参数类型	描述
front_protocol	是	String	客户端请求访问防护域名源站服务器的协议
back_protocol	是	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	否	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	是	String	客户端访问的源站服务器的IP地址
port	是	Integer	WAF转发客户端请求到源站服务的业务端口
type	是	String	源站地址为ipv4或ipv6

参数	是否必选	参数类型	描述
vpc_id	是	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none"> 1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网 2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

表 4-143 BlockPage

参数	是否必选	参数类型	描述
template	是	String	模板名称, 默认页面请输入 default, 自定义告警页面时请输入 custom, 重定向请输入 redirect
custom_page	否	CustomPage object	自定义告警页面
redirect_url	否	String	“重定向” 页面URL

表 4-144 CustomPage

参数	是否必选	参数类型	描述
status_code	是	String	返回状态码
content_type	是	String	“自定义” 告警页面内容类型, 可选择text/html、text/xml和 application/json三种类型
content	是	String	根据选择的“页面类型” 配置对应的页面内容, 具体示例可以参考“Web应用防火墙 WAF” 用户手册

表 4-145 TrafficMark

参数	是否必选	参数类型	描述
sip	否	Array of strings	IP标记，客户端最原始的IP地址的HTTP请求头字段。
cookie	否	String	Session标记，用于Cookie恶意请求的攻击惩罚功能。在选择Cookie拦截的攻击惩罚功能前，必须配置该标识
params	否	String	User标记，用于Params恶意请求的攻击惩罚功能。在选择Params拦截的攻击惩罚功能前，必须配置该标识。

表 4-146 CircuitBreaker

参数	是否必选	参数类型	描述
switch	否	Boolean	熔断开关，是否开启连接保护 <ul style="list-style-type: none"> • true: 开启连接保护 • false: 关闭连接保护
dead_num	否	Integer	502/504数量阈值，每30s累加的502/504数量阈值
dead_ratio	否	Number	502/504数量占比(%), 总请求数量中502/504数量占比达到所设定值，并且与数量阈值同时满足时触发宕机保护
block_time	否	Integer	初次触发宕机的保护时间，即WAF将停止转发用户请求的时间。
superposition_num	否	Integer	连续触发时，保护时间延长最大倍数，叠加周期为3600s。例如，“初次保护时间”设置为180s，“连续触发叠加系数”设置为3。 <ul style="list-style-type: none"> • 当触发次数为2（即小于3）时，保护时间为360s。 • 当次数大于等于3时，保护时间为540s。 • 当累计保护时间超过1小时（3600s），叠加次数会从头计数。

参数	是否必选	参数类型	描述
suspend_num	否	Integer	读等待URL请求数量阈值，读等待URL请求数量到达设定值即触发连接保护
sus_block_time	否	Integer	读等待URL请求数量超过阈值后的熔断时间，达到数量阈值所触发的保护时间，即WAF将停止转发用户请求的时间。

表 4-147 TimeoutConfig

参数	是否必选	参数类型	描述
connect_timeout	否	Integer	WAF连接源站超时配置
send_timeout	否	Integer	WAF发送请求到源站超时配置
read_timeout	否	Integer	WAF接收源站响应超时配置

表 4-148 HostFlag

参数	是否必选	参数类型	描述
pci_3ds	否	String	是否开启pci_3ds合规认证，该参数需要与tls和cipher参数同时使用，且tls参数值需要设置为TLS v1.2，cipher参数值设置为cipher_2。注：pci_3ds合规认证开启后不支持关闭，在开启pci_3ds合规认证前，请先阅读帮助中心Web应用防火墙WAF文档中关于pci_3ds合规认证的说明 <ul style="list-style-type: none"> • true: 开启 • false: 不开启
pci_dss	否	String	是否开启pci_dss合规认证，该参数需要与tls和cipher参数同时使用，且tls参数值需要设置为TLS v1.2，cipher参数值设置为cipher_2。注：在开启pci_dss合规认证前，请先阅读帮助中心Web应用防火墙WAF文档中关于pci_dss合规认证的说明 <ul style="list-style-type: none"> • true: 开启 • false: 不开启

响应参数

状态码： 200

表 4-149 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	创建的独享模式防护域名
protocol	String	对外协议，客户端（例如浏览器）请求访问网站的协议类型
server	Array of PremiumWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">• false：不使用代理• true：使用代理
locked	Integer	预留参数，用于后期设计冻结域名，解锁域名功能，目前暂不支持
timestamp	Long	创建防护域名的时间
tls	String	配置的最低TLS版本（TLS v1.0/TLS v1.1/TLS v1.2），默认为TLS v1.0版本，对于低于最低TLS版本的请求，将无法访问网站

参数	参数类型	描述
cipher	String	<p>加密套件 (cipher_1, cipher_2, cipher_3, cipher_4, cipher_5, cipher_6, cipher_default) :</p> <ul style="list-style-type: none"> • cipher_1: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2: 加密算法为EECDH+AESGCM:EDH+AESGCM • cipher_3: 加密算法为ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4: 加密算法为ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5: 加密算法为AES128-SHA:AES256-SHA:AES128-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6: 加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default: 加密算法为ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。
flag	Flag object	特殊标识，用于前端使用

参数	参数类型	描述
mode	String	云模式elb接入域名返回此字段: elb-shared
loadbalancer_id	String	云模式elb接入域名返回此字段, 表示负载均衡器 (ELB) id
listener_id	String	云模式elb接入域名返回此字段, 表示监听器id
protocol_port	Integer	云模式elb接入域名返回此字段, 表示业务端口
description	String	域名描述
policyid	String	防护域名初始绑定的策略ID,可以通过策略名称调用查询防护策略列表 (ListPolicy) 接口查询到对应的策略id
domainid	String	账号ID,对应控制台用户名->我的凭证->账号ID
projectid	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
enterprise_project_id	String	企业项目ID, 对应控制台用户名->企业->项目管理->点击项目名称->ID
certificateid	String	https证书id
certificatename	String	证书名称
protect_status	Integer	域名防护状态: <ul style="list-style-type: none">● 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测● 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
web_tag	String	网站名称, 对应WAF控制台域名详情中的网站名称
lb_algorithm	String	LB负载均衡, 默认轮询, 不支持修改
block_page	BlockPage object	告警页面配置
traffic_mark	TrafficMark object	流量标识
timeout_config	TimeoutConfig object	超时配置

参数	参数类型	描述
forward_header_map	Map<String,String >	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host -\$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused
access_progress	Array of Access_progress objects	接入进度，仅用于新版console(前端)使用

表 4-150 PremiumWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

参数	参数类型	描述
vpc_id	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none"> 1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网 2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

表 4-151 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none"> true: 支持 false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none"> true: 支持 false: 不支持

表 4-152 BlockPage

参数	参数类型	描述
template	String	模板名称, 默认页面请输入default, 自定义告警页面时请输入custom, 重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向” 页面URL

表 4-153 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择 text/html、text/xml 和 application/json 三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-154 TrafficMark

参数	参数类型	描述
sip	Array of strings	IP 标记，客户端最原始的 IP 地址的 HTTP 请求头字段。
cookie	String	Session 标记，用于 Cookie 恶意请求的攻击惩罚功能。在选择 Cookie 拦截的攻击惩罚功能前，必须配置该标识
params	String	User 标记，用于 Params 恶意请求的攻击惩罚功能。在选择 Params 拦截的攻击惩罚功能前，必须配置该标识。

表 4-155 TimeoutConfig

参数	参数类型	描述
connect_timeout	Integer	WAF 连接源站超时配置
send_timeout	Integer	WAF 发送请求到源站超时配置
read_timeout	Integer	WAF 接收源站响应超时配置

表 4-156 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none"> • 1: 指回源 IP 加白 • 2: 指本地验证 • 3: 指修改 DNS 解析

参数	参数类型	描述
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

状态码: 400

表 4-157 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-158 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-159 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-160 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-161 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-162 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id和独享域名id为host_id的独享域名代理使用情况为不使用代理

```
PUT https://{Endpoint}/v1/{project_id}/premium-waf/host/{host_id}?enterprise_project_id=0
{
```

```
"proxy" : false  
}
```

响应示例

状态码： 200

OK

```
{  
  "id" : "27995fb98a2d4928a1e453e65ee8117a",  
  "hostname" : "www.demo.com",  
  "protocol" : "HTTP",  
  "server" : [ {  
    "address" : "192.168.0.209",  
    "port" : 80,  
    "type" : "ipv4",  
    "weight" : 1,  
    "front_protocol" : "HTTP",  
    "back_protocol" : "HTTP",  
    "vpc_id" : "cf6dbace-b36a-4d51-ae04-52a8459ae247"  
  } ],  
  "proxy" : false,  
  "locked" : 0,  
  "timestamp" : 1650590814885,  
  "flag" : {  
    "pci_3ds" : "false",  
    "pci_dss" : "false"  
  },  
  "description" : "",  
  "policyid" : "9555cda636ef4ca294dfe4b14bc94c47",  
  "domainid" : "d4ecb00b031941ce9171b7bc3386883f",  
  "projectid" : "05e33ecd328025dd2f7fc00696201fb4",  
  "enterprise_project_id" : "0",  
  "protect_status" : 1,  
  "access_status" : 0  
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id和独享域名id为host_id的独享域名代理使用情况为不使用代理

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class UpdatePremiumHostSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
UpdatePremiumHostRequest request = new UpdatePremiumHostRequest();
request.withHostId("{host_id}");
UpdatePremiumHostRequestBody body = new UpdatePremiumHostRequestBody();
body.withProxy(false);
request.withBody(body);
try {
    UpdatePremiumHostResponse response = client.updatePremiumHost(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id和独享域名id为host_id的独享域名代理使用情况为不使用代理

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePremiumHostRequest()
        request.host_id = "{host_id}"
        request.body = UpdatePremiumHostRequestBody(
            proxy=False
```

```
)
response = client.update_premium_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

在项目id为project_id和独享域名id为host_id的独享域名代理使用情况为不使用代理

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePremiumHostRequest{}
    request.HostId = "{host_id}"
    proxyUpdatePremiumHostRequestBody := false
    request.Body = &model.UpdatePremiumHostRequestBody{
        Proxy: &proxyUpdatePremiumHostRequestBody,
    }
    response, err := client.UpdatePremiumHost(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2.4 查看独享模式域名配置

功能介绍

查看独享模式域名配置

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/premium-waf/host/{host_id}

表 4-163 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
host_id	是	String	独享模式域名ID

表 4-164 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-165 请求 Header 参数

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

响应参数

状态码： 200

表 4-166 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	创建的独享模式防护域名
protocol	String	对外协议，客户端（例如浏览器）请求访问网站的协议类型
server	Array of PremiumWafServer objects	防护域名的源站服务器配置信息
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">false：不使用代理true：使用代理
locked	Integer	预留参数，用于后期设计冻结域名，解锁域名功能，目前暂不支持
timestamp	Long	创建防护域名的时间
tls	String	配置的最低TLS版本（TLS v1.0/TLS v1.1/TLS v1.2），默认为TLS v1.0版本，对外协议为https时才有tls参数

参数	参数类型	描述
cipher	String	<p>对外协议为https时才有cipher参数，加密套件（cipher_1，cipher_2，cipher_3，cipher_4，cipher_5，cipher_6，cipher_default）：</p> <ul style="list-style-type: none"> • cipher_1：加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!DES:!MD5:!PSK:!RC4:!kRSA:!SRP:!3DES:!DSS:!EXP:!CAMELLIA:@STRENGTH • cipher_2：加密算法为EECDH+AESGCM:EDH+AESGCM • cipher_3：加密算法为ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH • cipher_4：加密算法为ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!EDH • cipher_5：加密算法为AES128-SHA:AES256-SHA:AES128-SHA256:AES256-SHA256:HIGH:!MEDIUM:!LOW:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4:!DHE:@STRENGTH • cipher_6：加密算法为ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256 • cipher_default：加密算法为ECDHE-RSA-AES256-SHA384:AES256-SHA256:RC4:HIGH:!MD5:!aNULL:!eNULL:!NULL:!DH:!EDH:!AESGCM
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。
flag	Flag object	特殊标识，用于前端使用

参数	参数类型	描述
mode	String	云模式elb接入域名返回此字段: elb-shared
loadbalancer_id	String	云模式elb接入域名返回此字段, 表示负载均衡器 (ELB) id
listener_id	String	云模式elb接入域名返回此字段, 表示监听器id
protocol_port	Integer	云模式elb接入域名返回此字段, 表示业务端口
description	String	域名描述
policyid	String	防护域名初始绑定的策略ID,可以通过策略名称调用查询防护策略列表 (ListPolicy) 接口查询到对应的策略id
domainid	String	账号ID,对应控制台用户名->我的凭证->账号ID
projectid	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
enterprise_project_id	String	企业项目ID, 对应控制台用户名->企业->项目管理->点击项目名称->ID
certificateid	String	https证书id
certificatename	String	证书名称
protect_status	Integer	域名防护状态: <ul style="list-style-type: none"> 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
web_tag	String	网站名称, 对应WAF控制台域名详情中的网站名称
block_page	BlockPage object	告警页面配置
traffic_mark	TrafficMark object	流量标识
timeout_config	TimeoutConfig object	超时配置

参数	参数类型	描述
forward_header_map	Map<String,String >	<p>字段转发配置，WAF会将添加的字段插到header中，转给源站；Key不能跟nginx原生字段重复。Value支持的值包括：</p> <ul style="list-style-type: none"> • \$time_local • \$request_id • \$connection_requests • \$tenant_id • \$project_id • \$remote_addr • \$remote_port • \$scheme • \$request_method • \$http_host -\$origin_uri • \$request_length • \$ssl_server_name • \$ssl_protocol • \$ssl_curves • \$ssl_session_reused
access_progress	Array of Access_progress objects	接入进度，仅用于新版console(前端)使用

表 4-167 PremiumWafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重，负载均衡算法将按该权重将请求分配给源站，默认值是1，云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

参数	参数类型	描述
vpc_id	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none"> 1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网 2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

表 4-168 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none"> true: 开启 false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none"> true: 支持 false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none"> true: 支持 false: 不支持

表 4-169 BlockPage

参数	参数类型	描述
template	String	模板名称, 默认页面请输入default, 自定义告警页面时请输入custom, 重定向请输入redirect
custom_page	CustomPage object	自定义告警页面
redirect_url	String	“重定向” 页面URL

表 4-170 CustomPage

参数	参数类型	描述
status_code	String	返回状态码
content_type	String	“自定义”告警页面内容类型，可选择 text/html、text/xml 和 application/json 三种类型
content	String	根据选择的“页面类型”配置对应的页面内容，具体示例可以参考“Web应用防火墙 WAF”用户手册

表 4-171 TrafficMark

参数	参数类型	描述
sip	Array of strings	IP标记，客户端最原始的IP地址的HTTP请求头字段。
cookie	String	Session标记，用于Cookie恶意请求的攻击惩罚功能。在选择Cookie拦截的攻击惩罚功能前，必须配置该标识
params	String	User标记，用于Params恶意请求的攻击惩罚功能。在选择Params拦截的攻击惩罚功能前，必须配置该标识。

表 4-172 TimeoutConfig

参数	参数类型	描述
connect_timeout	Integer	WAF连接源站超时配置
send_timeout	Integer	WAF发送请求到源站超时配置
read_timeout	Integer	WAF接收源站响应超时配置

表 4-173 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none"> • 1: 指回源IP加白 • 2: 指本地验证 • 3: 指修改DNS解析

参数	参数类型	描述
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

状态码: 400

表 4-174 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-175 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-176 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-177 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-178 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-179 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查看项目id为project_id，独享模式域名id为host_id的独享域名配置。

```
GET https://{Endpoint}/v1/{project_id}/premium-waf/host/{host_id}?enterprise_project_id=0
```

响应示例

状态码: 200

OK

```
{
  "id": "ee896796e1a84f3f85865ae0853d8974",
  "hostname": "www.demo.com",
  "protocol": "HTTPS",
  "server": [ {
    "address": "1.2.3.4",
    "port": 443,
    "type": "ipv4",
    "weight": 1,
    "front_protocol": "HTTPS",
    "back_protocol": "HTTPS",
    "vpc_id": "ebfc553a-386d-4746-b0c2-18ff3f0e903d"
  } ],
  "proxy": false,
  "locked": 0,
  "timestamp": 1650593801380,
  "tls": "TLS v1.0",
  "cipher": "cipher_1",
  "flag": {
    "pci_3ds": "false",
    "pci_dss": "false"
  },
  "description": "",
  "policyid": "df15d0eb84194950a8fdc615b6c012dc",
  "domainid": "0ee78615ca08419f81f539d97c9ee353",
  "projectid": "550500b49078408682d0d4f7d923f3e1",
  "protect_status": 1,
  "access_status": 0,
  "certificateid": "360f992501a64de0a65c50a64d1ca7b3",
  "certificatename": "certificatename75315"
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowPremiumHostSolution {

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

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ShowPremiumHostRequest request = new ShowPremiumHostRequest();
request.withHostId("{host_id}");
try {
    ShowPremiumHostResponse response = client.showPremiumHost(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowPremiumHostRequest()
        request.host_id = "{host_id}"
        response = client.show_premium_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2.5 删除独享模式域名

功能介绍

删除独享模式域名

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/premium-waf/host/{host_id}

表 4-180 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
host_id	是	String	独享模式域名ID

表 4-181 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
keepPolicy	否	Boolean	是否保留规则。false表示不保留该域名的防护策略；true表示保留该域名的防护策略。当要删除的防护域名的防护策略防护多个防护域名时，该参数不传。

请求参数

表 4-182 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

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

响应参数

状态码： 200

表 4-183 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostname	String	域名
extend	Map<String,String>	扩展字段，用于保存防护域名的一些配置信息。
region	String	区域ID，控制台创建的域名会携带此参数，api调用创建的域名此参数为空，可以通过地区和终端节点文档查询区域ID对应的中文名称
flag	Flag object	特殊标识，用于前端使用
description	String	域名描述
policyid	String	防护域名初始绑定的防护策略ID,可以通过策略名称调用查询防护策略列表 (ListPolicy) 接口查询到对应的策略id
protect_status	Integer	域名防护状态： <ul style="list-style-type: none"> 0：暂停防护，WAF只转发该域名的请求，不做攻击检测 1：开启防护，WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态，0表示未接入，1表示已接入
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称
host_id	String	域名id，和id的值是一样的，属于冗余字段

表 4-184 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

状态码: 400

表 4-185 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-186 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码

参数	参数类型	描述
error_msg	String	下游服务报错信息

状态码： 401

表 4-187 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-188 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-189 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-190 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中删除独享域名id为host_id的独享域名。

```
DELETE https://{Endpoint}/v1/{project_id}/premium-waf/host/{host_id}?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{
  "id" : "ee896796e1a84f3f85865ae0853d8974",
  "hostname" : "www.demo.com",
  "region" : "cn-north-4",
  "flag" : {
    "pci_3ds" : "false",
    "pci_dss" : "false"
  },
  "description" : "",
  "policyid" : "df15d0eb84194950a8fdc615b6c012dc",
  "protect_status" : 1,
  "access_status" : 0,
  "hostid" : "ee896796e1a84f3f85865ae0853d8974"
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;
```

```
public class DeletePremiumHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeletePremiumHostRequest request = new DeletePremiumHostRequest();
        request.withHostId("{host_id}");
        try {
            DeletePremiumHostResponse response = client.deletePremiumHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

```
try:
    request = DeletePremiumHostRequest()
    request.host_id = "{host_id}"
    response = client.delete_premium_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK

状态码	描述
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.2.6 修改独享模式域名防护状态

功能介绍

修改独享模式域名防护状态

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/premium-waf/host/{host_id}/protect-status

表 4-191 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
host_id	是	String	独享模式域名ID

表 4-192 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-193 请求 Header 参数

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

表 4-194 请求 Body 参数

参数	是否必选	参数类型	描述
protect_status	是	Integer	域名防护状态： <ul style="list-style-type: none">0：暂停防护，WAF只转发该域名的请求，不做攻击检测1：开启防护，WAF根据您配置的策略进行攻击检测

响应参数

状态码： 200

表 4-195 响应 Body 参数

参数	参数类型	描述
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">0：暂停防护，WAF只转发该域名的请求，不做攻击检测1：开启防护，WAF根据您配置的策略进行攻击检测

状态码： 400

表 4-196 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-197 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-198 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-199 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-200 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-201 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id和域名id为host_id的独享域名的防护状态为开启防护

```
PUT https://{Endpoint}/v1/{project_id}/premium-waf/host/{host_id}/protect-status?enterprise_project_id=0
{
  "protect_status" : 1
}
```

响应示例

状态码： 200

OK

```
{
  "protect_status" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改项目id为project_id和域名id为host_id的独享域名的防护状态为开启防护

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

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

public class UpdatePremiumHostProtectStatusSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePremiumHostProtectStatusRequest request = new UpdatePremiumHostProtectStatusRequest();
        request.withHostId("{host_id}");
        UpdatePremiumHostProtectStatusRequestBody body = new
        UpdatePremiumHostProtectStatusRequestBody();
        body.withProtectStatus(1);
        request.withBody(body);
        try {
            UpdatePremiumHostProtectStatusResponse response =
            client.updatePremiumHostProtectStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改项目id为project_id和域名id为host_id的独享域名的防护状态为开启防护

```
# coding: utf-8

import os
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePremiumHostProtectStatusRequest()
        request.host_id = "{host_id}"
        request.body = UpdatePremiumHostProtectStatusRequestBody(
            protect_status=1
        )
        response = client.update_premium_host_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

修改项目id为project_id和域名id为host_id的独享域名的防护状态为开启防护

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

```
WithCredential(auth).
Build()

request := &model.UpdatePremiumHostProtectStatusRequest{}
request.HostId = "{host_id}"
request.Body = &model.UpdatePremiumHostProtectStatusRequestBody{
    ProtectStatus: int32(1),
}
response, err := client.UpdatePremiumHostProtectStatus(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	OK
400	非法请求
401	Token无权限
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3 防护策略管理

4.3.1 查询防护策略列表

功能介绍

查询防护策略列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy

表 4-202 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID

表 4-203 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。
pagesize	否	Integer	分页查询时, 每页包含多少条结果。范围1-100, 默认值为10, 表示每页包含10条结果。
name	否	String	策略名称

请求参数

表 4-204 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-205 响应 Body 参数

参数	参数类型	描述
total	Integer	防护策略的数量

参数	参数类型	描述
items	Array of PolicyResponse objects	防护策略的具体内容

表 4-206 PolicyResponse

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名
level	Integer	Web基础防护等级 <ul style="list-style-type: none"> • 1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。 • 2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。 • 3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none"> • false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截 • true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String, Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	Array of strings	与防护策略绑定的防护的域名id数组

参数	参数类型	描述
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String,String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-207 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none"> log: 仅记录 block: 拦截

表 4-208 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-209 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启
webshell	Boolean	Webshell检测是否开启

参数	参数类型	描述
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启，该特性是公测特性，在公测期间，只支持仅记录模式。

表 4-210 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码：400

表 4-211 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-212 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-213 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-214 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-215 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-216 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

获取项目id为project_id的防护策略列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "id": "41cba8aee2e94bcd57460874205494",
    "name": "policy_2FHwFOKz",
    "level": 2,
    "action": {
      "category": "log"
    },
  },
  "options": {
    "webattack": true,
    "common": true,
    "crawler": true,
    "crawler_engine": false,
    "crawler_scanner": true,
    "crawler_script": false,
    "crawler_other": false,
    "webshell": false,
    "cc": true,
    "custom": true,
  }
}
```

```
"whiteblackip" : true,
"geop" : true,
"ignore" : true,
"privacy" : true,
"antitamper" : true,
"antileakage" : false,
"bot_enable" : true,
"modulex_enabled" : false
},
"hosts" : [ ],
"extend" : { },
"timestamp" : 1650527546218,
"full_detection" : false,
"bind_host" : [ ]
}]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListPolicySolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPolicyRequest request = new ListPolicyRequest();
        try {
            ListPolicyResponse response = client.listPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListPolicyRequest()
        response = client.list_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

```
Build()

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3.2 创建防护策略

功能介绍

创建防护策略，系统会在生成策略时配置一些默认的配置项，如果需要修改策略的默认配置项需要通过调用更新防护策略接口实现

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy

表 4-217 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-218 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-219 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)。
Content-Type	是	String	内容类型

表 4-220 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	策略名称（策略名称只能由数字、字母和下划线组成，长度不能超过64为字符）
log_action_replaced	否	Boolean	cc规则和精准防护规则“防护动作”选择“仅记录”时，Web基础防护是否命中策略规则并阻断，默认为true

响应参数

状态码： 200

表 4-221 响应 Body 参数

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名
level	Integer	Web基础防护等级 <ul style="list-style-type: none">1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none">false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String,Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	Array of strings	与防护策略绑定的防护的域名id数组
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String,String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-222 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none">log: 仅记录block: 拦截

表 4-223 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-224 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启
webshell	Boolean	Webshell检测是否开启
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启

参数	参数类型	描述
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启, 该特性是公测特性, 在公测期间, 只支持仅记录模式。

表 4-225 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式: cloud (云模式) / premium (独享模式)
mode	String	仅独享模式涉及特殊域名模式

状态码: 400

表 4-226 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-227 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-228** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-229 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 403**表 4-230** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-231 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-232 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-233 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中创建一个策略名称为demo的防护策略

```
POST https://{Endpoint}/v1/{project_id}/waf/policy?enterprise_project_id=0
{
  "name": "demo"
}
```

响应示例

状态码： 200

OK

```
{
  "id": "38ff0cb9a10e4d5293c642bc0350fa6d",
```

```
"name" : "demo",
"level" : 2,
"action" : {
  "category" : "log"
},
"options" : {
  "webattack" : true,
  "common" : true,
  "crawler" : true,
  "crawler_engine" : false,
  "crawler_scanner" : true,
  "crawler_script" : false,
  "crawler_other" : false,
  "webshell" : false,
  "cc" : true,
  "custom" : true,
  "whiteblackip" : true,
  "geoip" : true,
  "ignore" : true,
  "privacy" : true,
  "antitamper" : true,
  "antileakage" : false,
  "bot_enable" : true,
  "modulex_enabled" : false
},
"hosts" : [ ],
"extend" : { },
"timestamp" : 1650529538732,
"full_detection" : false,
"bind_host" : [ ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建一个策略名称为demo的防护策略

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

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

public class CreatePolicySolution {

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

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

```
        .withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
CreatePolicyRequest request = new CreatePolicyRequest();
CreatePolicyRequestBody body = new CreatePolicyRequestBody();
body.setName("demo");
request.withBody(body);
try {
    CreatePolicyResponse response = client.createPolicy(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id的项目中创建一个策略名称为demo的防护策略

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreatePolicyRequest()
        request.body = CreatePolicyRequestBody(
            name="demo"
        )
        response = client.create_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

在项目id为project_id的项目中创建一个策略名称为demo的防护策略

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreatePolicyRequest{}
    request.Body = &model.CreatePolicyRequestBody{
        Name: "demo",
    }
    response, err := client.CreatePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
403	资源配额不足

状态码	描述
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3.3 根据 Id 查询防护策略

功能介绍

根据Id查询防护策略

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}

表 4-234 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-235 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-236 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-237 响应 Body 参数

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名
level	Integer	Web基础防护等级 <ul style="list-style-type: none">1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none">false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息

参数	参数类型	描述
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String, Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	Array of strings	与防护策略绑定的防护的域名id数组
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String, String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-238 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none"> log: 仅记录 block: 拦截

表 4-239 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-240 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启

参数	参数类型	描述
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启
webshell	Boolean	Webshell检测是否开启
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启，该特性是公测特性，在公测期间，只支持仅记录模式。

表 4-241 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400

表 4-242 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-243 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-244 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-245 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-246 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-247 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id和防护策略id为policy_id的防护策略信息。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{  
  "id": "38ff0cb9a10e4d5293c642bc0350fa6d",  
  "name": "demo",  
  "level": 2,  
}
```

```
"action" : {
  "category" : "log"
},
"options" : {
  "webattack" : true,
  "common" : true,
  "crawler" : true,
  "crawler_engine" : false,
  "crawler_scanner" : true,
  "crawler_script" : false,
  "crawler_other" : false,
  "webshell" : false,
  "cc" : true,
  "custom" : true,
  "whiteblackip" : true,
  "geoip" : true,
  "ignore" : true,
  "privacy" : true,
  "antitamper" : true,
  "antileakage" : false,
  "bot_enable" : true,
  "modulex_enabled" : false
},
"hosts" : [ ],
"extend" : { },
"timestamp" : 1650529538732,
"full_detection" : false,
"bind_host" : [ ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowPolicySolution {

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

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

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



```
        .build();
        ShowPolicyRequest request = new ShowPolicyRequest();
        request.withPolicyId("{policy_id}");
        try {
            ShowPolicyResponse response = client.showPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowPolicyRequest()
        request.policy_id = "{policy_id}"
        response = client.show_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)
```

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowPolicyRequest{}
    request.PolicyId = "{policy_id}"
    response, err := client.ShowPolicy(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3.4 更新防护策略

功能介绍

更新防护策略，请求体可只传需要更新的部分

调用方法

请参见[如何调用API](#)。

URI

PATCH /v1/{project_id}/waf/policy/{policy_id}

表 4-248 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-249 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-250 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)。
Content-Type	是	String	内容类型

表 4-251 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	防护策略名

参数	是否必选	参数类型	描述
level	否	Integer	Web基础防护等级 <ul style="list-style-type: none"> 1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。 2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。 3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	否	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none"> false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截 true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	否	Action object	用于存放网站反爬虫中特征反爬虫动作信息
action	否	PolicyAction object	防护动作信息
options	否	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	否	Map<String,Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	否	Array of strings	与防护策略绑定的防护的域名id数组（该参数不支持修改，仅作为预留参数，用于后续功能扩展，可忽略）。
bind_host	否	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息（该参数不支持修改，仅作为预留参数，用于后续功能扩展，可忽略）。

参数	是否必选	参数类型	描述
extend	否	Map<String,String>	扩展字段，用于存放Web基础防护中一些开关配置等信息,详情请参考示例使用。 <ul style="list-style-type: none"> • deep_decode: 深度检测 • check_all_headers : header 全检测 • shiro_rememberMe_enable : Shiro解密检测

表 4-252 Action

参数	是否必选	参数类型	描述
category	否	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none"> • log: 仅记录 • block: 拦截

表 4-253 PolicyAction

参数	是否必选	参数类型	描述
category	否	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	否	String	攻击惩罚规则ID

表 4-254 PolicyOption

参数	是否必选	参数类型	描述
webattack	否	Boolean	基础防护是否开启
common	否	Boolean	常规检测是否开启
crawler	否	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	否	Boolean	搜索engine是否开启
crawler_scanner	否	Boolean	反爬虫检测是否开启
crawler_script	否	Boolean	脚本反爬虫是否开启

参数	是否必选	参数类型	描述
crawler_other	否	Boolean	其他爬虫是否开启
webshell	否	Boolean	Webshell检测是否开启
cc	否	Boolean	cc规则是否开启
custom	否	Boolean	精准防护是否开启
whiteblackip	否	Boolean	黑白名单防护是否开启
geoip	否	Boolean	地理位置访问控制规则是否开启
ignore	否	Boolean	误报屏蔽是否开启
privacy	否	Boolean	隐私屏蔽是否开启
antitamper	否	Boolean	网页防篡改规则是否开启
antileakage	否	Boolean	防敏感信息泄露规则是否开启
bot_enable	否	Boolean	网站反爬虫总开关是否开启
modulex_enabled	否	Boolean	modulex智能cc防护是否开启, 该特性是公测特性, 在公测期间, 只支持仅记录模式。

表 4-255 BindHost

参数	是否必选	参数类型	描述
id	否	String	域名ID
hostname	否	String	域名
waf_type	否	String	域名对应模式: cloud (云模式) / premium (独享模式)
mode	否	String	仅独享模式涉及特殊域名模式

响应参数

状态码: 200

表 4-256 响应 Body 参数

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名

参数	参数类型	描述
level	Integer	<p>Web基础防护等级</p> <ul style="list-style-type: none"> 1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。 2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。 3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	<p>精准防护中的检测模式。</p> <ul style="list-style-type: none"> false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截 true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String, Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	Array of strings	与防护策略绑定的防护的域名id数组
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String, String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-257 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none">log: 仅记录block: 拦截

表 4-258 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-259 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启
webshell	Boolean	Webshell检测是否开启
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启

参数	参数类型	描述
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启，该特性是公测特性，在公测期间，只支持仅记录模式。

表 4-260 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码：400

表 4-261 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-262 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-263 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-264 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-265 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-266 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

- 修改项目id为project_id和防护策略id为policy_id的防护策略中的基础防护开关配置等信息，打开深度检测、header全检测、Shiro解密检测的防护开关。

```
PATCH https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}?enterprise_project_id=0
```

```
{
  "extend" : {
    "extend" : "{\"deep_decode\":true,\"check_all_headers\":true,\"shiro_rememberMe_enable\":true}"
  }
}
```

- 关闭项目id为project_id和防护策略id为policy_id的黑白名单防护

```
PATCH https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}?enterprise_project_id=0
```

```
{
  "options" : {
    "whiteblackip" : false
  }
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name" : "demo",
  "level" : 2,
  "action" : {
    "category" : "log"
  },
  "options" : {
    "webattack" : true,
    "common" : true,
    "crawler" : true,
    "crawler_engine" : false,
    "crawler_scanner" : true,
    "crawler_script" : false,
    "crawler_other" : false,
    "webshell" : false,
    "cc" : true,
    "custom" : true,
    "whiteblackip" : false,
    "geoip" : true,
    "ignore" : true,
    "privacy" : true,
    "antitamper" : true,
    "antileakage" : false,
    "bot_enable" : true
  },
  "hosts" : [ "c0268b883a854adc8a2cd352193b0e13" ],
  "timestamp" : 1650529538732,
}
```

```
"full_detection" : false,
"bind_host" : [ {
  "id" : "c0268b883a854adc8a2cd352193b0e13",
  "hostname" : "www.demo.com",
  "waf_type" : "cloud"
} ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 修改项目id为project_id和防护策略id为policy_id的防护策略中的基础防护开关配置等信息，打开深度检测、header全检测、Shiro解密检测的防护开关。

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

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

import java.util.Map;
import java.util.HashMap;

public class UpdatePolicySolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePolicyRequest request = new UpdatePolicyRequest();
        request.withPolicyId("{policy_id}");
        UpdatePolicyRequestBody body = new UpdatePolicyRequestBody();
        Map<String, String> listbodyExtend = new HashMap<>();
        listbodyExtend.put("extend",
            "{\"deep_decode\":true,\"check_all_headers\":true,\"shiro_rememberMe_enable\":true}");
        body.withExtend(listbodyExtend);
        request.withBody(body);
        try {
            UpdatePolicyResponse response = client.updatePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

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

- 关闭项目id为project_id和防护策略id为policy_id的黑白名单防护

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

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

public class UpdatePolicySolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePolicyRequest request = new UpdatePolicyRequest();
        request.withPolicyId("{policy_id}");
        UpdatePolicyRequestBody body = new UpdatePolicyRequestBody();
        PolicyOption optionsbody = new PolicyOption();
        optionsbody.withWhiteblackip(false);
        body.withOptions(optionsbody);
        request.withBody(body);
        try {
            UpdatePolicyResponse response = client.updatePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 修改项目id为project_id和防护策略id为policy_id的防护策略中的基础防护开关配置等信息，打开深度检测、header全检测、Shiro解密检测的防护开关。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePolicyRequest()
        request.policy_id = "{policy_id}"
        listExtendbody = {
            "extend": "{\"deep_decode\":true,\"check_all_headers\":true,\"shiro_rememberMe_enable\":true}"
        }
        request.body = UpdatePolicyRequestBody(
            extend=listExtendbody
        )
        response = client.update_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)
```

- 关闭项目id为project_id和防护策略id为policy_id的黑白名单防护

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = UpdatePolicyRequest()
    request.policy_id = "{policy_id}"
    optionsbody = PolicyOption(
        whiteblackip=False
    )
    request.body = UpdatePolicyRequestBody(
        options=optionsbody
    )
    response = client.update_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

- 修改项目id为project_id和防护策略id为policy_id的防护策略中的基础防护开关配置等信息，打开深度检测、header全检测、Shiro解密检测的防护开关。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePolicyRequest{}
    request.PolicyId = "{policy_id}"
    var listExtendbody = map[string]string{
        "extend": "{\"deep_decode\":true,\"check_all_headers\":true,\"shiro_rememberMe_enable\":true}",
    }
    request.Body = &model.UpdatePolicyRequestBody{
        Extend: listExtendbody,
    }
    response, err := client.UpdatePolicy(request)
```

```
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 关闭项目id为project_id和防护策略id为policy_id的黑白名单防护

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePolicyRequest{}
    request.PolicyId = "{policy_id}"
    whiteblackipOptions := false
    optionsbody := &model.PolicyOption{
        Whiteblackip: &whiteblackipOptions,
    }
    request.Body = &model.UpdatePolicyRequestBody{
        Options: optionsbody,
    }
    response, err := client.UpdatePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3.5 删除防护策略

功能介绍

删除防护策略，若策略正在使用，则需要先解除域名与策略的绑定关系才能删除策略。

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}

表 4-267 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-268 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-269 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-270 响应 Body 参数

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名
level	Integer	Web基础防护等级 <ul style="list-style-type: none">1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none">false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息

参数	参数类型	描述
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String, Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性
hosts	Array of strings	与防护策略绑定的防护的域名id数组
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String, String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-271 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none">• log: 仅记录• block: 拦截

表 4-272 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-273 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启

参数	参数类型	描述
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启
webshell	Boolean	Webshell检测是否开启
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启，该特性是公测特性，在公测期间，只支持仅记录模式。

表 4-274 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400

表 4-275 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-276 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-277 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-278 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-279 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-280 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中删除防护策略id为policy_id的防护策略。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{  
  "id": "62169e2fc4e64148b775ec01b24a1947",  
  "name": "demo",  
  "level": 2,  
}
```

```
"action" : {
  "category" : "log",
  "modulex_category" : "log"
},
"options" : {
  "webattack" : true,
  "common" : true,
  "crawler" : true,
  "crawler_engine" : false,
  "crawler_scanner" : true,
  "crawler_script" : false,
  "crawler_other" : false,
  "webshell" : false,
  "cc" : true,
  "custom" : true,
  "precise" : false,
  "whiteblackip" : true,
  "geoip" : true,
  "ignore" : true,
  "privacy" : true,
  "antitamper" : true,
  "anticrawler" : false,
  "antileakage" : false,
  "followed_action" : false,
  "bot_enable" : true,
  "modulex_enabled" : false
},
"hosts" : [ ],
"extend" : { },
"timestamp" : 1649316510603,
"full_detection" : false,
"bind_host" : [ ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeletePolicySolution {

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

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

```
WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
DeletePolicyRequest request = new DeletePolicyRequest();
request.withPolicyId("{policy_id}");
try {
    DeletePolicyResponse response = client.deletePolicy(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeletePolicyRequest()
        request.policy_id = "{policy_id}"
        response = client.delete_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"
```



```
waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.3.6 更新防护策略的域名

功能介绍

更新防护策略的防护域名

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}

表 4-281 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-282 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
hosts	是	String	域名id，您可以通过调用查询云模式防护域名列表（ListHost）获取域名id

请求参数

表 4-283 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-284 响应 Body 参数

参数	参数类型	描述
id	String	防护策略id
name	String	防护策略名
level	Integer	Web基础防护等级 <ul style="list-style-type: none"> 1：宽松，防护粒度较粗，只拦截攻击特征比较明显的请求。当误报情况较多的场景下，建议选择“宽松”模式。 2：中等，默认为“中等”防护模式，满足大多数场景下的Web防护需求。 3：严格，防护粒度最精细，可以拦截具有复杂的绕过特征的攻击请求，例如jolokia网络攻击、探测CGI漏洞、探测 Druid SQL注入攻击
full_detection	Boolean	精准防护中的检测模式。 <ul style="list-style-type: none"> false：短路检测，当用户的请求符合精准防护中的拦截条件时，便立刻终止检测，进行拦截 true：全检测，请求符合精准防护中的拦截条件时，全检测不会立即拦截，会继续执行其他防护的检测，最后进行拦截。
robot_action	Action object	用于存放网站反爬虫中特征反爬虫动作信息
action	PolicyAction object	防护动作信息
options	PolicyOption object	该参数存放防护策略里防护项的开关信息
modulex_options	Map<String, Object>	智能访问控制防护项相关配置信息，目前该特性还处于公测阶段，只有部分局点支持该特性

参数	参数类型	描述
hosts	Array of strings	与防护策略绑定的防护的域名id数组
bind_host	Array of BindHost objects	与防护策略绑定的防护的域名信息数组，相对于hosts字段，包含更详细的域名信息
extend	Map<String,String>	扩展字段，用于存放Web基础防护中一些开关配置等信息
timestamp	Long	创建防护策略的时间

表 4-285 Action

参数	参数类型	描述
category	String	特征反爬虫中防护动作信息 <ul style="list-style-type: none">log: 仅记录block: 拦截

表 4-286 PolicyAction

参数	参数类型	描述
category	String	web基础防护动作（log为仅记录、block为拦截）
followed_action_id	String	攻击惩罚规则ID

表 4-287 PolicyOption

参数	参数类型	描述
webattack	Boolean	基础防护是否开启
common	Boolean	常规检测是否开启
crawler	Boolean	预留参数，该参数值一直为true，用户可忽略该参数值
crawler_engine	Boolean	搜索engine是否开启
crawler_scanner	Boolean	反爬虫检测是否开启
crawler_script	Boolean	脚本反爬虫是否开启
crawler_other	Boolean	其他爬虫是否开启

参数	参数类型	描述
webshell	Boolean	Webshell检测是否开启
cc	Boolean	cc规则是否开启
custom	Boolean	精准防护是否开启
whiteblackip	Boolean	黑白名单防护是否开启
geoip	Boolean	地理位置访问控制规则是否开启
ignore	Boolean	误报屏蔽是否开启
privacy	Boolean	隐私屏蔽是否开启
antitamper	Boolean	网页防篡改规则是否开启
antileakage	Boolean	防敏感信息泄露规则是否开启
bot_enable	Boolean	网站反爬虫总开关是否开启
modulex_enabled	Boolean	modulex智能cc防护是否开启, 该特性是公测特性, 在公测期间, 只支持仅记录模式。

表 4-288 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式: cloud (云模式) / premium (独享模式)
mode	String	仅独享模式涉及特殊域名模式

状态码: 400

表 4-289 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-290 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-291 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-292 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-293 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-294 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id和防护策略id为policy_id的防护域名id为c0268b883a854adc8a2cd352193b0e13。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}?
enterprise_project_id=0&hosts=c0268b883a854adc8a2cd352193b0e13
```

响应示例

状态码: 200

OK

```
{
  "id": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name": "demo",
  "level": 2,
  "action": {
    "category": "log"
  },
  "options": {
    "webattack": true,
    "common": true,
    "crawler": true,
    "crawler_engine": false,
    "crawler_scanner": true,
    "crawler_script": false,
    "crawler_other": false,
    "webshell": false,
    "cc": true,
  }
}
```

```
"custom" : true,
"whiteblackip" : true,
"geoip" : true,
"ignore" : true,
"privacy" : true,
"antitamper" : true,
"antileakage" : false,
"bot_enable" : true,
"modulex_enabled" : false
},
"hosts" : [ "c0268b883a854adc8a2cd352193b0e13" ],
"extend" : { },
"timestamp" : 1650529538732,
"full_detection" : false,
"bind_host" : [ {
  "id" : "c0268b883a854adc8a2cd352193b0e13",
  "hostname" : "www.demo.com",
  "waf_type" : "cloud"
} ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class UpdatePolicyProtectHostSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePolicyProtectHostRequest request = new UpdatePolicyProtectHostRequest();
        request.withPolicyId("{policy_id}");
        try {
            UpdatePolicyProtectHostResponse response = client.updatePolicyProtectHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```



```
        e.printStackTrace();
    } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePolicyProtectHostRequest()
        request.policy_id = "{policy_id}"
        response = client.update_policy_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

```
package main

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

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

```
projectId := "{project_id}"

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4 策略规则管理

4.4.1 修改单条规则的状态

功能介绍

修改单条规则的状态，用于开启或者关闭单条规则，比如关闭精准防护中某一条规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/{ruletype}/{rule_id}/status

表 4-295 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	策略id (策略id从查询防护策略列表接口获取)
ruletype	是	String	策略类型
rule_id	是	String	规则id, 根据不同的规则类型 (ruletype) 调用规则列表接口获取规则id, 例如黑白名单 (whiteblackip) 规则id, 您可以通过调用查询黑白名单规则列表 (ListWhiteblackipRule) 获取规则id

表 4-296 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-297 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)。
Content-Type	是	String	内容类型

表 4-298 请求 Body 参数

参数	是否必选	参数类型	描述
status	否	Integer	状态（开启：1，关闭：0）

响应参数

状态码：200

表 4-299 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	规则创建时间
description	String	规则描述
status	Integer	状态（开启：1，关闭：0）

状态码：400

表 4-300 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-301 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码

参数	参数类型	描述
error_msg	String	下游服务报错信息

状态码： 401

表 4-302 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-303 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-304 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-305 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id，防护策略id为policy_id，策略类型为ruletype，规则id为rule_id的防护状态，将防护状态修改为关闭。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/{ruletype}/{rule_id}/status?enterprise_project_id=0
```

```
{
  "status" : 0
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "709bfd0d62a9410394ffa9e25eb82c36",
  "policyid" : "62fd7f8c36234a4ebedabc2ce451ed45",
  "timestamp" : 1650362797070,
  "description" : "demo",
  "status" : 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改项目id为project_id，防护策略id为policy_id，策略类型为ruletype，规则id为rule_id的防护状态，将防护状态修改为关闭。

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

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

```
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class UpdatePolicyRuleStatusSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePolicyRuleStatusRequest request = new UpdatePolicyRuleStatusRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleType(UpdatePolicyRuleStatusRequest.RuleTypeEnum.fromValue("{ruletype}"));
        request.withRuleId("{rule_id}");
        UpdatePolicyRuleStatusRequestBody body = new UpdatePolicyRuleStatusRequestBody();
        body.withStatus(0);
        request.withBody(body);
        try {
            UpdatePolicyRuleStatusResponse response = client.updatePolicyRuleStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改项目id为project_id，防护策略id为policy_id，策略类型为ruletype，规则id为rule_id的防护状态，将防护状态修改为关闭。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = UpdatePolicyRuleStatusRequest()
    request.policy_id = "{policy_id}"
    request.ruletype = "{ruletype}"
    request.rule_id = "{rule_id}"
    request.body = UpdatePolicyRuleStatusRequestBody(
        status=0
    )
    response = client.update_policy_rule_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

修改项目id为project_id，防护策略id为policy_id，策略类型为ruletype，规则id为rule_id的防护状态，将防护状态修改为关闭。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePolicyRuleStatusRequest{}
    request.PolicyId = "{policy_id}"
    request.Ruletype = model.GetUpdatePolicyRuleStatusRequestRuleTypeEnum().RULETYPE
    request.RuleId = "{rule_id}"
    statusUpdatePolicyRuleStatusRequestBody := int32(0)
```



```
request.Body = &model.UpdatePolicyRuleStatusRequestBody{
    Status: &statusUpdatePolicyRuleStatusRequestBody,
}
response, err := client.UpdatePolicyRuleStatus(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.2 查询 cc 规则列表

功能介绍

查询cc规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/cc

表 4-306 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

参数	是否必选	参数类型	描述
policy_id	是	String	策略id (策略id从查询防护策略列表接口获取)

表 4-307 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
offset	是	Integer	偏移量, 表示查询该偏移量之后的记录。
limit	是	Integer	查询返回记录的数量限制。

请求参数

表 4-308 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-309 响应 Body 参数

参数	参数类型	描述
total	Integer	Number of rules in the policy
items	Array of CcrulesListInfo objects	Array of Cc rules

表 4-310 CcrulesListInfo

参数	参数类型	描述
name	String	规则名称
id	String	Rule ID.
policyid	String	Policy ID.
url	String	当mode值为0时, 该参数有返回值。规则应用的URL链接, 不包含域名。
prefix	Boolean	路径是否为前缀模式, 当防护url以*结束, 则为前缀模式。当mode值为0时, 该参数有返回值。
mode	Integer	cc规则防护模式, 对应console上的mode, 现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0: 标准, 只支持对域名的防护路径做限制。1: 高级, 支持对路径、IP、Cookie、Header、Params字段做限制。
status	Integer	规则状态, 0: 关闭, 1: 开启
conditions	Array of CcCondition objects	cc规则防护规则限速条件, 当cc防护规则为高级模式 (mode参数值为1) 时, 该参数必填。
action	action object	请求次数限制到达后采取的防护动作
tag_type	String	限速模式: <ul style="list-style-type: none">ip: IP限速, 根据IP区分单个Web访问者。cookie: 用户限速, 根据Cookie键值区分单个Web访问者。header: 用户限速, 根据Header区分单个Web访问者。other: 根据Referer (自定义请求访问的来源) 字段区分单个Web访问者。policy: 策略限速domain: 域名限速url: url限速

参数	参数类型	描述
tag_index	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者
limit_num	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	Integer	限速周期，单位为秒，范围1~3600
unlock_num	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	Boolean	是否开启域名聚合统计。
region_aggregation	Boolean	是否开启全局计数。
description	String	规则描述
total_num	Integer	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
unaggregation	Boolean	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
aging_time	Integer	规则老化时间，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

参数	参数类型	描述
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
timestamp	Long	创建规则时间戳

表 4-311 CcCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6，匹配逻辑可以为：equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	Array of strings	条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时，需要传该参数。

参数	参数类型	描述
value_list_id	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。
index	String	子字段，当字段类型（category）选择“params”、“cookie”、“header”时，请根据实际需求配置子字段且该参数必填。

表 4-312 action

参数	参数类型	描述
category	String	动作类型： <ul style="list-style-type: none">• captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。• -block: 阻断。• log: 仅记录• dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。 注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	detail object	阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。 <ul style="list-style-type: none">• 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。• 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-313 detail

参数	参数类型	描述
response	response object	阻断页面

表 4-314 response

参数	参数类型	描述
content_type	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	String	阻断页面内容

表 4-315 tag_condition

参数	参数类型	描述
category	String	用户标识字段，其值固定为referer
contents	Array of strings	用户标识字段内容

状态码：400

表 4-316 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-317 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码：401

表 4-318 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-319 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-320 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-321 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的cc规则列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/cc?offset=0&limit=1
```

响应示例

状态码： 200

Request succeeded.

```
{
  "total": 1,
  "items": [ {
    "id": "f88c5eabff9b4ff9ba6e7dd8e38128ba",
    "policyid": "d471eef691684f1c8d7784532fd8f4bd",
    "timestamp": 1678873040603,
    "name": "test",
    "description": "",
    "status": 1,
    "mode": 1,
    "conditions": [ {
      "category": "url",
      "contents": [ "/url" ],
      "logic_operation": "contain"
    } ],
    "action": {
      "category": "captcha"
    },
    "producer": 1,
    "unaggregation": false,
    "total_num": 0,
    "limit_num": 10,
    "limit_period": 60,
    "lock_time": 0,
    "tag_type": "ip",
    "aging_time": 0,
    "region_aggregation": false,
    "domain_aggregation": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListCcRulesSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListCcRulesRequest request = new ListCcRulesRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListCcRulesResponse response = client.listCcRules(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = ListCcRulesRequest()
    request.policy_id = "{policy_id}"
    response = client.list_cc_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	Request succeeded.
400	Request failed.
401	The token does not have required permissions.
500	Internal server error.

错误码

请参见[错误码](#)。

4.4.3 创建 cc 规则

功能介绍

创建cc规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/cc

表 4-322 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	策略id（策略id从查询防护策略列表接口获取）

表 4-323 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-324 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)。
Content-Type	是	String	内容类型

表 4-325 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	规则名称
mode	是	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0：标准，只支持对域名的防护路径做限制。1：高级，支持对路径、IP、Cookie、Header、Params字段做限制。
conditions	是	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	是	action object	请求次数限制到达后采取的防护动作

参数	是否必选	参数类型	描述
tag_type	是	String	限速模式： <ul style="list-style-type: none">ip: IP限速，根据IP区分单个Web访问者。cookie: 用户限速，根据Cookie键值区分单个Web访问者。header: 用户限速，根据Header区分单个Web访问者。other: 根据Referer（自定义请求访问的来源）字段区分单个Web访问者。policy: 策略限速domain: 域名限速url: url限速
tag_index	否	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	否	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者
limit_num	是	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	是	Integer	限速周期，单位为秒，范围1~3600

参数	是否必选	参数类型	描述
unlock_num	否	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	否	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	否	Boolean	是否开启域名聚合统计。
region_aggregation	否	Boolean	是否开启全局计数。
description	否	String	规则描述

表 4-326 CcCondition

参数	是否必选	参数类型	描述
category	是	String	字段类型

参数	是否必选	参数类型	描述
logic_operation	是	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6, 匹配逻辑可以为: equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	否	Array of strings	条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时, 需要传该参数。

参数	是否必选	参数类型	描述
value_list_id	否	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。
index	否	String	子字段，当字段类型（category）选择“params”、“cookie”、“header”时，请根据实际需求配置子字段且该参数必填。

表 4-327 action

参数	是否必选	参数类型	描述
category	是	String	<p>动作类型：</p> <ul style="list-style-type: none"> • captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。 • -block: 阻断。 • log: 仅记录 • dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	否	detail object	<p>阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。</p> <ul style="list-style-type: none"> • 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。 • 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-328 detail

参数	是否必选	参数类型	描述
response	否	response object	返回页面

表 4-329 response

参数	是否必选	参数类型	描述
content_type	否	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	否	String	防护页面内容

表 4-330 tag_condition

参数	是否必选	参数类型	描述
category	否	String	用户标识字段，其值固定为 referer
contents	否	Array of strings	用户标识字段内容

响应参数

状态码： 200

表 4-331 响应 Body 参数

参数	参数类型	描述
name	String	规则名称
id	String	Rule ID.
policyid	String	Policy ID.
url	String	当mode值为0时，该参数有返回值。规则应用的URL链接。
prefix	Boolean	路径是否为前缀模式，当防护url以*结束，则为前缀模式。

参数	参数类型	描述
mode	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0：标准，只支持对域名的防护路径做限制。1：高级，支持对路径、IP、Cookie、Header、Params字段做限制。
status	Integer	规则状态，0：关闭，1：开启
conditions	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	action object	请求次数限制到达后采取的防护动作
tag_type	String	限速模式： <ul style="list-style-type: none">ip：IP限速，根据IP区分单个Web访问者。cookie：用户限速，根据Cookie键值区分单个Web访问者。header：用户限速，根据Header区分单个Web访问者。other：根据Referer（自定义请求访问的来源）字段区分单个Web访问者。policy：策略限速domain：域名限速url：url限速
tag_index	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者

参数	参数类型	描述
limit_num	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	Integer	限速周期，单位为秒，范围1~3600
unlock_num	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	Boolean	是否开启域名聚合统计。
region_aggregation	Boolean	是否开启全局计数。
description	String	规则描述
total_num	Integer	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
unaggregation	Boolean	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
aging_time	Integer	规则老化时间，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
timestamp	Long	创建规则时间戳

表 4-332 CcCondition

参数	参数类型	描述
category	String	字段类型

参数	参数类型	描述
logic_operation	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6, 匹配逻辑可以为: equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	Array of strings	<p>条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时, 需要传该参数。</p>
value_list_id	String	<p>引用表id。当logic_operation参数以any或者all结尾时, 需要传该参数。此外, 引用表类型要与category类型保持一致。</p>
index	String	<p>子字段, 当字段类型 (category) 选择“params”、“cookie”、“header”时, 请根据实际需求配置子字段且该参数必填。</p>

表 4-333 action

参数	参数类型	描述
category	String	动作类型： <ul style="list-style-type: none">captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。-block: 阻断。log: 仅记录dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。 注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	detail object	阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。 <ul style="list-style-type: none">如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-334 detail

参数	参数类型	描述
response	response object	阻断页面

表 4-335 response

参数	参数类型	描述
content_type	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	String	阻断页面内容

表 4-336 tag_condition

参数	参数类型	描述
category	String	用户标识字段，其值固定为referer

参数	参数类型	描述
contents	Array of strings	用户标识字段内容

状态码： 400

表 4-337 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-338 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-339 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-340 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-341 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-342 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id和防护策略id为policy_id的防护策略下创建cc规则，规则的名称test55,限速模式为ip限速，限制频率为10，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，cc规则防护规则限速条件的字段类型是URL包含“/url”，没有子字段，不开启全局计数


```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/cc?
```

```
{
  "description": "",
  "name": "test55",
  "tag_type": "ip",
  "limit_num": 10,
  "limit_period": 60,
  "action": {
    "category": "captcha"
  },
  "mode": 1,
  "domain_aggregation": false,
  "conditions": [ {
    "category": "url",
    "logic_operation": "contain",
    "contents": [ "/url" ],
    "index": null
  } ],
  "region_aggregation": false
}
```

响应示例

状态码： 200

Request succeeded.

```
{
  "id": "f88c5eabff9b4ff9ba6e7dd8e38128ba",
  "policyid": "d471eef691684f1c8d7784532fd8f4bd",
  "name": "test55",
  "timestamp": 1678873040603,
  "description": "",
  "status": 1,
  "mode": 1,
  "conditions": [ {
    "category": "url",
    "contents": [ "/url" ],
    "logic_operation": "contain"
  } ],
  "action": {
    "category": "captcha"
  },
  "producer": 1,
  "unaggregation": false,
  "total_num": 0,
  "limit_num": 10,
  "limit_period": 60,
  "lock_time": 0,
  "tag_type": "ip",
  "aging_time": 0,
  "region_aggregation": false,
  "domain_aggregation": false
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id和防护策略id为policy_id的防护策略下创建cc规则，规则的名称test55,限速模式为ip限速，限制频率为10，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，cc规则防护规则限速条件的字段类型是URL包含“/url”，没有子字段，不开启全局计数

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

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

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

public class CreateCcRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateCcRuleRequest request = new CreateCcRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateCcRuleRequestBody body = new CreateCcRuleRequestBody();
        CreateCcRuleRequestBodyAction actionbody = new CreateCcRuleRequestBodyAction();
        actionbody.withCategory(CreateCcRuleRequestBodyAction.CategoryEnum.fromValue("captcha"));
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("/url");
        List<CcCondition> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new CcCondition()
                .withCategory(CcCondition.CategoryEnum.fromValue("url"))
                .withLogicOperation("contain")
                .withContents(listConditionsContents)
        );
        body.withDescription("");
        body.withRegionAggregation(false);
        body.withDomainAggregation(false);
        body.withLimitPeriod(60);
        body.withLimitNum(10);
        body.withTagType(CreateCcRuleRequestBody.TagTypeEnum.fromValue("ip"));
        body.withAction(actionbody);
        body.withConditions(listbodyConditions);
        body.withMode(1);
        body.withName("test55");
        request.withBody(body);
        try {
            CreateCcRuleResponse response = client.createCcRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

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

Python

在项目id为project_id和防护策略id为policy_id的防护策略下创建cc规则，规则的名称test55,限速模式为ip限速，限制频率为10，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，cc规则防护规则限速条件的字段类型是URL包含“/url”，没有子字段，不开启全局计数

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateCcRuleRequest()
        request.policy_id = "{policy_id}"
        actionbody = CreateCcRuleRequestBodyAction(
            category="captcha"
        )
        listContentsConditions = [
            "/url"
        ]
        listConditionsbody = [
            CcCondition(
                category="url",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
        request.body = CreateCcRuleRequestBody(
            description="",
            region_aggregation=False,
            domain_aggregation=False,
            limit_period=60,
            limit_num=10,
            tag_type="ip",
            action=actionbody,
            conditions=listConditionsbody,
            mode=1,
            name="test55"
        )
    
```

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

Go

在项目id为project_id和防护策略id为policy_id的防护策略下创建cc规则，规则的名称test55,限速模式为ip限速，限制频率为10，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，cc规则防护规则限速条件的字段类型是URL包含“/url”，没有子字段，不开启全局计数

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateCcRuleRequest{}
    request.PolicyId = "{policy_id}"
    actionbody := &model.CreateCcRuleRequestBodyAction{
        Category: model.GetCreateCcRuleRequestBodyActionCategoryEnum().CAPTCHA,
    }
    var listContentsConditions = []string{
        "/url",
    }
    var listConditionsbody = []model.CcCondition{
        {
            Category: model.GetCcConditionCategoryEnum().URL,
            LogicOperation: "contain",
            Contents: &listContentsConditions,
        },
    }
    descriptionCreateCcRuleRequestBody := ""
    regionAggregationCreateCcRuleRequestBody := false
    domainAggregationCreateCcRuleRequestBody := false
    nameCreateCcRuleRequestBody := "test55"
    request.Body = &model.CreateCcRuleRequestBody{
        Description: &descriptionCreateCcRuleRequestBody,
```

```
RegionAggregation: &regionAggregationCreateCcRuleRequestBody,  
DomainAggregation: &domainAggregationCreateCcRuleRequestBody,  
LimitPeriod: int32(60),  
LimitNum: int32(10),  
TagType: model.GetCreateCcRuleRequestBodyTagTypeEnum().IP,  
Action: actionbody,  
Conditions: listConditionsbody,  
Mode: int32(1),  
Name: &nameCreateCcRuleRequestBody,  
}  
response, err := client.CreateCcRule(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

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

状态码

状态码	描述
200	Request succeeded.
400	Request failed.
401	The token does not have required permissions.
500	Internal server error.

错误码

请参见[错误码](#)。

4.4.4 根据 Id 查询 cc 防护规则

功能介绍

根据Id查询cc防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}

表 4-343 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	策略id (策略id从查询防护策略列表接口获取)
rule_id	是	String	cc规则的ID。可以通过调用 ListCcRules API获取。

表 4-344 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-345 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-346 响应 Body 参数

参数	参数类型	描述
name	String	规则名称
id	String	Rule ID.
policyid	String	Policy ID.

参数	参数类型	描述
url	String	当mode值为0时，该参数有返回值。规则应用的URL链接，不包含域名。
prefix	Boolean	路径是否为前缀模式，当防护url以*结束，则为前缀模式。
mode	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0：标准，只支持对域名的防护路径做限制。1：高级，支持对路径、IP、Cookie、Header、Params字段做限制。
status	Integer	规则状态，0：关闭，1：开启
conditions	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	action object	请求次数限制到达后采取的防护动作
tag_type	String	限速模式： <ul style="list-style-type: none">ip：IP限速，根据IP区分单个Web访问者。cookie：用户限速，根据Cookie键值区分单个Web访问者。header：用户限速，根据Header区分单个Web访问者。other：根据Referer（自定义请求访问的来源）字段区分单个Web访问者。policy：策略限速domain：域名限速url：url限速

参数	参数类型	描述
tag_index	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者
limit_num	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	Integer	限速周期，单位为秒，范围1~3600
unlock_num	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	Boolean	是否开启域名聚合统计。
region_aggregation	Boolean	是否开启全局计数。
description	String	规则描述
total_num	Integer	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
unaggregation	Boolean	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
aging_time	Integer	规则老化时间，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

参数	参数类型	描述
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
timestamp	Long	创建规则时间戳

表 4-347 CcCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6，匹配逻辑可以为：equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	Array of strings	条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时，需要传该参数。

参数	参数类型	描述
value_list_id	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。
index	String	子字段，当字段类型（category）选择“params”、“cookie”、“header”时，请根据实际需求配置子字段且该参数必填。

表 4-348 action

参数	参数类型	描述
category	String	动作类型： <ul style="list-style-type: none">• captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。• -block: 阻断。• log: 仅记录• dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。 注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	detail object	阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。 <ul style="list-style-type: none">• 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。• 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-349 detail

参数	参数类型	描述
response	response object	阻断页面

表 4-350 response

参数	参数类型	描述
content_type	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	String	阻断页面内容

表 4-351 tag_condition

参数	参数类型	描述
category	String	用户标识字段，其值固定为referer
contents	Array of strings	用户标识字段内容

状态码：400

表 4-352 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-353 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码：401

表 4-354 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-355 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-356 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-357 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id，规则id为rule_id的cc防护规则

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}?
```

响应示例

状态码： 200

Request succeeded.

```
{
  "id": "f88c5eabff9b4ff9ba6e7dd8e38128ba",
  "policyid": "d471eef691684f1c8d7784532fd8f4bd",
  "name": "test55",
  "timestamp": 1678873040603,
  "description": "",
  "status": 1,
  "mode": 1,
  "conditions": [ {
    "category": "url",
    "contents": [ "/url" ],
    "logic_operation": "contain"
  } ],
  "action": {
    "category": "captcha"
  },
  "producer": 1,
  "unaggregation": false,
  "total_num": 0,
  "limit_num": 10,
  "limit_period": 60,
  "lock_time": 0,
  "tag_type": "ip",
  "aging_time": 0,
  "region_aggregation": false,
  "domain_aggregation": 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.waf.v1.region.WafRegion;
```

```
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowCcRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowCcRuleRequest request = new ShowCcRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowCcRuleResponse response = client.showCcRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(WafRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = ShowCcRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.show_cc_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowCcRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowCcRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	Request succeeded.
400	Request failed.
401	The token does not have required permissions.
500	Internal server error.

错误码

请参见[错误码](#)。

4.4.5 更新 cc 防护规则

功能介绍

更新cc防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}

表 4-358 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	策略id（策略id从查询防护策略列表接口获取）
rule_id	是	String	cc规则的ID。可以通过调用ListCcRules API获取。

表 4-359 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-360 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)。
Content-Type	是	String	内容类型

表 4-361 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	规则名称
mode	是	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0: 标准，只支持对域名的防护路径做限制。1: 高级，支持对路径、IP、Cookie、Header、Params字段做限制。
url	否	String	需要防护的域名路径，当cc防护规则为标准模式（mode参数值为0）时，该参数必填。
conditions	是	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	是	action object	请求次数限制到达后采取的防护动作

参数	是否必选	参数类型	描述
tag_type	是	String	<p>限速模式：</p> <ul style="list-style-type: none"> ip: IP限速，根据IP区分单个Web访问者。 cookie: 用户限速，根据Cookie键值区分单个Web访问者。 header: 用户限速，根据Header区分单个Web访问者。 other: 根据Referer（自定义请求访问的来源）字段区分单个Web访问者。 policy: 策略限速 domain: 域名限速 url: url限速
tag_index	否	String	<p>用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。</p> <ul style="list-style-type: none"> 选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。 选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	否	tag_condition object	<p>用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者</p>
limit_num	是	Integer	<p>限制频率，单位为次，范围为1~2147483647</p>
limit_period	是	Integer	<p>限速周期，单位为秒，范围1~3600</p>

参数	是否必选	参数类型	描述
unlock_num	否	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	否	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	否	Boolean	是否开启域名聚合统计。
region_aggregation	否	Boolean	是否开启全局计数。
description	否	String	规则描述

表 4-362 CcCondition

参数	是否必选	参数类型	描述
category	是	String	字段类型

参数	是否必选	参数类型	描述
logic_operation	是	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6, 匹配逻辑可以为: equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	否	Array of strings	条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时, 需要传该参数。

参数	是否必选	参数类型	描述
value_list_id	否	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。
index	否	String	子字段，当字段类型（category）选择“params”、“cookie”、“header”时，请根据实际需求配置子字段且该参数必填。

表 4-363 action

参数	是否必选	参数类型	描述
category	是	String	动作类型： <ul style="list-style-type: none">• captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。• -block: 阻断。• log: 仅记录• dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	否	detail object	阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。 <ul style="list-style-type: none">• 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。• 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-364 detail

参数	是否必选	参数类型	描述
response	否	response object	返回页面

表 4-365 response

参数	是否必选	参数类型	描述
content_type	否	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	否	String	防护页面内容

表 4-366 tag_condition

参数	是否必选	参数类型	描述
category	否	String	用户标识字段，其值固定为 referer
contents	否	Array of strings	用户标识字段内容

响应参数

状态码： 200

表 4-367 响应 Body 参数

参数	参数类型	描述
name	String	规则名称
id	String	Rule ID.
policyid	String	Policy ID.
url	String	当mode值为0时，该参数有返回值。规则应用的URL链接，不包含域名。
prefix	Boolean	路径是否为前缀模式，当防护url以*结束，则为前缀模式。

参数	参数类型	描述
mode	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0：标准，只支持对域名的防护路径做限制。1：高级，支持对路径、IP、Cookie、Header、Params字段做限制。
conditions	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	action object	请求次数限制到达后采取的防护动作
tag_type	String	限速模式： <ul style="list-style-type: none">ip：IP限速，根据IP区分单个Web访问者。cookie：用户限速，根据Cookie键值区分单个Web访问者。header：用户限速，根据Header区分单个Web访问者。other：根据Referer（自定义请求访问的来源）字段区分单个Web访问者。policy：策略限速domain：域名限速url：url限速
tag_index	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者

参数	参数类型	描述
limit_num	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	Integer	限速周期，单位为秒，范围1~3600
unlock_num	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	Boolean	是否开启域名聚合统计。
region_aggregation	Boolean	是否开启全局计数。
description	String	规则描述
total_num	Integer	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
unaggregation	Boolean	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
aging_time	Integer	规则老化时间，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

表 4-368 CcCondition

参数	参数类型	描述
category	String	字段类型

参数	参数类型	描述
logic_operation	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6, 匹配逻辑可以为: equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	Array of strings	<p>条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时, 需要传该参数。</p>
value_list_id	String	<p>引用表id。当logic_operation参数以any或者all结尾时, 需要传该参数。此外, 引用表类型要与category类型保持一致。</p>
index	String	<p>子字段, 当字段类型 (category) 选择“params”、“cookie”、“header”时, 请根据实际需求配置子字段且该参数必填。</p>

表 4-369 action

参数	参数类型	描述
category	String	<p>动作类型：</p> <ul style="list-style-type: none"> captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。 -block: 阻断。 log: 仅记录 dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。 注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	detail object	<p>阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。</p> <ul style="list-style-type: none"> 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-370 detail

参数	参数类型	描述
response	response object	阻断页面

表 4-371 response

参数	参数类型	描述
content_type	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	String	阻断页面内容

表 4-372 tag_condition

参数	参数类型	描述
category	String	用户标识字段，其值固定为referer

参数	参数类型	描述
contents	Array of strings	用户标识字段内容

状态码： 400

表 4-373 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-374 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-375 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-376 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-377 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-378 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id，防护策略id为policy_id，规则id为rule_id的cc规则，修改限速模式为ip限速，限速频率为10次，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，规则的名称为test55，不开启域名聚合统计，cc规则防护规则限速条件是URL包含“/url”，没有子字段，不开启全局计数。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}?
{
  "description": "",
  "tag_type": "ip",
  "limit_num": 10,
  "limit_period": 60,
  "action": {
    "category": "captcha"
  },
  "mode": 1,
  "name": "test55",
  "domain_aggregation": false,
  "conditions": [ {
    "category": "url",
    "logic_operation": "contain",
    "contents": [ "/url" ],
    "index": null
  } ],
  "region_aggregation": false
}
```

响应示例

状态码： 200

Request succeeded.

```
{
  "id": "f88c5eabff9b4ff9ba6e7dd8e38128ba",
  "policyid": "d471eef691684f1c8d7784532fd8f4bd",
  "name": "test55",
  "description": "",
  "mode": 1,
  "conditions": [ {
    "category": "url",
    "contents": [ "/url" ],
    "logic_operation": "contain"
  } ],
  "action": {
    "category": "captcha"
  },
  "producer": 1,
  "unaggregation": false,
  "total_num": 0,
  "limit_num": 10,
  "limit_period": 60,
  "lock_time": 0,
  "tag_type": "ip",
  "aging_time": 0,
  "region_aggregation": false,
  "domain_aggregation": false
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改项目id为project_id，防护策略id为policy_id，规则id为rule_id的cc规则，修改限速模式为ip限速，限速频率为10次，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，规则的名称为test55，不开启域名聚合统计，cc规则防护规则限速条件是URL包含“/url”，没有子字段，不开启全局计数。

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

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

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

public class UpdateCcRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateCcRuleRequest request = new UpdateCcRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateCcRuleRequestBody body = new UpdateCcRuleRequestBody();
        CreateCcRuleRequestBodyAction actionbody = new CreateCcRuleRequestBodyAction();
        actionbody.withCategory(CreateCcRuleRequestBodyAction.CategoryEnum.fromValue("captcha"));
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("/url");
        List<CcCondition> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new CcCondition()
                .withCategory(CcCondition.CategoryEnum.fromValue("url"))
                .withLogicOperation("contain")
                .withContents(listConditionsContents)
        );
        body.withDescription("");
        body.withRegionAggregation(false);
        body.withDomainAggregation(false);
        body.withLimitPeriod(60);
        body.withLimitNum(10);
        body.withTagType(UpdateCcRuleRequestBody.TagTypeEnum.fromValue("ip"));
        body.withAction(actionbody);
        body.withConditions(listbodyConditions);
        body.withMode(1);
        body.withName("test55");
        request.withBody(body);
        try {
            UpdateCcRuleResponse response = client.updateCcRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

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

Python

修改项目id为project_id，防护策略id为policy_id，规则id为rule_id的cc规则，修改限速模式为ip限速，限速频率为10次，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，规则的名称为test55，不开启域名聚合统计，cc规则防护规则限速条件是URL包含“/url”，没有子字段，不开启全局计数。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateCcRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        actionbody = CreateCcRuleRequestBodyAction(
            category="captcha"
        )
        listContentsConditions = [
            "/url"
        ]
        listConditionsbody = [
            CcCondition(
                category="url",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
        request.body = UpdateCcRuleRequestBody(
            description="",
            region_aggregation=False,
            domain_aggregation=False,
            limit_period=60,
            limit_num=10,
            tag_type="ip",
            action=actionbody,
            conditions=listConditionsbody,
            mode=1,
```

```
        name="test55"
    )
    response = client.update_cc_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

修改项目id为project_id，防护策略id为policy_id，规则id为rule_id的cc规则，修改限速模式为ip限速，限速频率为10次，限速周期为60秒，请求次数限制到达后采取的防护动作为人机验证，cc规则防护模式为高级，规则的名称为test55，不开启域名聚合统计，cc规则防护规则限速条件是URL包含“/url”，没有子字段，不开启全局计数。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateCcRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    actionbody := &model.CreateCcRuleRequestBodyAction{
        Category: model.GetCreateCcRuleRequestBodyActionCategoryEnum().CAPTCHA,
    }
    var listContentsConditions = []string{
        "/url",
    }
    var listConditionsbody = []model.CcCondition{
        {
            Category: model.GetCcConditionCategoryEnum().URL,
            LogicOperation: "contain",
            Contents: &listContentsConditions,
        },
    }
    descriptionUpdateCcRuleRequestBody:= ""
    regionAggregationUpdateCcRuleRequestBody:= false
    domainAggregationUpdateCcRuleRequestBody:= false
```



```
nameUpdateCcRuleRequestBody:= "test55"
request.Body = &model.UpdateCcRuleRequestBody{
    Description: &descriptionUpdateCcRuleRequestBody,
    RegionAggregation: &regionAggregationUpdateCcRuleRequestBody,
    DomainAggregation: &domainAggregationUpdateCcRuleRequestBody,
    LimitPeriod: int32(60),
    LimitNum: int32(10),
    TagType: model.GetUpdateCcRuleRequestBodyTagTypeEnum().IP,
    Action: actionbody,
    Conditions: listConditionsbody,
    Mode: int32(1),
    Name: &nameUpdateCcRuleRequestBody,
}
response, err := client.UpdateCcRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	Request succeeded.
400	Request failed.
401	The token does not have required permissions.
500	Internal server error.

错误码

请参见[错误码](#)。

4.4.6 删除 cc 防护规则

功能介绍

删除cc防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}

表 4-379 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	策略id (策略id从查询防护策略列表接口获取)
rule_id	是	String	cc规则的ID。可以通过调用 ListCcRules API获取。

表 4-380 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-381 请求 Header 参数

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

响应参数

状态码: 200

表 4-382 响应 Body 参数

参数	参数类型	描述
name	String	规则名称
id	String	Rule ID.
policyid	String	Policy ID.

参数	参数类型	描述
url	String	当mode值为0时，该参数有返回值。规则应用的URL链接，不包含域名。
prefix	Boolean	路径是否为前缀模式，当防护url以*结束，则为前缀模式。
mode	Integer	cc规则防护模式，对应console上的mode，现在只支持创建高级cc规则防护模式。 <ul style="list-style-type: none">0：标准，只支持对域名的防护路径做限制。1：高级，支持对路径、IP、Cookie、Header、Params字段做限制。
status	Integer	规则状态，0：关闭，1：开启
conditions	Array of CcCondition objects	cc规则防护规则限速条件，当cc防护规则为高级模式（mode参数值为1）时，该参数必填。
action	action object	请求次数限制到达后采取的防护动作
tag_type	String	限速模式： <ul style="list-style-type: none">ip：IP限速，根据IP区分单个Web访问者。cookie：用户限速，根据Cookie键值区分单个Web访问者。header：用户限速，根据Header区分单个Web访问者。other：根据Referer（自定义请求访问的来源）字段区分单个Web访问者。policy：策略限速domain：域名限速url：url限速

参数	参数类型	描述
tag_index	String	用户标识，当限速模式为用户限速(cookie或header)时，需要传该参数。 <ul style="list-style-type: none">选择cookie时，设置cookie字段名，即用户需要根据网站实际情况配置唯一可识别Web访问者的cookie中的某属性变量名。用户标识的cookie，不支持正则，必须完全匹配。例如：如果网站使用cookie中的某个字段name唯一标识用户，那么可以选取name字段来区分Web访问者。选择header时，设置需要防护的自定义HTTP首部，即用户需要根据网站实际情况配置可识别Web访问者的HTTP首部。
tag_condition	tag_condition object	用户标识，当限速模式为other时，需要传该参数。根据Referer（自定义请求访问的来源）字段区分单个Web访问者
limit_num	Integer	限制频率，单位为次，范围为1~2147483647
limit_period	Integer	限速周期，单位为秒，范围1~3600
unlock_num	Integer	放行频率，单位为次，范围为0~2147483647。只有当防护动作类型为dynamic_block时，才需要传该参数。
lock_time	Integer	阻断时间，单位为秒，范围为0~65535。当“防护动作”选择“阻断”时，可设置阻断后恢复正常访问页面的时间。
domain_aggregation	Boolean	是否开启域名聚合统计。
region_aggregation	Boolean	是否开启全局计数。
description	String	规则描述
total_num	Integer	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
unaggregation	Boolean	该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
aging_time	Integer	规则老化时间，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

参数	参数类型	描述
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数
timestamp	Long	创建规则时间戳

表 4-383 CcCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是url，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip或者ipv6，匹配逻辑可以为：equal、not_equal、equal_any或者not_equal_all 如果字段类型category是params、cookie或者header，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal、len_not_equal、num_greater、num_less、num_equal、num_not_equal、exist或者not_exist
contents	Array of strings	条件列表逻辑匹配内容。当logic_operation参数不以any或者all结尾时，需要传该参数。

参数	参数类型	描述
value_list_id	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。
index	String	子字段，当字段类型（category）选择“params”、“cookie”、“header”时，请根据实际需求配置子字段且该参数必填。

表 4-384 action

参数	参数类型	描述
category	String	<p>动作类型：</p> <ul style="list-style-type: none"> captcha: 人机验证，阻断后用户需要输入正确的验证码，恢复正确的访问页面。 -block: 阻断。 log: 仅记录 dynamic_block: 上一个限速周期内，请求频率超过“限速频率”将被阻断，那么在下一个限速周期内，请求频率超过“放行频率”将被阻断。 注：只有当cc防护规则模式为高级模式时才支持设置dynamic_block防护动作。
detail	detail object	<p>阻断页面信息。当防护动作（category）选择阻断（block）或者动态阻断（dynamic_block）时，需要设置返回的阻断页面。</p> <ul style="list-style-type: none"> 如果需要返回的阻断页面为系统默认的阻断页面，不需要传该参数。 如果用户想防护自定义的阻断页面，可以通过此参数设置

表 4-385 detail

参数	参数类型	描述
response	response object	阻断页面

表 4-386 response

参数	参数类型	描述
content_type	String	内容类型，值可为“application/json”、“text/html”、“text/xml”。
content	String	阻断页面内容

表 4-387 tag_condition

参数	参数类型	描述
category	String	用户标识字段，其值固定为referer
contents	Array of strings	用户标识字段内容

状态码： 400

表 4-388 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-389 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-390 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-391 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-392 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-393 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id，防护规则为policy_id，规则id为rule_id的cc防护规则。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/cc/{rule_id}?
```

响应示例

状态码： 200

Request succeeded.

```
{
  "id": "f88c5eabff9b4ff9ba6e7dd8e38128ba",
  "policyid": "d471eef691684f1c8d7784532fd8f4bd",
  "name": "test55",
  "timestamp": 1678873040603,
  "description": "",
  "status": 1,
  "mode": 1,
  "conditions": [ {
    "category": "url",
    "contents": [ "/url" ],
    "logic_operation": "contain"
  } ],
  "action": {
    "category": "captcha"
  },
  "producer": 1,
  "unaggregation": false,
  "total_num": 0,
  "limit_num": 10,
  "limit_period": 60,
  "lock_time": 0,
  "tag_type": "ip",
  "aging_time": 0,
  "region_aggregation": false,
  "domain_aggregation": 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.waf.v1.region.WafRegion;
```

```
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteCcRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteCcRuleRequest request = new DeleteCcRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeleteCcRuleResponse response = client.deleteCcRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(WafRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = DeleteCcRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.delete_cc_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteCcRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeleteCcRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	Request succeeded.
400	Request failed.
401	The token does not have required permissions.
500	Internal server error.

错误码

请参见[错误码](#)。

4.4.7 查询精准防护规则列表

功能介绍

查询精准防护规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/custom

表 4-394 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-395 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
offset	是	Integer	偏移量，表示查询该偏移量之后的记录。
limit	是	Integer	查询返回记录的数量限制。

请求参数

表 4-396 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-397 响应 Body 参数

参数	参数类型	描述
total	Integer	数量
items	Array of CustomRule objects	防护规则列表

表 4-398 CustomRule

参数	参数类型	描述
id	String	规则ID
policyid	String	策略ID

参数	参数类型	描述
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
conditions	Array of conditions objects	匹配条件列表, 匹配条件必须同时满足。
action	CustomAction object	精准防护规则命中后操作对象
action_mode	Boolean	预留参数, 可忽略。
priority	Integer	执行该规则的优先级, 值越小, 优先级越高, 值相同时, 规则创建时间早, 优先级越高。取值范围: 0到1000。
timestamp	Long	创建精准防护规则的时间戳
time	Boolean	精准防护规则生效时间: <ul style="list-style-type: none"> “false”: 表示该规则立即生效。 “true”: 表示自定义生效时间。
start	Long	精准防护规则生效的起始时间戳 (秒)。当time=true, 才会返回该参数。
terminal	Long	精准防护规则生效的终止时间戳 (秒)。当time=true, 才会返回该参数。
producer	Integer	规则创建对象, 该参数为预留参数, 用于后续功能扩展, 当前请用户忽略该参数
name	String	规则名称

表 4-399 conditions

参数	参数类型	描述
category	String	字段类型。可选值为: url、user-agent、ip、params、cookie、referer、header、request_line、method、request

参数	参数类型	描述
index	String	子字段： <ul style="list-style-type: none">• 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数• 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段
logic_operation	String	条件匹配逻辑。
contents	Array of strings	条件匹配的内容
value_list_id	String	引用表id。

表 4-400 CustomAction

参数	参数类型	描述
category	String	操作类型。 <ul style="list-style-type: none">• “block”：拦截。• “pass”：放行。• “log”：仅记录
followed_action_id	String	攻击惩罚规则id，只有当category参数值为block时才可配置该参数

状态码： 400

表 4-401 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-402 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-403 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-404 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-405 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-406 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护规则id为policy_id的精准防护规则列表

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/custom?offset=0&limit=1
```

响应示例

状态码: 200

ok

```
{
  "total": 1,
  "items": [ {
    "id": "c637138b6fe048e4a797d1c3712e85b3",
    "policyid": "41424a44c2904e1b9e505ccdbfe8c1fb",
    "timestamp": 1679888279852,
    "description": "",
    "status": 1,
    "time": false,
    "priority": 50,
    "action_mode": false,
    "conditions": [ {
      "category": "url",
      "contents": [ "test" ],
      "logic_operation": "contain"
    } ],
    "action": {
      "category": "block"
    },
    "producer": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListCustomRulesSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListCustomRulesRequest request = new ListCustomRulesRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListCustomRulesResponse response = client.listCustomRules(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *
```

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListCustomRulesRequest()
        request.policy_id = "{policy_id}"
        response = client.list_custom_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

```
}  
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.8 创建精准防护规则

功能介绍

创建精准防护规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/custom

表 4-407 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-408 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-409 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)。
Content-Type	是	String	内容类型

表 4-410 请求 Body 参数

参数	是否必选	参数类型	描述
time	是	Boolean	精准防护规则生效时间: <ul style="list-style-type: none"> “false”：表示该规则立即生效。 “true”：表示自定义生效时间。
start	否	Long	精准防护规则生效的起始时间戳（秒）。当time=true，才需要填写该参数。
terminal	否	Long	精准防护规则生效的终止时间戳（秒）。当time=true，才需要填写该参数。
description	否	String	规则描述
conditions	是	Array of CustomConditions objects	匹配条件列表
action	是	CustomAction object	精准防护规则命中后操作对象

参数	是否必选	参数类型	描述
priority	是	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。
name	是	String	规则名称

表 4-411 CustomConditions

参数	是否必选	参数类型	描述
category	否	String	字段类型。可选值为：url、user-agent、ip、params、cookie、referer、header、request_line、method、request
index	否	String	子字段： <ul style="list-style-type: none">• 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数• 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段

参数	是否必选	参数类型	描述
logic_operation	否	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是 url、user-agent或者 referer, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是ip, 匹配逻辑可以为: equal、not_equal、equal_any或者 not_equal_all 如果字段类型category是 method, 匹配逻辑可以为: equal或者not_equal n - 如果字段类型category是 request_line或者request, 匹配逻辑可以为: len_greater、len_less、len_equal或者len_not_equal 如果字段类型category是 params、cookie或者header, 匹配逻辑可以为: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、

参数	是否必选	参数类型	描述
			len_equal、 len_not_equal、 num_greater、num_less、 num_equal、 num_not_equal、exist或者 not_exist
contents	否	Array of strings	条件列表逻辑匹配内容。当 logic_operation参数不以any或者all结尾时，需要传该参数。
value_list_id	否	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。

表 4-412 CustomAction

参数	是否必选	参数类型	描述
category	是	String	操作类型。 <ul style="list-style-type: none"> “block”：拦截。 “pass”：放行。 “log”：仅记录
followed_action_id	否	String	攻击惩罚规则id，只有当category参数值为block时才可配置该参数

响应参数

状态码： 200

表 4-413 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	规则名称
policyid	String	策略id
description	String	规则描述
status	Integer	规则状态，0：关闭，1：开启

参数	参数类型	描述
conditions	Array of conditions objects	匹配条件列表，匹配条件必须同时满足。
action	CustomAction object	精准防护规则命中后操作对象
action_mode	Boolean	预留参数，可忽略。
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。
timestamp	Long	创建精准防护规则的
time	Boolean	精准防护规则生效时间： <ul style="list-style-type: none"> “false”：表示该规则立即生效。 “true”：表示自定义生效时间。
start	Long	精准防护规则生效的起始时间戳（秒）。当time=true，才会返回该参数。
terminal	Long	精准防护规则生效的终止时间戳（秒）。当time=true，才会返回该参数。
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

表 4-414 conditions

参数	参数类型	描述
category	String	字段类型。可选值为：url、user-agent、ip、params、cookie、referer、header、request_line、method、request
index	String	子字段： <ul style="list-style-type: none"> 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段
logic_operation	String	条件匹配逻辑。
contents	Array of strings	条件匹配的内容

参数	参数类型	描述
value_list_id	String	引用表id。

表 4-415 CustomAction

参数	参数类型	描述
category	String	操作类型。 <ul style="list-style-type: none">“block”：拦截。“pass”：放行。“log”：仅记录
followed_action_id	String	攻击惩罚规则id，只有当category参数值为block时才配置该参数

状态码： 400

表 4-416 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-417 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-418 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-419 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-420 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-421 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id和防护策略id为policy_id的防护策略下创建精准防护规则。规则的名称test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo字段包含“demo”，精准防护规则生效时间为立即生效。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/custom?enterprise_project_id=0
{
  "name": "test55",
  "description": "",
  "action": {
    "category": "block"
  },
  "priority": 50,
  "conditions": [ {
    "category": "header",
    "logic_operation": "contain",
    "index": "demo",
    "contents": [ "demo" ]
  } ],
  "time": false
}
```

响应示例

状态码： 200

ok

```
{
  "action": {
    "category": "block"
  },
  "action_mode": false,
  "conditions": [ {
    "category": "header",
    "index": "demo",
    "logic_operation": "contain",
    "contents": [ "demo" ]
  } ],
  "description": "",
  "name": "test55",
  "id": "2a3caa2bc9814c09ad73d02e3485b4a4",
  "policyid": "1f016cde588646aca3fb19f277c44d03",
  "priority": 50,
  "status": 1,
  "time": false,
  "timestamp": 1656495488880
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id和防护策略id为policy_id的防护策略下创建精准防护规则。规则的名称test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo字段包含“demo”，精准防护规则生效时间为立即生效。

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

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

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

public class CreateCustomRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateCustomRuleRequest request = new CreateCustomRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateCustomRuleRequestBody body = new CreateCustomRuleRequestBody();
        CustomAction actionbody = new CustomAction();
        actionbody.withCategory(CustomAction.CategoryEnum.fromValue("block"));
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("demo");
        List<CustomConditions> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new CustomConditions()
                .withCategory(CustomConditions.CategoryEnum.fromValue("header"))
                .withIndex("demo")
                .withLogicOperation("contain")
                .withContents(listConditionsContents)
        );
        body.withName("test55");
        body.withPriority(50);
        body.withAction(actionbody);
        body.withConditions(listbodyConditions);
        body.withDescription("");
        body.withTime(false);
    }
}
```

```
request.withBody(body);
try {
    CreateCustomRuleResponse response = client.createCustomRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id和防护策略id为policy_id的防护策略下创建精准防护规则。规则的名称test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo字段包含“demo”，精准防护规则生效时间为立即生效。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateCustomRuleRequest()
        request.policy_id = "{policy_id}"
        actionbody = CustomAction(
            category="block"
        )
        listContentsConditions = [
            "demo"
        ]
        listConditionsbody = [
            CustomConditions(
                category="header",
                index="demo",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
        request.body = CreateCustomRuleRequestBody(
```

```
        name="test55",
        priority=50,
        action=actionbody,
        conditions=listConditionsbody,
        description="",
        time=False
    )
    response = client.create_custom_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在项目id为project_id和防护策略id为policy_id的防护策略下创建精准防护规则。规则的名称test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo字段包含“demo”，精准防护规则生效时间为立即生效。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

    request := &model.CreateCustomRuleRequest{}
    request.PolicyId = "{policy_id}"
    actionbody := &model.CustomAction{
        Category: model.GetCustomActionCategoryEnum().BLOCK,
    }
    var listContentsConditions = []string{
        "demo",
    }
    categoryConditions:= model.GetCustomConditionsCategoryEnum().HEADER
    indexConditions:= "demo"
    logicOperationConditions:= "contain"
    var listConditionsbody = []model.CustomConditions{
        {
            Category: &categoryConditions,
```

```
Index: &indexConditions,  
LogicOperation: &logicOperationConditions,  
Contents: &listContentsConditions,  
},  
}  
descriptionCreateCustomRuleRequestBody:= ""  
request.Body = &model.CreateCustomRuleRequestBody{  
    Name: "test55",  
    Priority: int32(50),  
    Action: actionbody,  
    Conditions: listConditionsbody,  
    Description: &descriptionCreateCustomRuleRequestBody,  
    Time: false,  
}  
response, err := client.CreateCustomRule(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.9 根据 Id 查询精准防护规则

功能介绍

根据Id查询精准防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}

表 4-422 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防护规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	精准防护规则id, 通过查询精准防护规则列表接口 (ListCustomRules) 获取

表 4-423 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-424 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-425 响应 Body 参数

参数	参数类型	描述
id	String	规则id

参数	参数类型	描述
name	String	规则名称
policyid	String	策略id
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
conditions	Array of conditions objects	匹配条件列表, 匹配条件必须同时满足。
action	CustomAction object	精准防护规则命中后操作对象
action_mode	Boolean	预留参数, 可忽略。
priority	Integer	执行该规则的优先级, 值越小, 优先级越高, 值相同时, 规则创建时间早, 优先级越高。取值范围: 0到1000。
timestamp	Long	创建精准防护规则的
time	Boolean	精准防护规则生效时间: <ul style="list-style-type: none">“false”: 表示该规则立即生效。“true”: 表示自定义生效时间。
start	Long	精准防护规则生效的起始时间戳 (秒)。当time=true, 才会返回该参数。
terminal	Long	精准防护规则生效的终止时间戳 (秒)。当time=true, 才会返回该参数。
producer	Integer	规则创建对象, 该参数为预留参数, 用于后续功能扩展, 当前请用户忽略该参数

表 4-426 conditions

参数	参数类型	描述
category	String	字段类型。可选值为: url、user-agent、ip、params、cookie、referer、header、request_line、method、request

参数	参数类型	描述
index	String	子字段： <ul style="list-style-type: none">• 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数• 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段
logic_operation	String	条件匹配逻辑。
contents	Array of strings	条件匹配的内容
value_list_id	String	引用表id。

表 4-427 CustomAction

参数	参数类型	描述
category	String	操作类型。 <ul style="list-style-type: none">• “block”：拦截。• “pass”：放行。• “log”：仅记录
followed_action_id	String	攻击惩罚规则id，只有当category参数值为block时才配置该参数

状态码： 400

表 4-428 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-429 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-430 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-431 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-432 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-433 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的精准防护策略规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}?
```

响应示例

状态码: 200

ok

```
{
  "action": {
    "category": "block"
  },
  "action_mode": false,
  "conditions": [ {
    "category": "header",
    "index": "demo",
    "logic_operation": "contain",
    "contents": [ "demo" ]
  } ],
  "description": "",
  "name": "test55",
  "id": "2a3caa2bc9814c09ad73d02e3485b4a4",
  "policyid": "1f016cde588646aca3fb19f277c44d03",
  "priority": 50,
  "status": 1,
  "time": false,
  "timestamp": 1656495488880
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowCustomRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowCustomRuleRequest request = new ShowCustomRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowCustomRuleResponse response = client.showCustomRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = ShowCustomRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.show_custom_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowCustomRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowCustomRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.10 更新精准防护规则

功能介绍

更新精准防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}

表 4-434 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	精准防护规则id，通过查询精准防护规则列表接口（ListCustomRules）获取

表 4-435 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-436 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)。
Content-Type	是	String	内容类型

表 4-437 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	规则名称
time	是	Boolean	精准防护规则生效时间: <ul style="list-style-type: none"> “false”：表示该规则立即生效。 “true”：表示自定义生效时间。
start	否	Long	精准防护规则生效的起始时间戳（秒）。当time=true，才需要填写该参数。
terminal	否	Long	精准防护规则生效的终止时间戳（秒）。当time=true，才需要填写该参数。
description	否	String	规则描述
conditions	是	Array of CustomConditions objects	匹配条件列表
action	是	CustomAction object	精准防护规则命中后操作对象

参数	是否必选	参数类型	描述
priority	是	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-438 CustomConditions

参数	是否必选	参数类型	描述
category	否	String	字段类型。可选值为：url、user-agent、ip、params、cookie、referer、header、request_line、method、request
index	否	String	子字段： <ul style="list-style-type: none">• 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数• 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段

参数	是否必选	参数类型	描述
logic_operation	否	String	<p>条件列表匹配逻辑。</p> <ul style="list-style-type: none"> 如果字段类型category是 url、user-agent 或者 referer，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、len_equal 或者 len_not_equal 如果字段类型category是 ip，匹配逻辑可以为：equal、not_equal、equal_any 或者 not_equal_all 如果字段类型category是 method，匹配逻辑可以为：equal 或者 not_equal n - 如果字段类型category是 request_line 或者 request，匹配逻辑可以为：len_greater、len_less、len_equal 或者 len_not_equal 如果字段类型category是 params、cookie 或者 header，匹配逻辑可以为：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all、len_greater、len_less、

参数	是否必选	参数类型	描述
			len_equal、 len_not_equal、 num_greater、num_less、 num_equal、 num_not_equal、exist或者 not_exist
contents	否	Array of strings	条件列表逻辑匹配内容。当 logic_operation参数不以any或者all结尾时，需要传该参数。
value_list_id	否	String	引用表id。当logic_operation参数以any或者all结尾时，需要传该参数。此外，引用表类型要与category类型保持一致。

表 4-439 CustomAction

参数	是否必选	参数类型	描述
category	是	String	操作类型。 <ul style="list-style-type: none"> “block”：拦截。 “pass”：放行。 “log”：仅记录
followed_action_id	否	String	攻击惩罚规则id，只有当category参数值为block时才可配置该参数

响应参数

状态码： 200

表 4-440 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	规则名称
policyid	String	策略id
description	String	规则描述
status	Integer	规则状态，0：关闭，1：开启

参数	参数类型	描述
conditions	Array of conditions objects	匹配条件列表，匹配条件必须同时满足。
action	CustomAction object	精准防护规则命中后操作对象
action_mode	Boolean	预留参数，可忽略。
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。
time	Boolean	精准防护规则生效时间： <ul style="list-style-type: none">“false”：表示该规则立即生效。“true”：表示自定义生效时间。
start	Long	精准防护规则生效的起始时间戳（秒）。当time=true，才会返回该参数。
terminal	Long	精准防护规则生效的终止时间戳（秒）。当time=true，才会返回该参数。
producer	Integer	规则创建对象，该参数为预留参数，用于后续功能扩展，当前请用户忽略该参数

表 4-441 conditions

参数	参数类型	描述
category	String	字段类型。可选值为：url、user-agent、ip、params、cookie、referer、header、request_line、method、request
index	String	子字段： <ul style="list-style-type: none">字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段
logic_operation	String	条件匹配逻辑。
contents	Array of strings	条件匹配的内容
value_list_id	String	引用表id。

表 4-442 CustomAction

参数	参数类型	描述
category	String	操作类型。 <ul style="list-style-type: none">“block”：拦截。“pass”：放行。“log”：仅记录
followed_action_id	String	攻击惩罚规则id，只有当category参数值为block时才可配置该参数

状态码： 400

表 4-443 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-444 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-445 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-446 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-447 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-448 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

修改项目id为project_id和防护策略id为policy_id的规则id为rule_id的精准防护策略规则。修改规则的名称为test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo2字段包含“demo”，精准防护规则生效时间为立即生效。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}?
```

```
{
  "name": "test55",
  "description": "",
  "action": {
    "category": "block"
  },
  "priority": 50,
  "conditions": [ {
    "category": "header",
    "logic_operation": "contain",
    "index": "demo2",
    "contents": [ "demo" ]
  } ],
  "time": false
}
```

响应示例

状态码： 200

ok

```
{
  "action": {
    "category": "block"
  },
  "action_mode": false,
  "conditions": [ {
    "category": "header",
    "index": "demo2",
    "logic_operation": "contain",
    "contents": [ "demo" ]
  } ],
  "description": "",
  "name": "test55",
  "id": "2a3caa2bc9814c09ad73d02e3485b4a4",
  "policyid": "1f016cde588646aca3fb19f277c44d03",
  "priority": 50,
  "status": 1,
  "time": false
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改项目id为project_id和防护策略id为policy_id的规则id为rule_id的精准防护策略规则。修改规则的名称为test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo2字段包含“demo”，精准防护规则生效时间为立即生效。

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

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

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

public class UpdateCustomRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateCustomRuleRequest request = new UpdateCustomRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateCustomRuleRequestBody body = new UpdateCustomRuleRequestBody();
        CustomAction actionbody = new CustomAction();
        actionbody.withCategory(CustomAction.CategoryEnum.fromValue("block"));
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("demo");
        List<CustomConditions> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new CustomConditions()
                .withCategory(CustomConditions.CategoryEnum.fromValue("header"))
                .withIndex("demo2")
                .withLogicOperation("contain")
                .withContents(listConditionsContents)
        );
        body.withPriority(50);
        body.withAction(actionbody);
        body.withConditions(listbodyConditions);
        body.withDescription("");
        body.withTime(false);
        body.withName("test55");
        request.withBody(body);
        try {
            UpdateCustomRuleResponse response = client.updateCustomRule(request);
            System.out.println(response.toString());
        }
    }
}
```

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

Python

修改项目id为project_id和防护策略id为policy_id的规则id为rule_id的精准防护策略规则。修改规则的名称为test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo2字段包含“demo”，精准防护规则生效时间为立即生效。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateCustomRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        actionbody = CustomAction(
            category="block"
        )
        listContentsConditions = [
            "demo"
        ]
        listConditionsbody = [
            CustomConditions(
                category="header",
                index="demo2",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
        request.body = UpdateCustomRuleRequestBody(
            priority=50,
            action=actionbody,
            conditions=listConditionsbody,
```

```
        description="",
        time=False,
        name="test55"
    )
    response = client.update_custom_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

修改项目id为project_id和防护策略id为policy_id的规则id为rule_id的精准防护策略规则。修改规则的名称为test55，精准防护规则命中后防护动作为拦截，执行该规则的优先级为50，匹配条件为header中发现demo2字段包含“demo”，精准防护规则生效时间为立即生效。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateCustomRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    actionbody := &model.CustomAction{
        Category: model.GetCustomActionCategoryEnum().BLOCK,
    }
    var listContentsConditions = []string{
        "demo",
    }
    categoryConditions:= model.GetCustomConditionsCategoryEnum().HEADER
    indexConditions:= "demo2"
    logicOperationConditions:= "contain"
    var listConditionsbody = []model.CustomConditions{
        {
            Category: &categoryConditions,
            Index: &indexConditions,
            LogicOperation: &logicOperationConditions,
        }
    }
}
```

```
        Contents: &listContentsConditions,
    },
}
descriptionUpdateCustomRuleRequestBody:= ""
request.Body = &model.UpdateCustomRuleRequestBody{
    Priority: int32(50),
    Action: actionbody,
    Conditions: listConditionsbody,
    Description: &descriptionUpdateCustomRuleRequestBody,
    Time: false,
    Name: "test55",
}
response, err := client.UpdateCustomRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.11 删除精准防护规则

功能介绍

删除精准防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}

表 4-449 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防护规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	精准防护规则id, 通过查询精准防护规则列表接口 (ListCustomRules) 获取

表 4-450 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-451 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-452 响应 Body 参数

参数	参数类型	描述
id	String	规则id

参数	参数类型	描述
name	String	规则名称
policyid	String	策略id
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
conditions	Array of conditions objects	匹配条件列表, 匹配条件必须同时满足。
action	CustomAction object	精准防护规则命中后操作对象
action_mode	Boolean	预留参数, 可忽略。
priority	Integer	执行该规则的优先级, 值越小, 优先级越高, 值相同时, 规则创建时间早, 优先级越高。取值范围: 0到1000。
timestamp	Long	创建精准防护规则的
time	Boolean	精准防护规则生效时间: <ul style="list-style-type: none"> “false”: 表示该规则立即生效。 “true”: 表示自定义生效时间。
start	Long	精准防护规则生效的起始时间戳 (秒)。当time=true, 才会返回该参数。
terminal	Long	精准防护规则生效的终止时间戳 (秒)。当time=true, 才会返回该参数。
producer	Integer	规则创建对象, 该参数为预留参数, 用于后续功能扩展, 当前请用户忽略该参数

表 4-453 conditions

参数	参数类型	描述
category	String	字段类型。可选值为: url、user-agent、ip、params、cookie、referer、header、request_line、method、request

参数	参数类型	描述
index	String	子字段： <ul style="list-style-type: none">• 字段类型为url、user-agent、ip、refer、request_line、method、request时，不需要传index参数• 字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段
logic_operation	String	条件匹配逻辑。
contents	Array of strings	条件匹配的内容
value_list_id	String	引用表id。

表 4-454 CustomAction

参数	参数类型	描述
category	String	操作类型。 <ul style="list-style-type: none">• “block”：拦截。• “pass”：放行。• “log”：仅记录
followed_action_id	String	攻击惩罚规则id，只有当category参数值为block时才可配置该参数

状态码： 400

表 4-455 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-456 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-457 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-458 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-459 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-460 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的精准防护策略规则。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/custom/{rule_id}?
```

响应示例

状态码: 200

ok

```
{
  "action": {
    "category": "block"
  },
  "action_mode": false,
  "conditions": [ {
    "category": "header",
    "index": "demo",
    "logic_operation": "contain",
    "contents": [ "demo" ]
  } ],
  "description": "",
  "id": "2a3caa2bc9814c09ad73d02e3485b4a4",
  "policyid": "1f016cde588646aca3fb19f277c44d03",
  "priority": 50,
  "status": 1,
  "time": false,
  "timestamp": 1656495488880
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteCustomRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteCustomRuleRequest request = new DeleteCustomRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeleteCustomRuleResponse response = client.deleteCustomRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = DeleteCustomRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.delete_custom_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteCustomRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeleteCustomRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.12 创建全局白名单(原误报屏蔽)规则

功能介绍

创建全局白名单(原误报屏蔽)规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/ignore

表 4-461 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-462 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-463 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)。
Content-Type	是	String	内容类型

表 4-464 请求 Body 参数

参数	是否必选	参数类型	描述
domain	是	Array of strings	防护域名或防护网站，数组长度为0时，代表规则对全部域名或防护网站生效
conditions	是	Array of CreateCondition objects	条件列表
mode	是	Integer	固定值为1,代表v2版本误报屏蔽规则，v1版本仅用于兼容旧版本，不支持创建

参数	是否必选	参数类型	描述
rule	是	String	<p>需要屏蔽的规则，可屏蔽一个或者多个，屏蔽多个时使用半角符;分隔</p> <ul style="list-style-type: none"> 当需要屏蔽某一条内置规则时，该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询；也可以在防护事件的事件详情中查询内置规则id 当需要屏蔽web基础防护某一类规则时，该参数值为需要屏蔽的web基础防护某一类规则名。其中，xss: xss攻击；webshell: 网站木马；vuln: 其他类型攻击；sqli: sql注入攻击；robot: 恶意爬虫；rfi: 远程文件包含；lfi: 本地文件包含；cmdi: 命令注入攻击 当需要屏蔽Web基础防护模块，该参数值为: all 当需要屏蔽规则为所有检测模块时，该参数值为: bypass
advanced	否	IgnoreAdvanced object	高级配置项
description	否	String	屏蔽规则描述

表 4-465 CreateCondition

参数	是否必选	参数类型	描述
category	是	String	字段类型，可选值有ip、url、params、cookie、header
contents	是	Array of strings	内容,数组长度限制为1，内容格式根据字段类型变化，例如，字段类型为ip时，contents内容格式需为ip地址或ip地址段；字段类型为url时，contents内容格式需为标准url格式；字段类型为params,cookie,header时，内容的格式不做限制

参数	是否必选	参数类型	描述
logic_operation	是	String	匹配逻辑，匹配逻辑根据字段类型变化，字段类型为ip时，匹配逻辑支持（equal：等于，not_equal：不等于），字段类型为url、header、params、cookie时，匹配逻辑支持（equal：等于，not_equal：不等于，contain：包含，not_contain：不包含，prefix：前缀为，not_prefix：前缀不为，suffix：后缀为，not_suffix：后缀不为）
check_all_indexes_logic	否	Integer	该参数为预留参数，可忽略。
index	否	String	字段类型为ip且子字段为客户端ip时，不需要传index参数；子字段类型为X-Forwarded-For时，值为x-forwarded-for；字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段

表 4-466 IgnoreAdvanced

参数	是否必选	参数类型	描述
index	否	String	<p>字段类型，支持的字段类型有：params、cookie、header、body、multipart。</p> <ul style="list-style-type: none"> 当选择“params”、“cookie”或者“header”字段时，可以配置“全部”或根据需求配置子字段 当选择“body”或“multipart”字段时，可以配置“全部”
contents	否	Array of strings	指定字段类型的子字段，默认值为“全部”

响应参数

状态码： 200

表 4-467 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间戳
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
rule	String	需要屏蔽的规则, 可屏蔽一个或者多个, 屏蔽多个时使用半角符;分隔 <ul style="list-style-type: none">当需要屏蔽某一条内置规则时, 该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询;也可以在防护事件的事件详情中查询内置规则id当需要屏蔽web基础防护某一类规则时, 该参数值为需要屏蔽的web基础防护某一类规则名。其中, xss: xss攻击; webshell: 网站木马; vuln: 其他类型攻击; sqli: sql注入攻击; robot: 恶意爬虫; rfi: 远程文件包含; lfi: 本地文件包含; cmdi: 命令注入攻击当需要屏蔽Web基础防护模块, 该参数值为: all当需要屏蔽规则为所有检测模块时, 该参数值为: bypass
mode	Integer	版本号固定值为1,代表v2版本误报屏蔽规则, v1版本仅支持兼容旧版本, 不支持创建
conditions	Array of Condition objects	条件列表
advanced	IgnoreAdvanced object	高级配置项
domain	Array of strings	防护域名或防护网站

表 4-468 Condition

参数	参数类型	描述
category	String	字段类型, 可选值有ip、url、params、cookie、header

参数	参数类型	描述
contents	Array of strings	内容,数组长度限制为1,内容格式根据字段类型变化,例如,字段类型为ip时,contents内容格式需为ip地址或ip地址段;字段类型为url时,contents内容格式需为标准url格式;字段类型为params、cookie、header时,内容的格式不做限制
logic_operation	String	匹配逻辑,匹配逻辑根据字段类型变化,字段类型为ip时,匹配逻辑支持(equal:等于,not_equal:不等于),字段类型为url、header、params、cookie时,匹配逻辑支持(equal:等于,not_equal:不等于,contain:包含,not_contain:不包含,prefix:前缀为,not_prefix:前缀不为,suffix:后缀为,not_suffix:后缀不为)
check_all_indexes_logic	Integer	该参数为预留参数,可忽略。
index	String	字段类型为ip且子字段为客户端ip时,不存在index参数;子字段类型为X-Forwarded-For时,值为x-forwarded-for,字段类型为params、header、cookie并且子字段为自定义时,index的值为自定义子字段

表 4-469 IgnoreAdvanced

参数	参数类型	描述
index	String	字段类型,支持的字段类型有: params、cookie、header、body、multipart。 <ul style="list-style-type: none"> 当选择“params”、“cookie”或者“header”字段时,可以配置“全部”或根据需求配置子字段 当选择“body”或“multipart”字段时,可以配置“全部”
contents	Array of strings	指定字段类型的子字段,默认值为“全部”

状态码: 400

表 4-470 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-471 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-472 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-473 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-474 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-475 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id和防护策略id为policy_id的防护策略下创建全局白名单(原误报屏蔽)规则。防护域名为we.test.418lab.cn, 条件为url中包含”x.x.x.x“, 描述为”demo“, 需要屏蔽的规则id为091004

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/ignore?enterprise_project_id=0
```

```
{
  "domain": [ "we.test.418lab.cn" ],
  "conditions": [ {
    "category": "url",
    "logic_operation": "contain",
    "contents": [ "x.x.x.x" ],
    "index": null
  } ],
  "mode": 1,
```

```
"description" : "demo",  
"rule" : "091004"  
}
```

响应示例

状态码： 200

OK

```
{  
  "id" : "a57f20ced01e4e0d8bea8e7c49eea254",  
  "policyid" : "f385eceedf7c4c34a4d1def19eafbe85",  
  "timestamp" : 1650522310447,  
  "description" : "demo",  
  "status" : 1,  
  "rule" : "091004",  
  "mode" : 1,  
  "conditions" : [ {  
    "category" : "url",  
    "contents" : [ "x.x.x.x" ],  
    "logic_operation" : "contain"  
  } ],  
  "domain" : [ "we.test.418lab.cn" ]  
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id和防护策略id为policy_id的防护策略下创建全局白名单(原误报屏蔽)规则。防护域名为we.test.418lab.cn，条件为url中包含”x.x.x.x“，描述为”demo“，需要屏蔽的规则id为091004

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class CreateIgnoreRuleSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);
```

```
WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
CreateIgnoreRuleRequest request = new CreateIgnoreRuleRequest();
request.withPolicyId("{policy_id}");
CreateIgnoreRuleRequestBody body = new CreateIgnoreRuleRequestBody();
List<String> listConditionsContents = new ArrayList<>();
listConditionsContents.add("x.x.x.x");
List<CreateCondition> listbodyConditions = new ArrayList<>();
listbodyConditions.add(
    new CreateCondition()
        .withCategory("url")
        .withContents(listConditionsContents)
        .withLogicOperation("contain")
);
List<String> listbodyDomain = new ArrayList<>();
listbodyDomain.add("we.test.418lab.cn");
body.withDescription("demo");
body.withRule("091004");
body.withMode(1);
body.withConditions(listbodyConditions);
body.withDomain(listbodyDomain);
request.withBody(body);
try {
    CreateIgnoreRuleResponse response = client.createIgnoreRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id和防护策略id为policy_id的防护策略下创建全局白名单(原误报屏蔽)规则。防护域名为we.test.418lab.cn，条件为url中包含”x.x.x.x“，描述为”demo“，需要屏蔽的规则id为091004

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = CreateIgnoreRuleRequest()
    request.policy_id = "{policy_id}"
    listContentsConditions = [
        "x.x.x.x"
    ]
    listConditionsbody = [
        CreateCondition(
            category="url",
            contents=listContentsConditions,
            logic_operation="contain"
        )
    ]
    listDomainbody = [
        "we.test.418lab.cn"
    ]
    request.body = CreateIgnoreRuleRequestBody(
        description="demo",
        rule="091004",
        mode=1,
        conditions=listConditionsbody,
        domain=listDomainbody
    )
    response = client.create_ignore_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在项目id为project_id和防护策略id为policy_id的防护策略下创建全局白名单(原误报屏蔽)规则。防护域名为we.test.418lab.cn，条件为url中包含”x.x.x.x“，描述为”demo“，需要屏蔽的规则id为091004

```
package main

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

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

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

```
client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateIgnoreRuleRequest{}
request.PolicyId = "{policy_id}"
var listContentsConditions = []string{
    "x.x.x.x",
}
var listConditionsbody = []model.CreateCondition{
    {
        Category: "url",
        Contents: listContentsConditions,
        LogicOperation: "contain",
    },
}
var listDomainbody = []string{
    "we.test.418lab.cn",
}
descriptionCreateIgnoreRuleRequestBody:= "demo"
request.Body = &model.CreateIgnoreRuleRequestBody{
    Description: &descriptionCreateIgnoreRuleRequestBody,
    Rule: "091004",
    Mode: int32(1),
    Conditions: listConditionsbody,
    Domain: listDomainbody,
}
response, err := client.CreateIgnoreRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.13 查询全局白名单(原误报屏蔽)防护规则

功能介绍

查询全局白名单(原误报屏蔽)防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}

表 4-476 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	误报屏蔽规则id，您可以通过查询误报屏蔽规则列表（ListIgnoreRule）接口的响应体中的id字段获取误报屏蔽规则id

表 4-477 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-478 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-479 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间戳
description	String	规则描述
status	Integer	规则状态，0：关闭，1：开启
url	String	误报规则屏蔽路径，仅在mode为0的状态下有该字段

参数	参数类型	描述
rule	String	<p>需要屏蔽的规则，可屏蔽一个或者多个，屏蔽多个时使用半角符;分隔</p> <ul style="list-style-type: none"> 当需要屏蔽某一条内置规则时，该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询；也可以在防护事件的事件详情中查询内置规则id 当需要屏蔽web基础防护某一类规则时，该参数值为需要屏蔽的web基础防护某一类规则名。其中，xss: xss攻击；webshell: 网站木马；vuln: 其他类型攻击；sqli: sql注入攻击；robot: 恶意爬虫；rfi: 远程文件包含；lfi: 本地文件包含；cmdi: 命令注入攻击 当需要屏蔽Web基础防护模块，该参数值为：all 当需要屏蔽规则为所有检测模块时，该参数值为：bypass
mode	Integer	版本号，0代表v1旧版本，1代表v2新版本；mode为0时，不存在conditions字段，存在url和url_logic字段；mode为1时，不存在url和url_logic字段，存在conditions字段
url_logic	String	url匹配逻辑
conditions	Array of Condition objects	条件
advanced	IgnoreAdvanced object	高级配置项
domain	Array of strings	防护域名或防护网站

表 4-480 Condition

参数	参数类型	描述
category	String	字段类型，可选值有ip、url、params、cookie、header

参数	参数类型	描述
contents	Array of strings	内容,数组长度限制为1,内容格式根据字段类型变化,例如,字段类型为ip时,contents内容格式需为ip地址或ip地址段;字段类型为url时,contents内容格式需为标准url格式;字段类型为params、cookie、header时,内容的格式不做限制
logic_operation	String	匹配逻辑,匹配逻辑根据字段类型变化,字段类型为ip时,匹配逻辑支持(equal:等于,not_equal:不等于),字段类型为url、header、params、cookie时,匹配逻辑支持(equal:等于,not_equal:不等于,contain:包含,not_contain:不包含,prefix:前缀为,not_prefix:前缀不为,suffix:后缀为,not_suffix:后缀不为)
check_all_indexes_logic	Integer	该参数为预留参数,可忽略。
index	String	字段类型为ip且子字段为客户端ip时,不存在index参数;子字段类型为X-Forwarded-For时,值为x-forwarded-for,字段类型为params、header、cookie并且子字段为自定义时,index的值为自定义子字段

表 4-481 IgnoreAdvanced

参数	参数类型	描述
index	String	字段类型,支持的字段类型有: params、cookie、header、body、multipart。 <ul style="list-style-type: none"> 当选择“params”、“cookie”或者“header”字段时,可以配置“全部”或根据需求配置子字段 当选择“body”或“multipart”字段时,可以配置“全部”
contents	Array of strings	指定字段类型的子字段,默认值为“全部”

状态码: 400

表 4-482 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-483 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-484 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-485 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-486 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-487 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的全局白名单(原误报屏蔽)防护规则

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}
```

响应示例

状态码： 200

请求成功

```
{  
  "id": "16e81d9a9e0244359204d7f00326ee4f",  
  "policyid": "0681f69f94ac408e9688373e45a61fdb",  
}
```

```
"timestamp" : 1679106005786,
"description" : "",
"status" : 1,
"rule" : "webshell;vuln",
"mode" : 1,
"conditions" : [ {
  "category" : "url",
  "contents" : [ "/test" ],
  "logic_operation" : "contain"
}],
"domain" : [ ],
"advanced" : {
  "index" : "params",
  "contents" : [ ]
}
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowIgnoreRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowIgnoreRuleRequest request = new ShowIgnoreRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowIgnoreRuleResponse response = client.showIgnoreRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
        System.out.println(e.getStatusCode());
        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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowIgnoreRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_ignore_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowIgnoreRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.ShowIgnoreRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.14 更新全局白名单(原误报屏蔽)防护规则

功能介绍

更新全局白名单(原误报屏蔽)防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}

表 4-488 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	误报屏蔽规则id, 您可以通过查询误报屏蔽规则列表 (ListIgnoreRule) 接口的响应体中的id字段获取误报屏蔽规则id

表 4-489 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-490 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)。
Content-Type	是	String	内容类型

表 4-491 请求 Body 参数

参数	是否必选	参数类型	描述
domain	是	Array of strings	防护域名或防护网站, 数组长度为0时, 代表规则对全部域名或防护网站生效

参数	是否必选	参数类型	描述
conditions	是	Array of CreateCondition objects	条件列表
mode	是	Integer	固定值为1,代表v2版本误报屏蔽规则, v1版本仅用于兼容旧版本, 不支持创建
rule	是	String	<p>需要屏蔽的规则, 可屏蔽一个或者多个, 屏蔽多个时使用半角符;分隔</p> <ul style="list-style-type: none"> 当需要屏蔽某一条内置规则时, 该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询;也可以在防护事件的事件详情中查询内置规则id 当需要屏蔽web基础防护某一类规则时, 该参数值为需要屏蔽的web基础防护某一类规则名。其中, xss: xss攻击; webshell: 网站木马; vuln: 其他类型攻击; sql: sql注入攻击; robot: 恶意爬虫; rfi: 远程文件包含; lfi: 本地文件包含; cmdi: 命令注入攻击 当需要屏蔽Web基础防护模块, 该参数值为: all 当需要屏蔽规则为所有检测模块时, 该参数值为: bypass
advanced	否	IgnoreAdvanced object	高级配置项
description	否	String	屏蔽规则描述

表 4-492 CreateCondition

参数	是否必选	参数类型	描述
category	是	String	字段类型, 可选值有ip、url、params、cookie、header

参数	是否必选	参数类型	描述
contents	是	Array of strings	内容,数组长度限制为1,内容格式根据字段类型变化,例如,字段类型为ip时,contents内容格式需为ip地址或ip地址段;字段类型为url时,contents内容格式需为标准url格式;字段类型为params,cookie,header时,内容的格式不做限制
logic_operation	是	String	匹配逻辑,匹配逻辑根据字段类型变化,字段类型为ip时,匹配逻辑支持(equal:等于,not_equal:不等于),字段类型为url、header、params、cookie时,匹配逻辑支持(equal:等于,not_equal:不等于,contain:包含,not_contain:不包含,prefix:前缀为,not_prefix:前缀不为,suffix:后缀为,not_suffix:后缀不为)
check_all_indexes_logic	否	Integer	该参数为预留参数,可忽略。
index	否	String	字段类型为ip且子字段为客户端ip时,不需要传index参数;子字段类型为X-Forwarded-For时,值为x-forwarded-for;字段类型为params、header、cookie并且子字段为自定义时,index的值为自定义子字段

表 4-493 IgnoreAdvanced

参数	是否必选	参数类型	描述
index	否	String	<p>字段类型,支持的字段类型有: params、cookie、header、body、multipart。</p> <ul style="list-style-type: none"> 当选择“params”、“cookie”或者“header”字段时,可以配置“全部”或根据需求配置子字段 当选择“body”或“multipart”字段时,可以配置“全部”

参数	是否必选	参数类型	描述
contents	否	Array of strings	指定字段类型的子字段，默认值为“全部”

响应参数

状态码： 200

表 4-494 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间戳
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
url	String	误报规则屏蔽路径, 仅在mode为0的状态下有该字段
rule	String	需要屏蔽的规则, 可屏蔽一个或者多个, 屏蔽多个时使用半角符;分隔 <ul style="list-style-type: none">当需要屏蔽某一条内置规则时, 该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询; 也可以在防护事件的事件详情中查询内置规则id当需要屏蔽web基础防护某一类规则时, 该参数值为需要屏蔽的web基础防护某一类规则名。其中, xss: xss攻击; webshell: 网站木马; vuln: 其他类型攻击; sqli: sql注入攻击; robot: 恶意爬虫; rfi: 远程文件包含; lfi: 本地文件包含; cmdi: 命令注入攻击当需要屏蔽Web基础防护模块, 该参数值为: all当需要屏蔽规则为所有检测模块时, 该参数值为: bypass

参数	参数类型	描述
mode	Integer	版本号, 0代表v1旧版本, 1代表v2新版本; mode为0时, 不存在conditions字段, 存在url和url_logic字段; mode为1时, 不存在url和url_logic字段, 存在conditions字段
url_logic	String	url匹配逻辑
conditions	Array of Condition objects	条件
advanced	IgnoreAdvanced object	高级配置项
domain	Array of strings	防护域名或防护网站

表 4-495 Condition

参数	参数类型	描述
category	String	字段类型, 可选值有ip、url、params、cookie、header
contents	Array of strings	内容,数组长度限制为1, 内容格式根据字段类型变化, 例如, 字段类型为ip时, contents内容格式需为ip地址或ip地址段; 字段类型为url时, contents内容格式需为标准url格式; 字段类型为params、cookie、header时, 内容的格式不做限制
logic_operation	String	匹配逻辑, 匹配逻辑根据字段类型变化, 字段类型为ip时, 匹配逻辑支持(equal: 等于, not_equal: 不等于), 字段类型为url、header、params、cookie时, 匹配逻辑支持(equal: 等于, not_equal: 不等于, contain: 包含, not_contain: 不包含, prefix: 前缀为, not_prefix: 前缀不为, suffix: 后缀为, not_suffix: 后缀不为)
check_all_indexes_logic	Integer	该参数为预留参数, 可忽略。
index	String	字段类型为ip且子字段为客户端ip时, 不存在index参数; 子字段类型为X-Forwarded-For时, 值为x-forwarded-for, 字段类型为params、header、cookie并且子字段为自定义时, index的值为自定义子字段

表 4-496 IgnoreAdvanced

参数	参数类型	描述
index	String	字段类型，支持的字段类型有： params、cookie、header、body、 multipart。 <ul style="list-style-type: none">当选择“params”、“cookie”或者“header”字段时，可以配置“全部”或根据需求配置子字段当选择“body”或“multipart”字段时，可以配置“全部”
contents	Array of strings	指定字段类型的子字段，默认值为“全部”

状态码： 400

表 4-497 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-498 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-499 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-500 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-501 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-502 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的全局白名单(原误报屏蔽)防护规则。修改防护条件为url包含”/test“，需要屏蔽的规则为网站木马和其他类型攻击，配置高级配置项为全部参数。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}
```

```
{
  "domain" : [],
  "mode" : 1,
  "description" : "",
  "conditions" : [ {
    "category" : "url",
    "logic_operation" : "contain",
    "index" : null,
    "contents" : [ "/test" ]
  } ],
  "rule" : "webshell;vuln",
  "advanced" : {
    "index" : "params",
    "contents" : [ ]
  }
}
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "16e81d9a9e0244359204d7f00326ee4f",
  "policyid" : "0681f69f94ac408e9688373e45a61fdb",
  "timestamp" : 1679106005786,
  "description" : "",
  "status" : 1,
  "rule" : "webshell;vuln",
  "mode" : 1,
  "conditions" : [ {
    "category" : "url",
    "contents" : [ "/test" ],
    "logic_operation" : "contain"
  } ],
  "domain" : [],
  "advanced" : {
    "index" : "params",
    "contents" : [ ]
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的全局白名单(原误报屏蔽)防护规则。修改防护条件为url包含” /test “，需要屏蔽的规则为网站木马和其他类型攻击，配置高级配置项为全部参数。

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

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

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

public class UpdateIgnoreRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateIgnoreRuleRequest request = new UpdateIgnoreRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateIgnoreRuleRequestBody body = new UpdateIgnoreRuleRequestBody();
        IgnoreAdvanced advancedbody = new IgnoreAdvanced();
        advancedbody.withIndex("params");
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("/test");
        List<CreateCondition> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new CreateCondition()
                .withCategory("url")
                .withContents(listConditionsContents)
                .withLogicOperation("contain")
        );
        body.withDescription("");
        body.withAdvanced(advancedbody);
        body.withRule("webshell;vuln");
        body.withMode(1);
        body.withConditions(listbodyConditions);
        request.withBody(body);
        try {
            UpdateIgnoreRuleResponse response = client.updateIgnoreRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

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

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的全局白名单(原误报屏蔽)防护规则。修改防护条件为url包含”/test“，需要屏蔽的规则为网站木马和其他类型攻击，配置高级配置项为全部参数。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateIgnoreRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        advancedbody = IgnoreAdvanced(
            index="params"
        )
        listContentsConditions = [
            "/test"
        ]
        listConditionsbody = [
            CreateCondition(
                category="url",
                contents=listContentsConditions,
                logic_operation="contain"
            )
        ]
        request.body = UpdateIgnoreRuleRequestBody(
            description="",
            advanced=advancedbody,
            rule="webshell;vuln",
            mode=1,
            conditions=listConditionsbody
        )
        response = client.update_ignore_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
```

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

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的全局白名单(原误报屏蔽)防护规则。修改防护条件为url包含”/test“，需要屏蔽的规则为网站木马和其他类型攻击，配置高级配置项为全部参数。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateIgnoreRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    indexAdvanced := "params"
    advancedbody := &model.IgnoreAdvanced{
        Index: &indexAdvanced,
    }
    var listContentsConditions = []string{
        "/test",
    }
    var listConditionsbody = []model.CreateCondition{
        {
            Category: "url",
            Contents: listContentsConditions,
            LogicOperation: "contain",
        },
    },
    descriptionUpdateIgnoreRuleRequestBody := ""
    request.Body = &model.UpdateIgnoreRuleRequestBody{
        Description: &descriptionUpdateIgnoreRuleRequestBody,
        Advanced: advancedbody,
        Rule: "webshell;vuln",
        Mode: int32(1),
        Conditions: listConditionsbody,
    }
}
```

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.15 删除全局白名单(原误报屏蔽)防护规则

功能介绍

删除全局白名单(原误报屏蔽)防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}

表 4-503 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

参数	是否必选	参数类型	描述
rule_id	是	String	误报屏蔽规则id, 您可以通过查询误报屏蔽规则列表 (ListIgnoreRule) 接口的响应体中的id字段获取误报屏蔽规则id

表 4-504 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-505 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-506 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间戳
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启

参数	参数类型	描述
url	String	误报规则屏蔽路径，仅在mode为0的状态下有该字段
rule	String	<p>需要屏蔽的规则，可屏蔽一个或者多个，屏蔽多个时使用半角符;分隔</p> <ul style="list-style-type: none"> 当需要屏蔽某一条内置规则时，该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询；也可以在防护事件的事件详情中查询内置规则id 当需要屏蔽web基础防护某一类规则时，该参数值为需要屏蔽的web基础防护某一类规则名。其中，xss: xss攻击；webshell: 网站木马；vuln: 其他类型攻击；sqli: sql注入攻击；robot: 恶意爬虫；rfi: 远程文件包含；lfi: 本地文件包含；cmdi: 命令注入攻击 当需要屏蔽Web基础防护模块，该参数值为: all 当需要屏蔽规则为所有检测模块时，该参数值为: bypass
mode	Integer	版本号，0代表v1旧版本，1代表v2新版本；mode为0时，不存在conditions字段，存在url和url_logic字段；mode为1时，不存在url和url_logic字段，存在conditions字段
url_logic	String	url匹配逻辑
conditions	Array of Condition objects	条件
advanced	IgnoreAdvanced object	高级配置项
domains	Array of strings	防护域名或防护网站

表 4-507 Condition

参数	参数类型	描述
category	String	字段类型，可选值有ip、url、params、cookie、header

参数	参数类型	描述
contents	Array of strings	内容,数组长度限制为1,内容格式根据字段类型变化,例如,字段类型为ip时,contents内容格式需为ip地址或ip地址段;字段类型为url时,contents内容格式需为标准url格式;字段类型为params、cookie、header时,内容的格式不做限制
logic_operation	String	匹配逻辑,匹配逻辑根据字段类型变化,字段类型为ip时,匹配逻辑支持(equal:等于,not_equal:不等于),字段类型为url、header、params、cookie时,匹配逻辑支持(equal:等于,not_equal:不等于,contain:包含,not_contain:不包含,prefix:前缀为,not_prefix:前缀不为,suffix:后缀为,not_suffix:后缀不为)
check_all_indexes_logic	Integer	该参数为预留参数,可忽略。
index	String	字段类型为ip且子字段为客户端ip时,不存在index参数;子字段类型为X-Forwarded-For时,值为x-forwarded-for,字段类型为params、header、cookie并且子字段为自定义时,index的值为自定义子字段

表 4-508 IgnoreAdvanced

参数	参数类型	描述
index	String	字段类型,支持的字段类型有: params、cookie、header、body、multipart。 <ul style="list-style-type: none">当选择“params”、“cookie”或者“header”字段时,可以配置“全部”或根据需求配置子字段当选择“body”或“multipart”字段时,可以配置“全部”
contents	Array of strings	指定字段类型的子字段,默认值为“全部”

状态码: 400

表 4-509 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-510 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-511 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-512 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-513 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-514 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的全局白名单(原误报屏蔽)防护规则

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/ignore/{rule_id}
```

响应示例

状态码： 200

请求成功

```
{  
  "id": "40484384970948d79ffe4e4ae1fc54d",  
  "policyid": "f385eceedf7c4c34a4d1def19eafbe85",  
}
```

```
"timestamp" : 1650512535222,
"description" : "demo",
"status" : 1,
"rule" : "091004",
"mode" : 1,
"conditions" : [ {
  "category" : "ip",
  "contents" : [ "x.x.x.x" ],
  "logic_operation" : "equal"
} ],
"domain" : [ "we.test.418lab.cn" ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeletelgnoreRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeletelgnoreRuleRequest request = new DeletelgnoreRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeletelgnoreRuleResponse response = client.deletelgnoreRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = WafClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(WafRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = DeleteIgnoreRuleRequest()  
        request.policy_id = "{policy_id}"  
        request.rule_id = "{rule_id}"  
        response = client.delete_ignore_rule(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).
```

```
Build()

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.DeletelgnoreRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.DeletelgnoreRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.16 查询黑白名单规则列表

功能介绍

查询黑白名单规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/whiteblackip

表 4-515 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-516 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
page	否	Integer	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。
name	否	String	黑白名单规则名称

请求参数

表 4-517 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-518 响应 Body 参数

参数	参数类型	描述
total	Integer	黑白名单规则条数
items	Array of WhiteBlackIpResponseBody objects	黑白名单规则列表信息
size	Integer	ip地址总数

表 4-519 WhiteBlackIpResponseBody

参数	参数类型	描述
id	String	规则id
name	String	黑白名单规则名称
policyid	String	策略id
timestamp	Long	创建规则的时间戳（毫秒）
description	String	规则描述
status	Integer	规则状态，0：关闭，1：开启
addr	String	Ip/Ip段
white	Integer	防护动作： <ul style="list-style-type: none">• 0拦截• 1放行• 2仅记录
ip_group	ip_group object	Ip地址组
time_mode	String	生效模式，默认为permanent（立即生效）
start	Long	规则生效的开始时间，生效模式为自定义时，此字段才有效
terminal	Long	规则生效的结束时间，生效模式为自定义时，此字段才有效

表 4-520 Ip_group

参数	参数类型	描述
id	String	Ip地址组id，在新增Ip地址组时系统自动生成的唯一标识

参数	参数类型	描述
name	String	Ip地址组名
size	Long	Ip地址组中包含Ip/Ip段的数量

状态码： 400

表 4-521 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-522 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-523 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-524 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-525 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-526 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的黑白名单规则列表

GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/whiteblackip?enterprise_project_id=0

响应示例

状态码: 200

OK

```
{
  "total": 1,
  "items": [ {
    "id": "3c96caf769ca4f57814fcf4259ea89a1",
    "policyid": "4dddfd44fc89453e9fd9cd6bfdc39db2",
    "name": "hkhtest",
    "timestamp": 1650362891844,
    "description": "demo",
    "status": 1,
    "addr": "x.x.x.x",
    "white": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListWhiteblackipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListWhiteblackipRuleRequest request = new ListWhiteblackipRuleRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListWhiteblackipRuleResponse response = client.listWhiteblackipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
```

```
        e.printStackTrace();
    } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListWhiteblackipRuleRequest()
        request.policy_id = "{policy_id}"
        response = client.list_whiteblackip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

```
projectId := "{project_id}"

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.17 创建黑白名单规则

功能介绍

创建黑白名单规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/whiteblackip

表 4-527 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-528 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-529 请求 Header 参数

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

表 4-530 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	规则名称只能由字母、数字、-、_和.组成, 长度不能超过64个字符
addr	否	String	黑白名单ip地址, 需要输入标准的ip地址或地址段, 例如: 42.123.120.66或 42.123.120.0/16
description	否	String	黑白名单规则描述

参数	是否必选	参数类型	描述
white	是	Integer	防护动作： <ul style="list-style-type: none"> • 0 拦截 • 1 放行 • 2 仅记录
ip_group_id	否	String	创建的Ip地址组id, 该参数与addr参数只能使用一个; Ip地址组可在控制台中对象管理->地址组管理中添加。
time_mode	否	String	生效模式, 默认为permanent (立即生效), 创建自定义生效规则时请输入: customize
start	否	Integer	规则生效开始时间, 生效模式为自定义时, 此字段才有效, 请输入时间戳
terminal	否	Integer	规则生效结束时间, 生效模式为自定义时, 此字段才有效, 请输入时间戳

响应参数

状态码: 200

表 4-531 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	黑白名单规则名称
policyid	String	策略id
addr	String	黑白名单ip地址, 需要输入标准的ip地址或地址段, 例如: 42.123.120.66或42.123.120.0/16
white	Integer	防护动作： <ul style="list-style-type: none"> • 0 拦截 • 1 放行 • 2 仅记录
time_mode	String	生效模式, 默认为permanent (立即生效)

参数	参数类型	描述
start	Long	规则生效的开始时间，生效模式为自定义时，此字段才有效
terminal	Long	规则生效的结束时间，生效模式为自定义时，此字段才有效
ip_group	Ip_group object	Ip地址组
status	Integer	规则状态，0：关闭，1：开启
description	String	规则描述
timestamp	Long	创建规则的时间戳,13位毫秒时间戳

表 4-532 Ip_group

参数	参数类型	描述
id	String	Ip地址组id，在新增Ip地址组时系统自动生成的唯一标识
name	String	Ip地址组名
size	Long	Ip地址组中包含Ip/Ip段的数量

状态码： 400

表 4-533 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-534 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-535 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-536 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-537 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-538 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id, 防护策略id为policy_id的防护策略下创建黑白名单规则, 规则名称为"demo", 防护动作为"拦截", 描述为'demo', 黑白名单ip地址为"x.x.x.x"。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/whiteblackip?enterprise_project_id=0
{
  "name": "demo",
  "white": 0,
  "description": "demo",
  "addr": "x.x.x.x"
}
```

响应示例

状态码: 200

OK

```
{
  "id": "5d43af25404341058d5ab17b7ba78b56",
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name": "demo",
  "timestamp": 1650531872900,
  "description": "demo",
  "status": 1,
  "addr": "x.x.x.x",
  "white": 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建黑白名单规则，规则名称为"demo"，防护动作为"拦截"，描述为“demo”，黑白名单ip地址为“X.X.X.X”。

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

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

public class CreateWhiteblackipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateWhiteblackipRuleRequest request = new CreateWhiteblackipRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateWhiteBlackIpRuleRequestBody body = new CreateWhiteBlackIpRuleRequestBody();
        body.withWhite(0);
        body.withDescription("demo");
        body.withAddr("x.x.x.x");
        body.withName("demo");
        request.withBody(body);
        try {
            CreateWhiteblackipRuleResponse response = client.createWhiteblackipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建黑白名单规则，规则名称为"demo"，防护动作为"拦截"，描述为'demo'，黑白名单ip地址为"x.x.x.x"。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateWhiteblackipRuleRequest()
        request.policy_id = "{policy_id}"
        request.body = CreateWhiteBlackIpRuleRequestBody(
            white=0,
            description="demo",
            addr="x.x.x.x",
            name="demo"
        )
        response = client.create_whiteblackip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建黑白名单规则，规则名称为"demo"，防护动作为"拦截"，描述为'demo'，黑白名单ip地址为"x.x.x.x"。

```
package main

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

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateWhiteblackipRuleRequest{}
request.PolicyId = "{policy_id}"
descriptionCreateWhiteBlackIpRuleRequestBody := "demo"
addrCreateWhiteBlackIpRuleRequestBody := "x.x.x.x"
request.Body = &model.CreateWhiteBlackIpRuleRequestBody{
    White: int32(0),
    Description: &descriptionCreateWhiteBlackIpRuleRequestBody,
    Addr: &addrCreateWhiteBlackIpRuleRequestBody,
    Name: "demo",
}
response, err := client.CreateWhiteblackipRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.18 查询黑白名单防护规则

功能介绍

查询黑白名单防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}

表 4-539 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	黑白名单规则id，您可以通过调用查询黑白名单规则列表（ListWhiteblackipRule）获取黑白名单规则id

表 4-540 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-541 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-542 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	黑白名单规则名称
policyid	String	策略id
addr	String	黑白名单ip地址，需要输入标准的ip地址或地址段，例如：42.123.120.66或42.123.120.0/16
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
time_mode	String	生效模式，默认为permanent（立即生效）
start	Long	规则生效的开始时间，生效模式为自定义时，此字段才有效
terminal	Long	规则生效的结束时间，生效模式为自定义时，此字段才有效
ip_group	Ip_group object	Ip地址组
status	Integer	规则状态，0：关闭，1：开启
description	String	规则描述
timestamp	Long	创建规则的时间戳,13位毫秒时间戳

表 4-543 Ip_group

参数	参数类型	描述
id	String	Ip地址组id，在新增Ip地址组时系统自动生成的唯一标识
name	String	Ip地址组名

参数	参数类型	描述
size	Long	Ip地址组中包含Ip/Ip段的数量

状态码： 400

表 4-544 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-545 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-546 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-547 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-548 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-549 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "5d43af25404341058d5ab17b7ba78b56",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name" : "demo",
  "timestamp" : 1650531872900,
  "description" : "demo",
  "status" : 1,
  "addr" : "1.1.1.2",
  "white" : 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowWhiteBlackIpRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowWhiteBlackIpRuleRequest request = new ShowWhiteBlackIpRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowWhiteBlackIpRuleResponse response = client.showWhiteBlackIpRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```



```
e.printStackTrace();
System.out.println(e.getStatusCode());
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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowWhiteBlackIpRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_white_black_ip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowWhiteBlackIpRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.ShowWhiteBlackIpRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.19 更新黑白名单防护规则

功能介绍

更新黑白名单防护规则，可以更新ip/ip段以及防护动作等信息

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}

表 4-550 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	黑白名单规则id, 您可以调用查询黑白名单规则列表 (ListWhiteblackipRule) 获取黑白名单规则id

表 4-551 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-552 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)。
Content-Type	是	String	内容类型

表 4-553 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	黑白名单规则名称
addr	否	String	黑白名单ip地址, 需要输入标准的ip地址或地址段, 例如: 42.123.120.66或 42.123.120.0/16

参数	是否必选	参数类型	描述
description	否	String	黑白名单规则描述
white	是	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
ip_group_id	否	String	创建的Ip地址组id, 该参数与addr参数使用一个即可; Ip地址组可在控制台中对象管理->地址组管理中添加。
time_mode	否	String	生效模式, 默认为permanent (立即生效), 创建自定义生效规则时请输入: customize
start	否	Integer	规则生效开始时间, 生效模式为自定义时, 此字段才有效, 请输入时间戳
terminal	否	Integer	规则生效结束时间, 生效模式为自定义时, 此字段才有效, 请输入时间戳

响应参数

状态码: 200

表 4-554 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	黑白名单规则名称
policyid	String	策略id
addr	String	黑白名单Ip/IP段
description	String	黑白名单规则描述
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
time_mode	String	生效模式, 默认为permanent (立即生效)

参数	参数类型	描述
start	Long	规则生效的开始时间，生效模式为自定义时，此字段才有效
terminal	Long	规则生效的结束时间，生效模式为自定义时，此字段才有效
ip_group	ip_group object	Ip地址组

表 4-555 Ip_group

参数	参数类型	描述
id	String	Ip地址组id，在新增Ip地址组时系统自动生成的唯一标识
name	String	Ip地址组名
size	Long	Ip地址组中包含Ip/Ip段的数量

状态码： 400

表 4-556 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-557 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-558 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-559 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-560 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-561 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。更新规则名称为"demo"，防护动作为"拦截"，描述为“demo”，黑白名单ip地址为“1.1.1.2”。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}?enterprise_project_id=0
{
  "name": "demo",
  "white": 0,
  "description": "demo",
  "addr": "1.1.1.2"
}
```

响应示例

状态码：200

请求成功

```
{
  "id": "5d43af25404341058d5ab17b7ba78b56",
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name": "demo",
  "description": "demo",
  "addr": "1.1.1.2",
  "white": 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。更新规则名称为"demo"，防护动作为"拦截"，描述为“demo”，黑白名单ip地址为“1.1.1.2”。

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

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

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateWhiteblackipRuleRequest request = new UpdateWhiteblackipRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateWhiteBlackIpRuleRequestBody body = new UpdateWhiteBlackIpRuleRequestBody();
        body.withWhite(0);
        body.withDescription("demo");
        body.withAddr("1.1.1.2");
        body.withName("demo");
        request.withBody(body);
        try {
            UpdateWhiteblackipRuleResponse response = client.updateWhiteblackipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。更新规则名称为"demo"，防护动作为"拦截"，描述为“demo”，黑白名单ip地址为“1.1.1.2”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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



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

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = UpdateWhiteblackipRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    request.body = UpdateWhiteBlackIpRuleRequestBody(
        white=0,
        description="demo",
        addr="1.1.1.2",
        name="demo"
    )
    response = client.update_whiteblackip_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。更新规则名称为"demo"，防护动作为"拦截"，描述为'demo'，黑白名单ip地址为"1.1.1.2"。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateWhiteblackipRuleRequest{}
    request.PolicyId = "{policy_id}"
```

```
request.RuleId = "{rule_id}"
descriptionUpdateWhiteBlackIpRuleRequestBody:= "demo"
addrUpdateWhiteBlackIpRuleRequestBody:= "1.1.1.2"
request.Body = &model.UpdateWhiteBlackIpRuleRequestBody{
    White: int32(0),
    Description: &descriptionUpdateWhiteBlackIpRuleRequestBody,
    Addr: &addrUpdateWhiteBlackIpRuleRequestBody,
    Name: "demo",
}
response, err := client.UpdateWhiteblackipRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.20 查询全局白名单(原误报屏蔽)规则列表

功能介绍

查询全局白名单(原误报屏蔽)规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/ignore

表 4-562 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的全局白名单规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-563 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。
pagesize	否	Integer	分页查询时, 每页包含多少条结果。范围1-100, 默认值为10, 表示每页包含10条结果。

请求参数

表 4-564 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-565 响应 Body 参数

参数	参数类型	描述
total	Integer	该策略下全局白名单规则数量
items	Array of IgnoreRuleBody objects	全局白名单规则信息数组

表 4-566 IgnoreRuleBody

参数	参数类型	描述
id	String	规则id
policyid	String	该规则属于的防护策略的id
timestamp	Long	创建规则的时间戳
description	String	规则描述
status	Integer	规则状态, 0: 关闭, 1: 开启
url	String	误报规则屏蔽路径, 仅在mode为0的状态下有该字段
rule	String	需要屏蔽的规则, 可屏蔽一个或者多个, 屏蔽多个时使用半角符;分隔 <ul style="list-style-type: none">当需要屏蔽某一条内置规则时, 该参数值为该内置规则id,可以在Web应用防火墙控制台的防护策略->策略名称->Web基础防护的高级设置->防护规则中查询; 也可以在防护事件的事件详情中查询内置规则id当需要屏蔽web基础防护某一类规则时, 该参数值为需要屏蔽的web基础防护某一类规则名。其中, xss: xss攻击; webshell: 网站木马; vuln: 其他类型攻击; sql: sql注入攻击; robot: 恶意爬虫; rfi: 远程文件包含; lfi: 本地文件包含; cmd: 命令注入攻击当需要屏蔽Web基础防护模块, 该参数值为: all当需要屏蔽规则为所有检测模块时, 该参数值为: bypass

参数	参数类型	描述
mode	Integer	版本号, 0代表v1旧版本, 1代表v2新版本; mode为0时, 不存在conditions字段, 存在url和url_logic字段; mode为1时, 不存在url和url_logic字段, 存在conditions字段
url_logic	String	匹配逻辑支持 (equal: 等于, not_equal: 不等于, contain: 包含, not_contain: 不包含, prefix: 前缀为, not_prefix: 前缀不为, suffix: 后缀为, not_suffix: 后缀不为)
conditions	Array of Condition objects	条件列表
domain	Array of strings	防护域名或防护网站
advanced	IgnoreAdvanced object	高级配置项

表 4-567 Condition

参数	参数类型	描述
category	String	字段类型, 可选值有ip、url、params、cookie、header
contents	Array of strings	内容,数组长度限制为1, 内容格式根据字段类型变化, 例如, 字段类型为ip时, contents内容格式需为ip地址或ip地址段; 字段类型为url时, contents内容格式需为标准url格式; 字段类型为params、cookie、header时, 内容的格式不做限制
logic_operation	String	匹配逻辑, 匹配逻辑根据字段类型变化, 字段类型为ip时, 匹配逻辑支持 (equal: 等于, not_equal: 不等于), 字段类型为url、header、params、cookie时, 匹配逻辑支持 (equal: 等于, not_equal: 不等于, contain: 包含, not_contain: 不包含, prefix: 前缀为, not_prefix: 前缀不为, suffix: 后缀为, not_suffix: 后缀不为)
check_all_indexes_logic	Integer	该参数为预留参数, 可忽略。

参数	参数类型	描述
index	String	字段类型为ip且子字段为客户端ip时，不存在index参数；子字段类型为X-Forwarded-For时，值为x-forwarded-for，字段类型为params、header、cookie并且子字段为自定义时，index的值为自定义子字段

表 4-568 IgnoreAdvanced

参数	参数类型	描述
index	String	字段类型，支持的字段类型有：params、cookie、header、body、multipart。 <ul style="list-style-type: none">当选择“params”、“cookie”或者“header”字段时，可以配置“全部”或根据需求配置子字段当选择“body”或“multipart”字段时，可以配置“全部”
contents	Array of strings	指定字段类型的子字段，默认值为“全部”

状态码： 400

表 4-569 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码(加密)后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-570 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-571 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-572 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 403

表 4-573 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-574 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-575 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-576 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的全局白名单(原误报屏蔽)规则列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/ignore?
enterprise_project_id=0&page=1&pagesize=10
```

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "items": [ {
    "id": "40484384970948d79ffe4e4ae1fc54d",
    "policyid": "f385eceedf7c4c34a4d1def19eafbe85",
    "timestamp": 1650512535222,
    "description": "demo",
    "status": 1,
    "rule": "091004",
    "mode": 1,
    "conditions": [ {
      "category": "ip",
      "contents": [ "x.x.x.x" ],
      "logic_operation": "equal"
    } ],
    "domain": [ "we.test.418lab.cn" ]
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListIgnoreRuleSolution {

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

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

```
        .withSk(sk);

    WafClient client = WafClient.newBuilder()
        .withCredential(auth)
        .withRegion(WafRegion.valueOf("<YOUR REGION>"))
        .build();
    ListIgnoreRuleRequest request = new ListIgnoreRuleRequest();
    request.withPolicyId("{policy_id}");
    try {
        ListIgnoreRuleResponse response = client.listIgnoreRule(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListIgnoreRuleRequest()
        request.policy_id = "{policy_id}"
        response = client.list_ignore_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
403	资源配额不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.21 删除黑白名单防护规则

功能介绍

删除黑白名单防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}

表 4-577 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	黑白名单规则id，您可以通过调用查询黑白名单规则列表（ListWhiteblackipRule）获取黑白名单规则id

表 4-578 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-579 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-580 响应 Body 参数

参数	参数类型	描述
id	String	黑白名单规则id
policyid	String	策略id
name	String	黑白名单规则名称
timestamp	Long	删除规则时间， 13位毫秒时间戳
description	String	描述
status	Integer	规则状态， 0： 关闭， 1： 开启
addr	String	黑白名单ip地址， 标准的ip地址或地址段， 例如： 42.123.120.66或42.123.120.0/16
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
time_mode	String	生效模式， 默认为permanent（立即生效）
start	Long	规则生效的开始时间， 生效模式为自定义时， 此字段才有效
terminal	Long	规则生效的结束时间， 生效模式为自定义时， 此字段才有效
ip_group	Ip_group object	Ip地址组

表 4-581 Ip_group

参数	参数类型	描述
id	String	Ip地址组id， 在新增Ip地址组时系统自动生成的唯一标识
name	String	Ip地址组名

参数	参数类型	描述
size	Long	Ip地址组中包含Ip/Ip段的数量

状态码： 400

表 4-582 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-583 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-584 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-585 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-586 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-587 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id，防护策略id为policy_id，规则id为rule_id的黑白名单规则。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/whiteblackip/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "5d43af25404341058d5ab17b7ba78b56",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name" : "demo",
  "timestamp" : 1650531872900,
  "description" : "demo",
  "status" : 1,
  "addr" : "1.1.1.2",
  "white" : 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteWhiteBlackIpRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteWhiteBlackIpRuleRequest request = new DeleteWhiteBlackIpRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeleteWhiteBlackIpRuleResponse response = client.deleteWhiteBlackIpRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```



```
e.printStackTrace();
System.out.println(e.getStatusCode());
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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteWhiteBlackIpRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_white_black_ip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.DeleteWhiteBlackIpRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.DeleteWhiteBlackIpRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.22 查询 JS 脚本反爬虫规则列表

功能介绍

查询JS脚本反爬虫规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/anticrawler

表 4-588 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防护规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-589 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
offset	是	Integer	偏移量, 表示查询该偏移量之后的记录。
limit	是	Integer	查询返回记录的数量限制。
type	否	String	JS脚本反爬虫规则防护模式 <ul style="list-style-type: none">• anticrawler_except_url: 防护所有路径模式, 在该模式下, 查询的JS脚本反爬虫规则为排除的防护路径规则• anticrawler_specific_url: 防护指定路径模式, 在该模式下, 查询的JS脚本反爬虫规则为指定要防护的路径规则

请求参数

表 4-590 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-591 响应 Body 参数

参数	参数类型	描述
total	Integer	该策略下反爬虫规则数量
items	Array of AnticrawlerRule objects	反爬虫规则列表

表 4-592 AnticrawlerRule

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
conditions	Array of AnticrawlerCondition objects	匹配条件列表
name	String	规则名称
type	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url
timestamp	Long	创建规则时间戳
status	Integer	规则状态，0：关闭，1：开启
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-593 AnticrawlerCondition

参数	参数类型	描述
category	String	字段类型

参数	参数类型	描述
logic_operation	String	条件列表匹配逻辑, 包括: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时, 需要传该参数。
value_list_id	String	引用表 id。当 logic_operation 参数以 any 或者 all 结尾时, 需要传该参数。此外, 引用表类型要与 category 类型保持一致。

状态码: 400

表 4-594 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码, 可参考 STS5 联调自验证。IAM5 鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5 鉴权错误时才会返回此字段。

表 4-595 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-596 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-597 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-598 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-599 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的JS脚本反爬虫规则列表

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler?
```

响应示例

状态码： 200

ok

```
{
  "total": 1,
  "items": [ {
    "id": "c06ec2e5d93241a694fcd9e0312657a1",
    "policyid": "0681f69f94ac408e9688373e45a61fdb",
    "name": "test",
    "timestamp": 1678931359146,
    "status": 1,
    "type": "anticrawler_except_url",
    "conditions": [ {
      "category": "url",
      "contents": [ "/test" ],
      "logic_operation": "contain"
    } ],
    "priority": 50
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListAnticrawlerRulesSolution {

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

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

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListAnticrawlerRulesRequest request = new ListAnticrawlerRulesRequest();
request.withPolicyId("{policy_id}");
try {
    ListAnticrawlerRulesResponse response = client.listAnticrawlerRules(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListAnticrawlerRulesRequest()
        request.policy_id = "{policy_id}"
        response = client.list_anticrawler_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.23 更新 JS 脚本反爬虫规则防护模式

功能介绍

更新JS脚本反爬虫规则防护模式，在创建JS脚本反爬虫规则前，需要调用该接口指定JS脚本反爬虫规则防护模式。

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/anticrawler

表 4-600 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-601 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-602 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)。
Content-Type	是	String	内容类型

表 4-603 请求 Body 参数

参数	是否必选	参数类型	描述
anticrawler_type	是	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url

响应参数

状态码： 200

表 4-604 响应 Body 参数

参数	参数类型	描述
anticrawler_type	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url

状态码： 400

表 4-605 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-606 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-607 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-608 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-609 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-610 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的的JS脚本反爬虫规则, 更新JS脚本反爬虫规则类型为排除防护路径。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler?
```

```
{  
  "anticrawler_type": "anticrawler_except_url"  
}
```

响应示例

状态码： 200

ok

```
{  
  "anticrawler_type": "anticrawler_except_url"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的的JS脚本反爬虫规则，更新JS脚本反爬虫规则类型为排除防护路径。

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

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

public class UpdateAnticrawlerRuleTypeSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateAnticrawlerRuleTypeRequest request = new UpdateAnticrawlerRuleTypeRequest();
        request.withPolicyId("{policy_id}");
        UpdateAnticrawlerRuleTypeRequestbody body = new UpdateAnticrawlerRuleTypeRequestbody();
        body.withAnticrawlerType("anticrawler_except_url");
        request.withBody(body);
        try {
            UpdateAnticrawlerRuleTypeResponse response = client.updateAnticrawlerRuleType(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的的JS脚本反爬虫规则，更新JS脚本反爬虫规则类型为排除防护路径。

```
# coding: utf-8

import os
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateAnticrawlerRuleTypeRequest()
        request.policy_id = "{policy_id}"
        request.body = UpdateAnticrawlerRuleTypeRequestbody(
            anticrawler_type="anticrawler_except_url"
        )
        response = client.update_anticrawler_rule_type(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的的JS脚本反爬虫规则，更新JS脚本反爬虫规则类型为排除防护路径。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
```

```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.UpdateAnticrawlerRuleTypeRequest{}
request.PolicyId = "{policy_id}"
request.Body = &model.UpdateAnticrawlerRuleTypeRequestbody{
    AnticrawlerType: "anticrawler_except_url",
}
response, err := client.UpdateAnticrawlerRuleType(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.24 创建 JS 脚本反爬虫规则

功能介绍

创建JS脚本反爬虫规则，在调用此接口创建防护规则前，需要调用更新JS脚本反爬虫规则防护模式（UpdateAnticrawlerRuleType）接口指定防护模式

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/anticrawler

表 4-611 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防护规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-612 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-613 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)。
Content-Type	是	String	内容类型

表 4-614 请求 Body 参数

参数	是否必选	参数类型	描述
conditions	是	Array of AnticrawlerCondition objects	匹配条件列表
name	是	String	规则名称

参数	是否必选	参数类型	描述
type	是	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url
priority	是	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-615 AnticrawlerCondition

参数	是否必选	参数类型	描述
category	否	String	字段类型
logic_operation	否	String	条件列表匹配逻辑, 包括： contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	否	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时，需要传该参数。
value_list_id	否	String	引用表id。当 logic_operation 参数以 any 或者 all 结尾时，需要传该参数。此外，引用表类型要与 category 类型保持一致。

响应参数

状态码： 200

表 4-616 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id

参数	参数类型	描述
conditions	Array of AnticrawlerCondition objects	匹配条件列表
name	String	规则名称
type	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url
timestamp	Long	创建规则时间戳
status	Integer	规则状态，0：关闭，1：开启
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-617 AnticrawlerCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	条件列表匹配逻辑，包括：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时，需要传该参数。
value_list_id	String	引用表id。当 logic_operation 参数以 any 或者 all 结尾时，需要传该参数。此外，引用表类型要与 category 类型保持一致。

状态码： 400

表 4-618 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-619 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-620 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-621 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-622 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-623 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id, 防护策略id为policy_id的防护策略下创建JS脚本反爬虫规则, 规则名称为“test66”, JS脚本反爬虫规则类型为排除防护路径, 匹配条为url包含“/test66”, 执行该规则的优先级为50。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler?
{
  "name": "test66",
  "type": "anticrawler_except_url",
  "conditions": [ {
    "category": "url",
    "logic_operation": "contain",
    "contents": [ "/test66" ]
  } ],
  "priority": 50
}
```

响应示例

状态码： 200

ok

```
{
  "id": "7e7983bf2c9c41029d642bcbf819346d",
  "policyid": "0681f69f94ac408e9688373e45a61fdb",
}
```

```
"name" : "test66",
"timestamp" : 1678931492172,
"status" : 1,
"type" : "anticrawler_except_url",
"conditions" : [ {
  "category" : "url",
  "contents" : [ "/test66" ],
  "logic_operation" : "contain"
} ],
"priority" : 50
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

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

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

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

public class CreateAnticrawlerRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateAnticrawlerRuleRequest request = new CreateAnticrawlerRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateAnticrawlerRuleRequestbody body = new CreateAnticrawlerRuleRequestbody();
        List<String> listConditionsContents = new ArrayList<>();
        listConditionsContents.add("/test66");
        List<AnticrawlerCondition> listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new AnticrawlerCondition()
                .withCategory(AnticrawlerCondition.CategoryEnum.fromValue("url"))
                .withLogicOperation("contain")
        );
    }
}
```

```
        .withContents(listConditionsContents)
    );
    body.withPriority(50);
    body.withType("anticrawler_except_url");
    body.withName("test66");
    body.withConditions(listbodyConditions);
    request.withBody(body);
    try {
        CreateAnticrawlerRuleResponse response = client.createAnticrawlerRule(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateAnticrawlerRuleRequest()
        request.policy_id = "{policy_id}"
        listContentsConditions = [
            "/test66"
        ]
        listConditionsbody = [
            AnticrawlerCondition(
                category="url",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
```

```
request.body = CreateAnticrawlerRuleRequestbody(  
    priority=50,  
    type="anticrawler_except_url",  
    name="test66",  
    conditions=listConditionsbody  
)  
response = client.create_anticrawler_rule(request)  
print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.CreateAnticrawlerRuleRequest{}  
    request.PolicyId = "{policy_id}"  
    var listContentsConditions = []string{  
        "/test66",  
    }  
    categoryConditions:= model.GetAnticrawlerConditionCategoryEnum().URL  
    logicOperationConditions:= "contain"  
    var listConditionsbody = []model.AnticrawlerCondition{  
        {  
            Category: &categoryConditions,  
            LogicOperation: &logicOperationConditions,  
            Contents: &listContentsConditions,  
        },  
    }  
    request.Body = &model.CreateAnticrawlerRuleRequestbody{  
        Priority: int32(50),
```



```
Type: "anticrawler_except_url",
Name: "test66",
Conditions: listConditionsbody,
}
response, err := client.CreateAnticrawlerRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.25 查询 JS 脚本反爬虫防护规则

功能介绍

根据Id查询JS脚本反爬虫防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}

表 4-624 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

参数	是否必选	参数类型	描述
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防护规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	规则id

表 4-625 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-626 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-627 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
conditions	Array of AnticrawlerCondition objects	匹配条件列表

参数	参数类型	描述
name	String	规则名称
type	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url
timestamp	Long	创建规则时间戳
status	Integer	规则状态，0：关闭，1：开启
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-628 AnticrawlerCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	条件列表匹配逻辑，包括：contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时，需要传该参数。
value_list_id	String	引用表 id。当 logic_operation 参数以 any 或者 all 结尾时，需要传该参数。此外，引用表类型要与 category 类型保持一致。

状态码： 400

表 4-629 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-630 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-631 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-632 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-633 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-634 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的JS脚本反爬虫防护规则

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}?
```

响应示例

状态码： 200

ok

```
{
  "id": "7e7983bf2c9c41029d642bcbf819346d",
  "policyid": "0681f69f94ac408e9688373e45a61fdb",
  "name": "test66",
  "timestamp": 1678931492172,
  "status": 1,
  "type": "anticrawler_except_url",
  "conditions": [ {
    "category": "url",
    "contents": [ "/test66" ],
    "logic_operation": "contain"
  } ],
  "priority": 50
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowAnticrawlerRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowAnticrawlerRuleRequest request = new ShowAnticrawlerRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowAnticrawlerRuleResponse response = client.showAnticrawlerRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *
```

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowAnticrawlerRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_anticrawler_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAnticrawlerRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowAnticrawlerRule(request)
    if err == nil {
```

```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.26 更新 JS 脚本反爬虫防护规则

功能介绍

更新JS脚本反爬虫防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}

表 4-635 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

参数	是否必选	参数类型	描述
rule_id	是	String	规则id

表 4-636 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-637 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)。
Content-Type	是	String	内容类型

表 4-638 请求 Body 参数

参数	是否必选	参数类型	描述
conditions	是	Array of AnticrawlerCondition objects	匹配条件列表
name	是	String	规则名称
type	是	String	JS脚本反爬虫规则类型，指定防护路径： anticrawler_specific_url 排除防护路径： anticrawler_except_url
priority	是	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-639 AnticrawlerCondition

参数	是否必选	参数类型	描述
category	否	String	字段类型
logic_operation	否	String	条件列表匹配逻辑, 包括: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	否	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时, 需要传该参数。
value_list_id	否	String	引用表 id。当 logic_operation 参数以 any 或者 all 结尾时, 需要传该参数。此外, 引用表类型要与 category 类型保持一致。

响应参数

状态码: 200

表 4-640 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
conditions	Array of AnticrawlerCondition objects	匹配条件列表
name	String	规则名称
type	String	JS脚本反爬虫规则类型, 指定防护路径: anticrawler_specific_url 排除防护路径: anticrawler_except_url
timestamp	Long	创建规则时间戳
status	Integer	规则状态, 0: 关闭, 1: 开启
priority	Integer	执行该规则的优先级, 值越小, 优先级越高, 值相同时, 规则创建时间早, 优先级越高。取值范围: 0到1000。

表 4-641 AnticrawlerCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	条件列表匹配逻辑, 包括: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时, 需要传该参数。
value_list_id	String	引用表 id。当 logic_operation 参数以 any 或者 all 结尾时, 需要传该参数。此外, 引用表类型要与 category 类型保持一致。

状态码: 400

表 4-642 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码, 可参考 STS5 联调自验证。IAM5 鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5 鉴权错误时才会返回此字段。

表 4-643 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-644** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-645 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-646** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-647 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}?
```

```
{
  "name": "test66",
  "type": "anticrawler_except_url",
  "conditions": [ {
    "category": "url",
    "logic_operation": "contain",
    "contents": [ "/test66" ]
  } ],
  "priority": 50
}
```

响应示例

状态码： 200

ok

```
{
  "id": "7e7983bf2c9c41029d642bcfb819346d",
  "policyid": "0681f69f94ac408e9688373e45a61fdb",
  "name": "test66",
  "timestamp": 1678931492172,
  "status": 1,
  "type": "anticrawler_except_url",
  "conditions": [ {
    "category": "url",
    "contents": [ "/test66" ],
    "logic_operation": "contain"
  } ],
  "priority": 50
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

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

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

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

public class UpdateAnticrawlerRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateAnticrawlerRuleRequest request = new UpdateAnticrawlerRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateAnticrawlerRuleRequestBody body = new UpdateAnticrawlerRuleRequestBody();
        List listConditionsContents = new ArrayList<>();
        listConditionsContents.add("/test66");
        List listbodyConditions = new ArrayList<>();
        listbodyConditions.add(
            new AnticrawlerCondition()
                .withCategory(AnticrawlerCondition.CategoryEnum.fromValue("url"))
                .withLogicOperation("contain")
                .withContents(listConditionsContents)
        );
        body.withPriority(50);
        body.withType("anticrawler_except_url");
        body.withName("test66");
        body.withConditions(listbodyConditions);
        request.withBody(body);
        try {
            UpdateAnticrawlerRuleResponse response = client.updateAnticrawlerRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的的JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateAnticrawlerRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        listContentsConditions = [
            "/test66"
        ]
        listConditionsbody = [
            AnticrawlerCondition(
                category="url",
                logic_operation="contain",
                contents=listContentsConditions
            )
        ]
        request.body = UpdateAnticrawlerRuleRequestBody(
            priority=50,
            type="anticrawler_except_url",
            name="test66",
            conditions=listConditionsbody
        )
        response = client.update_anticrawler_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的的JS脚本反爬虫规则，规则名称为“test66”，JS脚本反爬虫规则类型为排除防护路径，匹配条为url包含“/test66”，执行该规则的优先级为50。

```
package main
```

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAnticrawlerRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    var listContentsConditions = []string{
        "/test66",
    }
    categoryConditions:= model.GetAnticrawlerConditionCategoryEnum().URL
    logicOperationConditions:= "contain"
    var listConditionsbody = []model.AnticrawlerCondition{
        {
            Category: &categoryConditions,
            LogicOperation: &logicOperationConditions,
            Contents: &listContentsConditions,
        },
    }
    request.Body = &model.UpdateAnticrawlerRuleRequestBody{
        Priority: int32(50),
        Type: "anticrawler_except_url",
        Name: "test66",
        Conditions: listConditionsbody,
    }
    response, err := client.UpdateAnticrawlerRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.27 删除 JS 脚本反爬虫防护规则

功能介绍

删除JS脚本反爬虫防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}

表 4-648 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的防护规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	规则id

表 4-649 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-650 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-651 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
conditions	Array of AnticrawlerCondition objects	匹配条件列表
name	String	规则名称
type	String	JS脚本反爬虫规则类型，指定防护路径：anticrawler_specific_url 排除防护路径：anticrawler_except_url
timestamp	Long	创建规则时间戳
status	Integer	规则状态，0：关闭，1：开启
priority	Integer	执行该规则的优先级，值越小，优先级越高，值相同时，规则创建时间早，优先级越高。取值范围：0到1000。

表 4-652 AnticrawlerCondition

参数	参数类型	描述
category	String	字段类型
logic_operation	String	条件列表匹配逻辑, 包括: contain、not_contain、equal、not_equal、prefix、not_prefix、suffix、not_suffix、contain_any、not_contain_all、equal_any、not_equal_all、prefix_any、not_prefix_all、suffix_any、not_suffix_all
contents	Array of strings	条件列表逻辑匹配内容。当 logic_operation 参数不以 any 或者 all 结尾时, 需要传该参数。
value_list_id	String	引用表 id。当 logic_operation 参数以 any 或者 all 结尾时, 需要传该参数。此外, 引用表类型要与 category 类型保持一致。

状态码: 400

表 4-653 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码, 可参考 STS5 联调自验证。IAM5 鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5 鉴权错误时才会返回此字段。

表 4-654 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-655 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-656 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-657 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-658 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id，防护策略id为policy_id，规则id为rule_id的JS脚本反爬虫防护规则

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/anticrawler/{rule_id}?
```

响应示例

状态码： 200

ok

```
{
  "id": "7e7983bf2c9c41029d642bcbf819346d",
  "policyid": "0681f69f94ac408e9688373e45a61fdb",
  "name": "test66",
  "timestamp": 1678931492172,
  "status": 1,
  "type": "anticrawler_except_url",
  "conditions": [ {
    "category": "url",
    "contents": [ "/test66" ],
    "logic_operation": "contain"
  } ],
  "priority": 50
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteAnticrawlerRuleSolution {

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

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
DeleteAnticrawlerRuleRequest request = new DeleteAnticrawlerRuleRequest();
request.withPolicyId("{policy_id}");
request.withRuleId("{rule_id}");
try {
    DeleteAnticrawlerRuleResponse response = client.deleteAnticrawlerRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteAnticrawlerRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_antcrawler_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

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

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteAnticrawlerRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeleteAnticrawlerRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.28 查询隐私屏蔽防护规则列表

功能介绍

查询隐私屏蔽防护规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/privacy

表 4-659 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-660 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
page	否	Integer	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。

请求参数

表 4-661 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-662 响应 Body 参数

参数	参数类型	描述
total	Integer	规则条数
items	Array of PrivacyResponse Body objects	规则详情数组

表 4-663 PrivacyResponseBody

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳
status	Integer	规则状态，0：关闭，1：开启
url	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"/"号结尾代表路径前缀
category	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数

参数	参数类型	描述
index	String	屏蔽字段名，根据“屏蔽字段”设置字段名，被屏蔽的字段将不会出现在日志中。
description	String	规则描述，可选参数，设置该规则的备注信息。

状态码： 400

表 4-664 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-665 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-666 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-667 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-668 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-669 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的隐私屏蔽防护规则列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/privacy?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "id": "97e4d35f375f4736a21cccfad77613eb",
    "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
    "timestamp": 1650533191385,
    "description": "demo",
    "status": 1,
    "url": "/demo",
    "category": "cookie",
    "index": "demo"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListPrivacyRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPrivacyRuleRequest request = new ListPrivacyRuleRequest();
```

```
request.withPolicyId("{policy_id}");
try {
    ListPrivacyRuleResponse response = client.listPrivacyRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListPrivacyRuleRequest()
        request.policy_id = "{policy_id}"
        response = client.list_privacy_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
```

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.29 创建隐私屏蔽防护规则

功能介绍

创建隐私屏蔽防护规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/privacy

表 4-670 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-671 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-672 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)。
Content-Type	是	String	内容类型

表 4-673 请求 Body 参数

参数	是否必选	参数类型	描述
url	是	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"（星号）结尾代表路径前缀
category	是	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数
index	是	String	屏蔽字段名，根据“屏蔽字段”设置字段名，被屏蔽的字段将不会出现在日志中。屏蔽字段名的长度不能超过2048字节，且只能由数字、字母、下划线和中划线组成。
description	否	String	规则描述，可选参数，设置该规则的备注信息。

响应参数

状态码： 200

表 4-674 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳
status	Integer	规则状态，0：关闭，1：开启
url	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"号结尾代表路径前缀

参数	参数类型	描述
category	String	屏蔽字段 <ul style="list-style-type: none">Params: 请求参数Cookie: 根据Cookie区分的Web访问者Header: 自定义HTTP首部Form: 表单参数
index	String	屏蔽字段名, 根据“屏蔽字段”设置字段名, 被屏蔽的字段将不会出现在日志中。
description	String	规则描述, 可选参数, 设置该规则的备注信息。

状态码: 400

表 4-675 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-676 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-677 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-678 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-679 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-680 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id，防护策略id为policy_id的防护策略下创建隐私屏蔽防护规则，隐私屏蔽规则防护的url为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo”，规则描述为“demo”。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/privacy?enterprise_project_id=0
{
  "url" : "/demo",
  "category" : "cookie",
  "index" : "demo",
  "description" : "demo"
}
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "97e4d35f375f4736a21cccfad77613eb",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "timestamp" : 1650533191385,
  "description" : "demo",
  "status" : 1,
  "url" : "/demo",
  "category" : "cookie",
  "index" : "demo"
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建隐私屏蔽防护规则，隐私屏蔽规则防护的url为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo”，规则描述为“demo”。

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

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

```
public class CreatePrivacyRuleSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreatePrivacyRuleRequest request = new CreatePrivacyRuleRequest();  
        request.withPolicyId("{policy_id}");  
        CreatePrivacyRuleRequestBody body = new CreatePrivacyRuleRequestBody();  
        body.withDescription("demo");  
        body.withIndex("demo");  
        body.withCategory(CreatePrivacyRuleRequestBody.CategoryEnum.fromValue("cookie"));  
        body.withUrl("/demo");  
        request.withBody(body);  
        try {  
            CreatePrivacyRuleResponse response = client.createPrivacyRule(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建隐私屏蔽防护规则，隐私屏蔽规则防护的url为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo”，规则描述为“demo”。

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

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

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = CreatePrivacyRuleRequest()
    request.policy_id = "{policy_id}"
    request.body = CreatePrivacyRuleRequestBody(
        description="demo",
        index="demo",
        category="cookie",
        url="/demo"
    )
    response = client.create_privacy_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建隐私屏蔽防护规则，隐私屏蔽规则防护的url为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo”，规则描述为“demo”。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreatePrivacyRuleRequest{}
    request.PolicyId = "{policy_id}"
```

```
descriptionCreatePrivacyRuleRequestBody:= "demo"
request.Body = &model.CreatePrivacyRuleRequestBody{
    Description: &descriptionCreatePrivacyRuleRequestBody,
    Index: "demo",
    Category: model.GetCreatePrivacyRuleRequestBodyCategoryEnum().COOKIE,
    Url: "/demo",
}
response, err := client.CreatePrivacyRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.30 查询隐私屏蔽防护规则

功能介绍

查询隐私屏蔽防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}

表 4-681 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	隐私屏蔽规则id, 您可以通过调用查询隐私屏蔽规则列表 (ListPrivacyRule) 获取规则id

表 4-682 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-683 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-684 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id

参数	参数类型	描述
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳
description	String	规则描述，可选参数，设置该规则的备注信息。
status	Integer	规则状态，0：关闭，1：开启
url	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"/"号结尾代表路径前缀
category	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数
index	String	屏蔽字段名

状态码： 400

表 4-685 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-686 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-687** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-688 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-689** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-690 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id，规则id为rule_id的隐私屏蔽防护规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id": "97e4d35f375f4736a21cccfad77613eb",
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "timestamp": 1650533191385,
  "description": "demo",
  "status": 1,
  "url": "/demo",
  "category": "cookie",
  "index": "demo1"
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowPrivacyRuleSolution {

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

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ShowPrivacyRuleRequest request = new ShowPrivacyRuleRequest();
request.withPolicyId("{policy_id}");
request.withRuleId("{rule_id}");
try {
    ShowPrivacyRuleResponse response = client.showPrivacyRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowPrivacyRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_privacy_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowPrivacyRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowPrivacyRule(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.31 更新隐私屏蔽防护规则

功能介绍

更新隐私屏蔽防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}

表 4-691 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	隐私屏蔽规则id，您可以通过调用查询隐私屏蔽规则列表（ListPrivacyRule）获取规则id

表 4-692 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-693 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)。
Content-Type	是	String	内容类型

表 4-694 请求 Body 参数

参数	是否必选	参数类型	描述
url	是	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以“（星号）”结尾代表路径前缀
category	是	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数
index	是	String	屏蔽字段名，根据“屏蔽字段”设置字段名，被屏蔽的字段将不会出现在日志中。屏蔽字段名长度不能超过2048字节，且只能由数字、字母、下划线和中划线组成
description	否	String	规则描述，可选参数，设置该规则的备注信息。

响应参数

状态码： 200**表 4-695** 响应 Body 参数

参数	参数类型	描述
id	String	规则id

参数	参数类型	描述
policyid	String	策略id
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳
status	Integer	规则状态，0：关闭，1：开启
url	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"/"号结尾代表路径前缀
category	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数
index	String	屏蔽字段名，根据“屏蔽字段”设置字段名，被屏蔽的字段将不会出现在日志中。
description	String	规则描述，可选参数，设置该规则的备注信息。

状态码： 400

表 4-696 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-697 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-698 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-699 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-700 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-701 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防护策略下创建隐私屏蔽防护规则列表, 隐私屏蔽规则防护的ur为“/demo”, 屏蔽字段为Cookie, 屏蔽字段名为“demo1”, 规则描述为“demo”。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}?enterprise_project_id=0
{
  "url" : "/demo",
  "category" : "cookie",
  "index" : "demo1",
  "description" : "demo"
}
```

响应示例

状态码: 200

请求成功

```
{
  "id" : "97e4d35f375f4736a21cccfad77613eb",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "description" : "demo",
  "url" : "/demo",
  "category" : "cookie",
  "index" : "demo1"
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防护策略下创建隐私屏蔽防护规则列表，隐私屏蔽规则防护的ur为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo1”，规则描述为“demo”。

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

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

public class UpdatePrivacyRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdatePrivacyRuleRequest request = new UpdatePrivacyRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdatePrivacyRuleRequestBody body = new UpdatePrivacyRuleRequestBody();
        body.withDescription("demo");
        body.withIndex("demo1");
        body.withCategory(UpdatePrivacyRuleRequestBody.CategoryEnum.fromValue("cookie"));
        body.withUrl("/demo");
        request.withBody(body);
        try {
            UpdatePrivacyRuleResponse response = client.updatePrivacyRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防护策略下创建隐私屏蔽防护规则列表，隐私屏蔽规则防护的ur为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo1”，规则描述为“demo”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePrivacyRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        request.body = UpdatePrivacyRuleRequestBody(
            description="demo",
            index="demo1",
            category="cookie",
            url="/demo"
        )
        response = client.update_privacy_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防护策略下创建隐私屏蔽防护规则列表，隐私屏蔽规则防护的ur为“/demo”，屏蔽字段为Cookie，屏蔽字段名为“demo1”，规则描述为“demo”。

```
package main

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

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdatePrivacyRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
descriptionUpdatePrivacyRuleRequestBody := "demo"
request.Body = &model.UpdatePrivacyRuleRequestBody{
    Description: &descriptionUpdatePrivacyRuleRequestBody,
    Index: "demo1",
    Category: model.GetUpdatePrivacyRuleRequestBodyCategoryEnum().COOKIE,
    Url: "/demo",
}
response, err := client.UpdatePrivacyRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.32 删除隐私屏蔽防护规则

功能介绍

删除隐私屏蔽防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}

表 4-702 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	隐私屏蔽规则id，您可以通过调用查询隐私屏蔽规则列表（ListPrivacyRule）获取规则id

表 4-703 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-704 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-705 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳
description	String	规则描述，可选参数，设置该规则的备注信息。
status	Integer	规则状态，0：关闭，1：开启
url	String	隐私屏蔽规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/以"号结尾代表路径前缀
category	String	屏蔽字段 <ul style="list-style-type: none">Params：请求参数Cookie：根据Cookie区分的Web访问者Header：自定义HTTP首部Form：表单参数
index	String	屏蔽字段名

状态码： 400

表 4-706 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-707 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-708 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-709 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-710 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-711 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的隐私屏蔽防护规则

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/privacy/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id": "97e4d35f375f4736a21cccfad77613eb",
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "timestamp": 1650533191385,
  "description": "demo",
  "status": 1,
  "url": "/demo",
  "category": "cookie",
  "index": "demo1"
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeletePrivacyRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeletePrivacyRuleRequest request = new DeletePrivacyRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeletePrivacyRuleResponse response = client.deletePrivacyRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = DeletePrivacyRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.delete_privacy_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeletePrivacyRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeletePrivacyRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.33 查询攻击惩罚规则列表

功能介绍

查询攻击惩罚规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/punishment

表 4-712 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，通过指定防护策略id来指明查询该防护策略下的攻击惩罚规则，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-713 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
offset	是	Integer	偏移量，表示查询该偏移量之后的记录。
limit	是	Integer	查询返回记录的数量限制。

请求参数

表 4-714 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-715 响应 Body 参数

参数	参数类型	描述
total	Integer	攻击惩罚规则数量
items	Array of PunishmentInfo objects	攻击惩罚规则列表

表 4-716 PunishmentInfo

参数	参数类型	描述
id	String	规则id
policyid	String	所属策略id

参数	参数类型	描述
block_time	Integer	拦截时间
category	String	攻击惩罚类别
description	String	规则描述
timestamp	Long	创建规则时间戳

状态码： 400

表 4-717 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-718 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-719 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-720 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-721 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-722 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id的攻击惩罚规则列表

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/punishment?offset=0&limit=2
```

响应示例

状态码： 200

请求成功

```
{
  "items": [ {
    "block_time": 305,
    "category": "long_ip_block",
    "description": "test",
    "id": "2c3afdcc982b429da4f72ee483aece3e",
    "policyid": "2fcbcb23ef0d48d99d24d7dcff00307d",
    "timestamp": 1668148186106
  } ],
  "total": 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

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

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

public class ListPunishmentRulesSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPunishmentRulesRequest request = new ListPunishmentRulesRequest();
        request.withPolicyId("{policy_id}");
        try {
```

```
ListPunishmentRulesResponse response = client.listPunishmentRules(request);
System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListPunishmentRulesRequest()
        request.policy_id = "{policy_id}"
        response = client.list_punishment_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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



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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.34 创建攻击惩罚规则

功能介绍

创建攻击惩罚规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/punishment

表 4-723 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-724 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-725 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)。
Content-Type	是	String	内容类型

表 4-726 请求 Body 参数

参数	是否必选	参数类型	描述
category	是	String	攻击惩罚类别
block_time	是	Integer	拦截时间, 如果选择前缀为long的攻击惩罚类别, 则block_time时长范围设置为301-1800;选择前缀为short的攻击惩罚类别, 则block_time时长范围为0-300之间

参数	是否必选	参数类型	描述
description	否	String	规则描述

响应参数

状态码： 200

表 4-727 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	所属策略id
block_time	Integer	拦截时间
category	String	攻击惩罚类别
description	String	规则描述
timestamp	Long	创建规则时间戳

状态码： 400

表 4-728 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-729 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码

参数	参数类型	描述
error_msg	String	下游服务报错信息

状态码： 401

表 4-730 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-731 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-732 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-733 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id，防护策略id为policy_id的防护策略下创建攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“demo”。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/punishment?
{
  "category": "long_ip_block",
  "block_time": "1233",
  "description": "demo"
}
```

响应示例

状态码：200

请求成功

```
{
  "block_time": 1233,
  "category": "long_ip_block",
  "description": "demo",
  "id": "2c3afdcc982b429da4f72ee483aece3e",
  "policyid": "2fcbbcb23ef0d48d99d24d7dcff00307d",
  "timestamp": 1668148186106
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“demo”。

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

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

public class CreatePunishmentRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreatePunishmentRuleRequest request = new CreatePunishmentRuleRequest();
        request.withPolicyId("{policy_id}");
        CreatePunishmentRuleRequestBody body = new CreatePunishmentRuleRequestBody();
        body.withDescription("demo");
        body.withBlockTime(1233);
        body.withCategory(CreatePunishmentRuleRequestBody.CategoryEnum.fromValue("long_ip_block"));
        request.withBody(body);
        try {
            CreatePunishmentRuleResponse response = client.createPunishmentRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“demo”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *
```

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreatePunishmentRuleRequest()
        request.policy_id = "{policy_id}"
        request.body = CreatePunishmentRuleRequestBody(
            description="demo",
            block_time=1233,
            category="long_ip_block"
        )
        response = client.create_punishment_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“demo”。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
```

```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.CreatePunishmentRuleRequest{}
request.PolicyId = "{policy_id}"
descriptionCreatePunishmentRuleRequestBody := "demo"
request.Body = &model.CreatePunishmentRuleRequestBody{
    Description: &descriptionCreatePunishmentRuleRequestBody,
    BlockTime: int32(1233),
    Category: model.GetCreatePunishmentRuleRequestBodyCategoryEnum().LONG_IP_BLOCK,
}
response, err := client.CreatePunishmentRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.35 根据 Id 查询攻击惩罚防护规则

功能介绍

根据Id查询攻击惩罚防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}

表 4-734 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	攻击惩罚规则id, 通过查询攻击惩罚规则列表接口 (ListPunishmentRules) 获取

表 4-735 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-736 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-737 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	所属策略id

参数	参数类型	描述
block_time	Integer	拦截时间
category	String	攻击惩罚类别
description	String	规则描述
timestamp	Long	创建规则时间戳

状态码： 400

表 4-738 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-739 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-740 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-741 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-742 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-743 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id，规则id为rule_id的攻击惩罚防护规则

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}?
```

响应示例

状态码： 200

请求成功

```
{
  "block_time" : 1233,
  "category" : "long_ip_block",
  "description" : "demo",
  "id" : "2c3afdcc982b429da4f72ee483aece3e",
  "policyid" : "2fcbcb23ef0d48d99d24d7dcff00307d",
  "timestamp" : 1668148186106
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowPunishmentRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        ShowPunishmentRuleRequest request = new ShowPunishmentRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowPunishmentRuleResponse response = client.showPunishmentRule(request);
        }
    }
}
```

```
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowPunishmentRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_punishment_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowPunishmentRuleRequest{
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
}
response, err := client.ShowPunishmentRule(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.36 更新攻击惩罚规则

功能介绍

更新攻击惩罚规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}

表 4-744 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	攻击惩罚规则id，通过查询攻击惩罚规则列表接口（ListPunishmentRules）获取

表 4-745 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-746 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)。
Content-Type	是	String	内容类型

表 4-747 请求 Body 参数

参数	是否必选	参数类型	描述
category	是	String	攻击惩罚类别
block_time	是	Integer	拦截时间，如果选择前缀为long的攻击惩罚类别，则block_time时长范围设置为301-1800;选择前缀为short的攻击惩罚类别，则block_time时长范围为0-300之间
description	否	String	规则描述

响应参数

状态码： 200

表 4-748 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	所属策略id
block_time	Integer	拦截时间
category	String	攻击惩罚类别
description	String	规则描述

状态码： 400

表 4-749 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-750 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-751 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-752 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-753 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-754 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的攻击惩罚规则, 攻击惩罚规则的攻击惩罚规则为“long_ip_block”, 拦截时间为1233秒, 规则描述为“update”。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}?
```

```
{
  "category": "long_ip_block",
  "block_time": "1233",
  "description": "update"
}
```

响应示例

状态码: 200

请求成功

```
{
  "block_time": 1233,
  "category": "long_ip_block",
  "description": "update",
  "id": "2c3afdcc982b429da4f72ee483aece3e",
  "policyid": "2fcbcb23ef0d48d99d24d7dcff00307d"
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为”update“。

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

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

public class UpdatePunishmentRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdatePunishmentRuleRequest request = new UpdatePunishmentRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdatePunishmentRuleRequestBody body = new UpdatePunishmentRuleRequestBody();
        body.withDescription("update");
        body.withBlockTime(1233);
        body.withCategory(UpdatePunishmentRuleRequestBody.CategoryEnum.fromValue("long_ip_block"));
        request.withBody(body);
        try {
            UpdatePunishmentRuleResponse response = client.updatePunishmentRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“update”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdatePunishmentRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        request.body = UpdatePunishmentRuleRequestBody(
            description="update",
            block_time=1233,
            category="long_ip_block"
        )
        response = client.update_punishment_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的攻击惩罚规则，攻击惩罚规则的攻击惩罚规则为“long_ip_block”，拦截时间为1233秒，规则描述为“update”。

```
package main

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

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdatePunishmentRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
descriptionUpdatePunishmentRuleRequestBody := "update"
request.Body = &model.UpdatePunishmentRuleRequestBody{
    Description: &descriptionUpdatePunishmentRuleRequestBody,
    BlockTime: int32(1233),
    Category: model.GetUpdatePunishmentRuleRequestBodyCategoryEnum().LONG_IP_BLOCK,
}
response, err := client.UpdatePunishmentRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.37 删除攻击惩罚规则

功能介绍

删除攻击惩罚规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}

表 4-755 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	攻击惩罚规则id，通过查询攻击惩罚规则列表接口（ListPunishmentRules）获取

表 4-756 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-757 请求 Header 参数

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

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-758 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	所属策略id
block_time	Integer	拦截时间
category	String	攻击惩罚类别
description	String	规则描述
timestamp	Long	创建规则时间戳

状态码： 400

表 4-759 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-760 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码

参数	参数类型	描述
error_msg	String	下游服务报错信息

状态码： 401

表 4-761 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-762 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-763 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-764 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id，防护策略id为policy_id，规则id为rule_id的攻击惩罚规则

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/punishment/{rule_id}?
```

响应示例

状态码： 200

请求成功

```
{
  "block_time" : 1233,
  "category" : "long_ip_block",
  "description" : "update",
  "id" : "2c3afdcc982b429da4f72ee483aece3e",
  "policyid" : "2fcbbcb23ef0d48d99d24d7dcff00307d",
  "timestamp" : 1668148186106
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeletePunishmentRuleSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
DeletePunishmentRuleRequest request = new DeletePunishmentRuleRequest();
request.withPolicyId("{policy_id}");
request.withRuleId("{rule_id}");
try {
    DeletePunishmentRuleResponse response = client.deletePunishmentRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeletePunishmentRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_punishment_rule(request)
```

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

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeletePunishmentRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeletePunishmentRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败

状态码	描述
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.38 查询地理位置访问控制规则列表

功能介绍

查询地理位置访问控制规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/geoip

表 4-765 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-766 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。

参数	是否必选	参数类型	描述
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。

请求参数

表 4-767 请求 Header 参数

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

响应参数

状态码： 200

表 4-768 响应 Body 参数

参数	参数类型	描述
total	Integer	该策略下地理位置控制规则数量
items	Array of GeoIpItem objects	地理位置控制规则数组

表 4-769 GeoIpItem

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
name	String	地理位置控制规则名称

参数	参数类型	描述
geoip	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
status	Integer	规则状态, 0: 关闭, 1: 开启
timestamp	Long	创建规则时间戳

状态码: 400**表 4-770 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-771 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-772 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-773 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-774 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-775 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id的地理位置访问控制规则列表

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/geoip?enterprise_project_id=0
```

响应示例

状态码: 200

OK

```
{
  "total": 1,
  "items": [{
    "id": "06f07f6c229141b9a4a78614751bb687",
    "policyid": "2abeecefb9840e6bf05efbd80d0fcd7",
    "timestamp": 1636340038062,
    "status": 1,
    "geoip": "GD",
    "white": 1,
    "name": "demo"
  }]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListGeoipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListGeoipRuleRequest request = new ListGeoipRuleRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListGeoipRuleResponse response = client.listGeoipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = ListGeoipRuleRequest()
    request.policy_id = "{policy_id}"
    response = client.list_geoip_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.39 创建地理位置控制规则

功能介绍

创建地理位置控制规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/geoip

表 4-776 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-777 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-778 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)。
Content-Type	是	String	内容类型

表 4-779 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	地理位置控制规则名称
geoip	是	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	是	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
status	否	Integer	规则状态，0：关闭，1：开启

参数	是否必选	参数类型	描述
description	否	String	规则描述

响应参数

状态码： 200

表 4-780 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	地理位置控制规则名称
policyid	String	策略id
geoip	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
status	Integer	规则状态, 0: 关闭, 1: 开启
timestamp	Long	创建规则时间戳

状态码： 400

表 4-781 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-782 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-783 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-784 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-785 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-786 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id, 防护策略id为policy_id的防护策略下创建地理位置控制规则, 地理位置控制规则的防护动作为拦截, 规则描述为”demo“, 规则名称为”demo“, 地理位置封禁区域为上海和阿富汗。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/geoip?enterprise_project_id=0
{
  "white": 0,
  "description": "demo",
  "name": "demo",
  "geoip": "SH|Afghanistan"
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "02dafa406c4941368a1037b020f15a53",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name" : "demo",
  "timestamp" : 1650534513775,
  "status" : 1,
  "geoip" : "SH|Afghanistan",
  "white" : 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建地理位置控制规则，地理位置控制规则的防护动作为拦截，规则描述为”demo“，规则名称为”demo“，地理位置封禁区域为上海和阿富汗。

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

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

public class CreateGeoipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateGeoipRuleRequest request = new CreateGeoipRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateGeoipRuleRequestBody body = new CreateGeoipRuleRequestBody();
        body.withDescription("demo");
        body.withWhite(0);
        body.withGeoip("SH|Afghanistan");
        body.withName("demo");
    }
}
```



```
request.withBody(body);
try {
    CreateGeoipRuleResponse response = client.createGeoipRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建地理位置控制规则，地理位置控制规则的防护动作为拦截，规则描述为”demo“，规则名称为”demo“，地理位置封禁区域为上海和阿富汗。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateGeoipRuleRequest()
        request.policy_id = "{policy_id}"
        request.body = CreateGeoipRuleRequestBody(
            description="demo",
            white=0,
            geoip="SH|Afghanistan",
            name="demo"
        )
        response = client.create_geoip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建地理位置控制规则，地理位置控制规则的防护动作为拦截，规则描述为”demo“，规则名称为”demo“，地理位置封禁区域为上海和阿富汗。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateGeoipRuleRequest{}
    request.PolicyId = "{policy_id}"
    descriptionCreateGeoipRuleRequestBody := "demo"
    nameCreateGeoipRuleRequestBody := "demo"
    request.Body = &model.CreateGeoipRuleRequestBody{
        Description: &descriptionCreateGeoipRuleRequestBody,
        White: int32(0),
        Geoip: "SH|Afghanistan",
        Name: &nameCreateGeoipRuleRequestBody,
    }
    response, err := client.CreateGeoipRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.40 查询地理位置控制防护规则

功能介绍

查询地理位置控制防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/geoi/{rule_id}

表 4-787 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	地理位置控制防护规则id，规则id从查询地理位置控制防护规则列表（ListGeoiRule）接口获取，响应体的id字段

表 4-788 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-789 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-790 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id

参数	参数类型	描述
geoip	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	Integer	防护动作： <ul style="list-style-type: none"> • 0 拦截 • 1 放行 • 2 仅记录
status	Integer	规则状态, 0: 关闭, 1: 开启
timestamp	Long	创建规则时间戳

状态码: 400

表 4-791 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-792 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-793 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-794 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-795 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-796 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的地理位置控制防护规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/geoip/{rule_id}?enterprise_project_id=0
```

响应示例

状态码: 200

请求成功

```
{
  "id": "02dafa406c4941368a1037b020f15a53",
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",
  "status": 1,
  "geoip": "BJ|Afghanistan",
  "white": 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.waf.v1.region.WafRegion;
```

```
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowGeoipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowGeoipRuleRequest request = new ShowGeoipRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowGeoipRuleResponse response = client.showGeoipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
```



```
.with_region(WafRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = ShowGeoipRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.show_geoip_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowGeoipRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowGeoipRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.41 更新地理位置控制防护规则

功能介绍

更新地理位置控制防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/geoip/{rule_id}

表 4-797 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id, 响应体的id字段
rule_id	是	String	地理位置控制规则id, 规则id从查询地理位置规则列表 (ListGeoipRule) 接口获取, 响应体的id字段

表 4-798 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-799 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)。
Content-Type	是	String	内容类型

表 4-800 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	地理位置控制规则名称
description	否	String	描述

参数	是否必选	参数类型	描述
geoip	是	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	是	Integer	防护动作： <ul style="list-style-type: none"> • 0 拦截 • 1 放行 • 2 仅记录

响应参数

状态码： 200

表 4-801 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	地理位置控制规则名称
description	String	描述
policyid	String	策略id

参数	参数类型	描述
geoip	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	Integer	防护动作： <ul style="list-style-type: none"> • 0 拦截 • 1 放行 • 2 仅记录

状态码： 400

表 4-802 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-803 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-804 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-805 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-806 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-807 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的地理位置控制防护规则, 地理位置控制防护规则的防护动作为拦截, 规则名称为”demo“, 地理位置封禁区域为北京和阿富汗。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/geoip/{rule_id}?enterprise_project_id=0
{
  "white" : 0,
  "name" : "demo",
  "geoip" : "BJ|Afghanistan"
}
```

响应示例

状态码: 200

请求成功

```
{
  "id" : "02dafa406c4941368a1037b020f15a53",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "name" : "demo",
  "description" : "demo",
  "geoip" : "BJ|Afghanistan",
  "white" : 0
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的地理位置控制防护规则，地理位置控制防护规则的防护动作为拦截，规则名称为”demo“，地理位置封禁区域为北京和阿富汗。

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

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

public class UpdateGeoipRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateGeoipRuleRequest request = new UpdateGeoipRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateGeoipRuleRequestBody body = new UpdateGeoipRuleRequestBody();
        body.withWhite(0);
        body.withGeoip("BJ|Afghanistan");
        body.withName("demo");
        request.withBody(body);
        try {
            UpdateGeoipRuleResponse response = client.updateGeoipRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```


Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的地理位置控制防护规则，地理位置控制防护规则的防护动作为拦截，规则名称为”demo“，地理位置封禁区域为北京和阿富汗。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateGeoipRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        request.body = UpdateGeoipRuleRequestBody(
            white=0,
            geoip="BJ|Afghanistan",
            name="demo"
        )
        response = client.update_geoip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的地理位置控制防护规则，地理位置控制防护规则的防护动作为拦截，规则名称为”demo“，地理位置封禁区域为北京和阿富汗。

```
package main

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

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

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateGeoipRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
nameUpdateGeoipRuleRequestBody := "demo"
request.Body = &model.UpdateGeoipRuleRequestBody{
    White: int32(0),
    Geoip: "BJ|Afghanistan",
    Name: &nameUpdateGeoipRuleRequestBody,
}
response, err := client.UpdateGeoipRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.42 删除地理位置控制防护规则

功能介绍

删除地理位置控制防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/geoup/{rule_id}

表 4-808 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	地理位置控制防护规则id，规则id从查询地理位置控制防护规则列表（ListGeoupRule）接口获取，响应体的id字段

表 4-809 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-810 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-811 响应 Body 参数

参数	参数类型	描述
id	String	规则id
name	String	地理位置控制规则名称
policyid	String	策略id
geoip	String	地理位置封禁区域： (CA: 加拿大,US: 美国,AU: 澳大利亚,IN: 印度,JP: 日本,UK: 英国,FR: 法国,DE: 德国,BR: 巴西,Thailand: 泰国,Singapore: 新加坡,South Africa: 南非,Mexico: 墨西哥,Peru: 秘鲁,Indonesia: 印度尼西亚,GD: 广东,FJ: 福建,JL: 吉林,LN: 辽宁,TW: 中国-台湾,GZ: 贵州,AH: 安徽,HL: 黑龙江,HA: 河南,SC: 四川,HE: 河北,YN: 云南,HB: 湖北,HI: 海南,QH: 青海,HN: 湖南,JX: 江西,SX: 山西,SN: 陕西,ZJ: 浙江,GS: 甘肃,JS: 江苏,SD: 山东,BJ: 北京,SH: 上海,TJ: 天津,CQ: 重庆,MO: 中国-澳门,HK: 中国-香港,NX: 宁夏,GX: 广西,XJ: 新疆,XZ: 西藏,NM: 内蒙古)。具体的位置编码参见《附录-地理位置编码》查询。
white	Integer	防护动作： <ul style="list-style-type: none">• 0 拦截• 1 放行• 2 仅记录
status	Integer	规则状态, 0: 关闭, 1: 开启
description	String	描述
timestamp	Long	创建规则时间戳

状态码： 400

表 4-812 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-813 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-814 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-815 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-816 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-817 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的地理位置控制防护规则

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/geoip/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{  
  "id": "02dafa406c4941368a1037b020f15a53",  
  "policyid": "38ff0cb9a10e4d5293c642bc0350fa6d",  
}
```

```
"timestamp" : 1650534513775,  
"status" : 1,  
"geoip" : "BJ|Afghanistan",  
"white" : 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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class DeleteGeoipRuleSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeleteGeoipRuleRequest request = new DeleteGeoipRuleRequest();  
        request.withPolicyId("{policy_id}");  
        request.withRuleId("{rule_id}");  
        try {  
            DeleteGeoipRuleResponse response = client.deleteGeoipRule(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteGeoipRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_geoip_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
```



```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.DeleteGeoipRuleRequest{
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.DeleteGeoipRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.43 查询防篡改规则列表

功能介绍

查询防篡改规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/antitamper

表 4-818 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-819 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。
pagesize	否	Integer	分页查询时, 每页包含多少条结果。范围1-100, 默认值为10, 表示每页包含10条结果。

请求参数

表 4-820 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-821 响应 Body 参数

参数	参数类型	描述
total	Integer	网页防篡改规则总条数
items	Array of AntiTamperRuleResponseBody objects	网页防篡改规则数组

表 4-822 AntiTamperRuleResponseBody

参数	参数类型	描述
id	String	规则id
policyid	String	该规则所属防护策略的id
timestamp	Long	创建规则的时间戳
description	String	该规则备注
status	Integer	规则状态, 0: 关闭, 1: 开启
hostname	String	防篡改的域名
url	String	防篡改的url

状态码: 400

表 4-823 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-824 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-825 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-826 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-827 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-828 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id的防篡改规则列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antitamper?enterprise_project_id=0
```

响应示例

状态码: 200

OK

```
{
  "total": 1,
  "items": [{
    "id": "b77c3182957b46ed8f808a1998245cc4",
    "policyid": "bdba8e224cbd4d11915f244c991d1720",
    "timestamp": 1647499571037,
    "description": "",
    "status": 0,
    "hostname": "www.demo.com",
    "url": "/sdf"
  }]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListAntitamperRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListAntitamperRuleRequest request = new ListAntitamperRuleRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListAntitamperRuleResponse response = client.listAntitamperRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = ListAntitamperRuleRequest()
    request.policy_id = "{policy_id}"
    response = client.list_antitamper_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.44 创建防篡改规则

功能介绍

创建防篡改规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/antitamper

表 4-829 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-830 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-831 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)。
Content-Type	是	String	内容类型

表 4-832 请求 Body 参数

参数	是否必选	参数类型	描述
hostname	是	String	防护网站，查询云模式防护域名列表（ListHost）接口获取防护域名，响应体中的hostname字段
url	是	String	防篡改规则防护的url，需要填写标准的url格式，例如/admin/xxx或者/admin/，以"/"号结尾代表路径前缀
description	否	String	规则描述

响应参数

状态码： 200

表 4-833 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
hostname	String	防篡改的域名
url	String	防篡改的url，
description	String	创建规则的时间戳
status	Integer	规则状态，0：关闭，1：开启

状态码： 400

表 4-834 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-835 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-836 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-837 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-838 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-839 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id, 防护策略id为policy_id的防护策略下创建防篡改规则。创建防篡改规则的防护网站为"www.demo.com", 防篡改规则防护的url为"/test", 规则描述为"demo"。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antitamper?enterprise_project_id=0
{
  "hostname": "www.demo.com",
  "url": "/test",
  "description": "demo"
}
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "eed1c1e9c1b04b4bad4ba1186387a5d8",
  "policyid" : "38ff0cb9a10e4d5293c642bc0350fa6d",
  "description" : "demo",
  "status" : 1,
  "hostname" : "www.demo.com",
  "url" : "/test"
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id，防护策略id为policy_id的防护策略下创建防篡改规则。创建防篡改规则的防护网站为"www.demo.com"，防篡改规则防护的url为"/test"，规则描述为"demo"。

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

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

public class CreateAntiTamperRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateAntiTamperRuleRequest request = new CreateAntiTamperRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateAntiTamperRulesRequestBody body = new CreateAntiTamperRulesRequestBody();
        body.withDescription("demo");
        body.withUrl("/test");
        body.withHostname("www.demo.com");
        request.withBody(body);
        try {
```

```
        CreateAntiTamperRuleResponse response = client.createAntiTamperRule(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

在项目id为project_id，防护策略id为policy_id的防护策略下创建防篡改规则。创建防篡改规则的防护网站为"www.demo.com"，防篡改规则防护的url为"/test"，规则描述为"demo"。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateAntiTamperRuleRequest()
        request.policy_id = "{policy_id}"
        request.body = CreateAntiTamperRulesRequestBody(
            description="demo",
            url="/test",
            hostname="www.demo.com"
        )
        response = client.create_anti_tamper_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在项目id为project_id，防护策略id为policy_id的防护策略下创建防篡改规则。创建防篡改规则的防护网站为"www.demo.com"，防篡改规则防护的url为"/test"，规则描述为"demo"。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateAntiTamperRuleRequest{}
    request.PolicyId = "{policy_id}"
    descriptionCreateAntiTamperRulesRequestBody:= "demo"
    request.Body = &model.CreateAntiTamperRulesRequestBody{
        Description: &descriptionCreateAntiTamperRulesRequestBody,
        Url: "/test",
        Hostname: "www.demo.com",
    }
    response, err := client.CreateAntiTamperRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，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

状态码	描述
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.45 查询防篡改防护规则

功能介绍

查询防篡改防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}

表 4-840 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	防篡改规则id，通过查询防篡改规则列表接口（ListAntitamperRule）获取

表 4-841 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-842 请求 Header 参数

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

响应参数

状态码： 200

表 4-843 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
hostname	String	防篡改的域名
url	String	防篡改的url，
description	String	创建规则的时间戳
status	Integer	规则状态，0：关闭，1：开启
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳

状态码： 400

表 4-844 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码(加密)后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-845 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-846 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-847 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-848 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-849 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防篡改防护规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}?enterprise_project_id=0
```

响应示例

状态码: 200

请求成功

```
{
  "id": "b77c3182957b46ed8f808a1998245cc4",
  "policyid": "bdba8e224cbd4d11915f244c991d1720",
  "timestamp": 1647499571037,
  "description": "",
  "status": 0,
  "hostname": "www.demo.com",
  "url": "/sdf"
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowAntitamperRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowAntitamperRuleRequest request = new ShowAntitamperRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            ShowAntitamperRuleResponse response = client.showAntitamperRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = ShowAntitamperRuleRequest()
    request.policy_id = "{policy_id}"
    request.rule_id = "{rule_id}"
    response = client.show_antitamper_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAntitamperRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.ShowAntitamperRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.46 删除防篡改防护规则

功能介绍

删除防篡改防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}

表 4-850 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	防篡改规则id，通过查询防篡改规则列表接口（ListAntitamperRule）获取

表 4-851 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-852 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-853 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
url	String	防篡改的url
timestamp	Long	创建规则的时间戳

状态码： 400

表 4-854 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-855 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-856 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-857 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-858 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-859 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防篡改防护规则。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "id": "b77c3182957b46ed8f808a1998245cc4",
    "policyid": "bdba8e224cbd4d11915f244c991d1720",
    "policyname": "demo",
    "timestamp": 1647499571037,
    "description": "",
    "status": 0,
    "hostname": "www.demo.com",
    "url": "/sdf"
  } ]
}
```


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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteAntitamperRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteAntitamperRuleRequest request = new DeleteAntitamperRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        try {
            DeleteAntitamperRuleResponse response = client.deleteAntitamperRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *
```

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteAntitamperRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_antitamper_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteAntitamperRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    response, err := client.DeleteAntitamperRule(request)
    if err == nil {
```

```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.47 网页防篡改规则更新缓存

功能介绍

网页防篡改规则更新缓存

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}/refresh

表 4-860 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

参数	是否必选	参数类型	描述
rule_id	是	String	防篡改规则id, 通过查询防篡改规则列表接口 (ListAntitamperRule) 获取

表 4-861 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-862 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-863 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
hostname	String	防篡改的域名
url	String	防篡改的url
description	String	创建规则的时间戳
status	Integer	规则状态, 0: 关闭, 1: 开启
timestamp	Long	创建规则时间戳

状态码： 400**表 4-864** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-865 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-866** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-867 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-868 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-869 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

给项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防篡改防护规则更新缓存。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antitamper/{rule_id}/refresh?
```

响应示例

状态码： 200

ok

```
{  
  "description": "",  
  "hostname": "www.domain.com",
```

```
"id" : "0f59185b76c143f884d21cd0d88e6fa8",  
"policyid" : "1f016cde588646aca3fb19f277c44d03",  
"status" : 1,  
"timestamp" : 1666506256928,  
"url" : "/login"  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class UpdateAntiTamperRuleRefreshSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateAntiTamperRuleRefreshRequest request = new UpdateAntiTamperRuleRefreshRequest();  
        request.withPolicyId("{policy_id}");  
        request.withRuleId("{rule_id}");  
        try {  
            UpdateAntiTamperRuleRefreshResponse response = client.updateAntiTamperRuleRefresh(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateAntiTamperRuleRefreshRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.update_anti_tamper_rule_refresh(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
```



```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.UpdateAntiTamperRuleRefreshRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.UpdateAntiTamperRuleRefresh(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.48 查询防敏感信息泄露规则列表

功能介绍

查询防敏感信息泄露规则列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/antileakage

表 4-870 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 通过指定防护策略id来指明查询该防护策略下的防敏感信息泄露规则, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id

表 4-871 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id
offset	是	Integer	偏移量, 表示查询该偏移量之后的记录。
limit	是	Integer	查询返回记录的数量限制。

请求参数

表 4-872 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-873 响应 Body 参数

参数	参数类型	描述
total	Integer	防泄漏规则数量

参数	参数类型	描述
items	Array of LeakageListInfo objects	防泄漏规则列表

表 4-874 LeakageListInfo

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
url	String	规则应用的url
category	String	类别
contents	Array of strings	规则内容
timestamp	Long	创建规则时间戳
status	Integer	规则状态, 0: 关闭, 1: 开启
description	String	规则描述

状态码: 400

表 4-875 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-876 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-877 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-878 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-879 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-880 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护策略id为policy_id的防敏感信息泄露规则列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antileakage?offset=0&limit=2
```

响应示例

状态码: 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "id": "82c4f04f84fd4b2b9ba4b4ea0df8ee82",
    "policyid": "2fcbcb23ef0d48d99d24d7dcff00307d",
    "timestamp": 1668152426471,
    "description": "demo",
    "status": 1,
    "url": "/attack",
    "category": "sensitive",
    "contents": [ "id_card" ]
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListAntileakageRulesSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListAntileakageRulesRequest request = new ListAntileakageRulesRequest();
        request.withPolicyId("{policy_id}");
        try {
            ListAntileakageRulesResponse response = client.listAntileakageRules(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

```
client = WafClient.new_builder() \  
  .with_credentials(credentials) \  
  .with_region(WafRegion.value_of("<YOUR REGION>")) \  
  .build()  
  
try:  
  request = ListAntileakageRulesRequest()  
  request.policy_id = "{policy_id}"  
  response = client.list_antileakage_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"  
  waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
  region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
  // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
  // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
  // variables and decrypted during use to ensure security.  
  // In this example, AK and SK are stored in environment variables for authentication. Before running this  
  // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
  ak := os.Getenv("CLOUD_SDK_AK")  
  sk := os.Getenv("CLOUD_SDK_SK")  
  projectId := "{project_id}"  
  
  auth := basic.NewCredentialsBuilder().  
    WithAk(ak).  
    WithSk(sk).  
    WithProjectId(projectId).  
    Build()  
  
  client := waf.NewWafClient(  
    waf.WafClientBuilder().  
      WithRegion(region.ValueOf("<YOUR REGION>")).  
      WithCredential(auth).  
      Build())  
  
  request := &model.ListAntileakageRulesRequest{}  
  request.PolicyId = "{policy_id}"  
  response, err := client.ListAntileakageRules(request)  
  if err == nil {  
    fmt.Printf("%+v\n", response)  
  } else {  
    fmt.Println(err)  
  }  
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.49 创建防敏感信息泄露规则

功能介绍

创建防敏感信息泄露规则

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/policy/{policy_id}/antileakage

表 4-881 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id

表 4-882 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-883 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)。
Content-Type	是	String	内容类型

表 4-884 请求 Body 参数

参数	是否必选	参数类型	描述
url	是	String	规则应用的url
category	是	String	类别（响应码：code，敏感信息：sensitive）
contents	是	Array of strings	规则内容（http状态码：400、401、402、403、404、405、500、501、502、503、504、507；手机：phone、身份证号：id_card、邮箱：email）
action	否	action object	规则命中后操作对象
description	否	String	规则描述

表 4-885 action

参数	是否必选	参数类型	描述
category	是	String	操作类型： <ul style="list-style-type: none">• block：过滤• log：仅记录

响应参数

状态码：200

表 4-886 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
url	String	规则应用的url
category	String	类别
contents	Array of strings	内容
timestamp	Long	创建规则时间戳
status	Integer	规则状态, 0: 关闭, 1: 开启

状态码: 400

表 4-887 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-888 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-889 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-890 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-891 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-892 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在查询项目id为project_id，防护策略id为policy_id的防护策略下创建防敏感信息泄露规则。防敏感信息泄露规则的规则应用的url为"/attack"，类别为敏感信息，规则内容为身份证号。

```
POST https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antileakage?
```

```
{
  "url": "/attack",
  "category": "sensitive",
  "contents": [ "id_card" ]
}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "82c4f04f84fd4b2b9ba4b4ea0df8ee82",
  "policyid": "2fcbcb23ef0d48d99d24d7dcff00307d",
  "timestamp": 1668152426471,
  "status": 1,
  "url": "/attack",
  "category": "sensitive",
  "contents": [ "id_card" ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

在查询项目id为project_id，防护策略id为policy_id的防护策略下创建防敏感信息泄露规则。防敏感信息泄露规则的规则应用的url为"/attack"，类别为敏感信息，规则内容为身份证号。

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

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

import java.util.List;
```

```
import java.util.ArrayList;

public class CreateAntileakageRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateAntileakageRuleRequest request = new CreateAntileakageRuleRequest();
        request.withPolicyId("{policy_id}");
        CreateAntileakageRuleRequestBody body = new CreateAntileakageRuleRequestBody();
        List<String> listbodyContents = new ArrayList<>();
        listbodyContents.add("id_card");
        body.withContents(listbodyContents);
        body.withCategory(CreateAntileakageRuleRequestBody.CategoryEnum.fromValue("sensitive"));
        body.withUrl("/attack");
        request.withBody(body);
        try {
            CreateAntileakageRuleResponse response = client.createAntileakageRule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

在查询项目id为project_id，防护策略id为policy_id的防护策略下创建防敏感信息泄露规则。防敏感信息泄露规则的规则应用的url为"/attack"，类别为敏感信息，规则内容为身份证号。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

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

try:
    request = CreateAntileakageRuleRequest()
    request.policy_id = "{policy_id}"
    listContentsbody = [
        "id_card"
    ]
    request.body = CreateAntileakageRuleRequestBody(
        contents=listContentsbody,
        category="sensitive",
        url="/attack"
    )
    response = client.create_antileakage_rule(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在查询项目id为project_id，防护策略id为policy_id的防护策略下创建防敏感信息泄露规则。防敏感信息泄露规则的规则应用的url为"/attack"，类别为敏感信息，规则内容为身份证号。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.CreateAntileakageRuleRequest{}
request.PolicyId = "{policy_id}"
var listContentsbody = []string{
    "id_card",
}
request.Body = &model.CreateAntileakageRuleRequestBody{
    Contents: listContentsbody,
    Category: model.GetCreateAntileakageRuleRequestBodyCategoryEnum().SENSITIVE,
    Url: "/attack",
}
response, err := client.CreateAntileakageRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.50 查询防敏感信息泄露防护规则

功能介绍

根据Id查询防敏感信息泄露防护规则

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}

表 4-893 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id, 您可以通过调用查询防护策略列表 (ListPolicy) 获取策略id
rule_id	是	String	防敏感信息泄露规则id, 通过查询防敏感信息泄露规则列表接口 (ListAntileakageRules) 获取

表 4-894 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-895 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-896 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id

参数	参数类型	描述
url	String	规则应用的url
category	String	类别
contents	Array of strings	内容
timestamp	Long	创建规则时间戳
status	Integer	规则状态, 0: 关闭, 1: 开启
description	String	规则描述

状态码: 400

表 4-897 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-898 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-899 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-900 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-901 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-902 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，防护策略id为policy_id，规则id为rule_id的防敏感信息泄露防护规则。

```
GET https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}?
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "82c4f04f84fd4b2b9ba4b4ea0df8ee82",
  "policyid" : "2fcbcb23ef0d48d99d24d7dcff00307d",
  "timestamp" : 1668152426471,
  "description" : "demo",
  "status" : 1,
  "url" : "/attack",
  "category" : "sensitive",
  "contents" : [ "id_card" ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowAntileakageRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowAntileakageRuleRequest request = new ShowAntileakageRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
```

```
try {
    ShowAntileakageRuleResponse response = client.showAntileakageRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowAntileakageRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.show_antileakage_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
```

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

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowAntileakageRuleRequest{
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
}
response, err := client.ShowAntileakageRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.51 更新防敏感信息泄露防护规则

功能介绍

更新防敏感信息泄露防护规则

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}

表 4-903 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	防敏感信息泄露规则id，通过查询防敏感信息泄露规则列表接口（ListAntileakageRules）获取

表 4-904 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-905 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)。
Content-Type	是	String	内容类型

表 4-906 请求 Body 参数

参数	是否必选	参数类型	描述
url	是	String	规则应用的url
category	是	String	类别（响应码：code，敏感信息：sensitive）
contents	是	Array of strings	内容（http状态码：400、401、402、403、404、405、500、501、502、503、504、507；手机：phone、身份证号：id_card、邮箱：email）
action	否	action object	规则命中后操作对象
description	否	String	规则描述

表 4-907 action

参数	是否必选	参数类型	描述
category	是	String	操作类型： <ul style="list-style-type: none">• block：过滤• log：仅记录

响应参数

状态码： 200

表 4-908 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
url	String	规则应用的url
category	String	类别
contents	Array of strings	规则内容
status	Integer	规则状态，0：关闭，1：开启
description	String	规则描述

状态码： 400

表 4-909 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-910 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-911 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-912 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-913 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-914 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防篡改防护规则。防敏感信息泄露规则的规则应用的url为"/login", 类别为敏感信息, 规则内容为身份证号。

```
PUT https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}?
```

```
{
  "url": "/login",
  "category": "sensitive",
  "contents": [ "id_card" ]
}
```

响应示例

状态码： 200

请求成功

```
{
  "id" : "82c4f04f84fd4b2b9ba4b4ea0df8ee82",
  "policyid" : "2fcbb23ef0d48d99d24d7dcff00307d",
  "description" : "demo",
  "status" : 1,
  "url" : "/login",
  "category" : "sensitive",
  "contents" : [ "id_card" ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防篡改防护规则。防敏感信息泄露规则的规则应用的url为"/login"，类别为敏感信息，规则内容为身份证号。

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

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

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

public class UpdateAntileakageRuleSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateAntileakageRuleRequest request = new UpdateAntileakageRuleRequest();
        request.withPolicyId("{policy_id}");
        request.withRuleId("{rule_id}");
        UpdateAntileakageRuleRequestBody body = new UpdateAntileakageRuleRequestBody();
        List<String> listbodyContents = new ArrayList<>();
```

```
listbodyContents.add("id_card");
body.withContents(listbodyContents);
body.withCategory(UpdateAntileakageRuleRequestBody.CategoryEnum.fromValue("sensitive"));
body.withUrl("/login");
request.withBody(body);
try {
    UpdateAntileakageRuleResponse response = client.updateAntileakageRule(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防篡改防护规则。防敏感信息泄露规则的规则应用的url为"/login"，类别为敏感信息，规则内容为身份证号。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateAntileakageRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        listContentsbody = [
            "id_card"
        ]
        request.body = UpdateAntileakageRuleRequestBody(
            contents=listContentsbody,
            category="sensitive",
            url="/login"
        )
        response = client.update_antileakage_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
```

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

Go

更新项目id为project_id，防护策略id为policy_id，规则id为rule_id的防篡改防护规则。防敏感信息泄露规则的规则应用的url为"/login"，类别为敏感信息，规则内容为身份证号。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAntileakageRuleRequest{}
    request.PolicyId = "{policy_id}"
    request.RuleId = "{rule_id}"
    var listContentsbody = []string{
        "id_card",
    }
    request.Body = &model.UpdateAntileakageRuleRequestBody{
        Contents: listContentsbody,
        Category: model.GetUpdateAntileakageRuleRequestBodyCategoryEnum().SENSITIVE,
        Url: "/login",
    }
    response, err := client.UpdateAntileakageRule(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.52 删除防敏感信息泄露防护规则

功能介绍

删除防敏感信息泄露防护规则

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}

表 4-915 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
policy_id	是	String	防护策略id，您可以通过调用查询防护策略列表（ListPolicy）获取策略id
rule_id	是	String	防敏感信息泄露规则id，通过查询防敏感信息泄露规则列表接口（ListAntileakageRules）获取

表 4-916 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-917 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-918 响应 Body 参数

参数	参数类型	描述
id	String	规则id
policyid	String	策略id
url	String	规则应用的url
category	String	类别
contents	Array of strings	内容
timestamp	Long	创建规则时间戳
status	Integer	规则状态，0：关闭，1：开启
description	String	规则描述

状态码： 400

表 4-919 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-920 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-921 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-922 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-923 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-924 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 防护策略id为policy_id, 规则id为rule_id的防敏感信息泄露防护规则。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/policy/{policy_id}/antileakage/{rule_id}?
```

响应示例

状态码： 200

请求成功

```
{  
  "id": "82c4f04f84fd4b2b9ba4b4ea0df8ee82",  
  "policyid": "2fcbcb23ef0d48d99d24d7dcff00307d",  
}
```



```
"timestamp" : 1668152426471,  
"description" : "demo",  
"status" : 1,  
"url" : "/attack",  
"category" : "sensitive",  
"contents" : [ "id_card" ]  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class DeleteAntileakageRuleSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeleteAntileakageRuleRequest request = new DeleteAntileakageRuleRequest();  
        request.withPolicyId("{policy_id}");  
        request.withRuleId("{rule_id}");  
        try {  
            DeleteAntileakageRuleResponse response = client.deleteAntileakageRule(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteAntileakageRuleRequest()
        request.policy_id = "{policy_id}"
        request.rule_id = "{rule_id}"
        response = client.delete_antileakage_rule(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
```

```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.DeleteAntileakageRuleRequest{}
request.PolicyId = "{policy_id}"
request.RuleId = "{rule_id}"
response, err := client.DeleteAntileakageRule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.53 查询引用表列表

功能介绍

查询引用表列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/valuelist

表 4-925 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID

表 4-926 Query 参数

参数	是否必选	参数类型	描述
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。
pagesize	否	Integer	分页查询时, 每页包含多少条结果。范围1-100, 默认值为10, 表示每页包含10条结果。
name	否	String	引用表名称

请求参数

表 4-927 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-928 响应 Body 参数

参数	参数类型	描述
total	Integer	引用表条数
items	Array of ValueListResponseBody objects	引用表列表

表 4-929 ValueListResponseBody

参数	参数类型	描述
id	String	引用表id
name	String	引用表名称
type	String	引用表类型
timestamp	Long	引用表时间戳
values	Array of strings	引用表的值
producer	Integer	引用表来源, 1代表用户创建, 其它值代表moduleX自动生成
description	String	引用表描述

状态码: 400

表 4-930 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-931 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-932 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-933 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-934 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-935 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的引用表列表

```
GET https://{Endpoint}/v1/{project_id}/waf/valuelist?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "id": "3b03be27a40b45d3b21fe28a351e2021",
    "name": "ip_list848",
    "type": "ip",
    "values": [ "100.100.100.125" ],
    "timestamp": 1650421866870,
    "producer": 1,
    "description": "demo"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListValueListSolution {

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

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListValueListRequest request = new ListValueListRequest();
try {
    ListValueListResponse response = client.listValueList(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListValueListRequest()
        response = client.list_value_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"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListValueListRequest{}  
    response, err := client.ListValueList(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.54 创建引用表

功能介绍

创建引用表，引用表能够被CC攻击防护规则和精准访问防护中的规则所引用。

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/valuelist

表 4-936 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-937 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-938 请求 Header 参数

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

表 4-939 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	引用表名称，只能由数字、字母、中划线、下划线和英文句点组成，长度不能超过64
type	是	String	引用表类型，参见枚举列表
values	是	Array of strings	引用表的值
description	否	String	引用表描述，最长128字符

响应参数

状态码： 200

表 4-940 响应 Body 参数

参数	参数类型	描述
id	String	引用表id
name	String	引用表名称
type	String	引用表类型
description	String	引用表描述
timestamp	Long	引用表时间戳
values	Array of strings	引用表的值
producer	Integer	引用表来源： <ul style="list-style-type: none">• 1:表示来源于用户手动创建• 2:表示来源于智能访问控制自动创建

状态码： 400

表 4-941 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-942 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-943 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-944 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-945 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-946 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中创建引用表, 引用表名称为“demo”, 引用表类型为“url”, 值为“/124”, 描述为“demo”。

```
POST https://{Endpoint}/v1/{project_id}/waf/valuelist?enterprise_project_id=0
```

```
{
  "name": "demo",
  "type": "url",
  "values": ["/124"],
  "description": "demo"
}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "e5d9032d8da64d169269175c3e4c2849",
  "name": "demo",
  "type": "url",
  "values": ["/124"],
  "timestamp": 1650524684892,
  "description": "demo",
  "producer": 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建引用表，引用表名称为“demo”，引用表类型为“url”，值为“/124”，描述为“demo”。

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

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

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

public class CreateValueListSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateValueListRequest request = new CreateValueListRequest();
        CreateValueListRequestBody body = new CreateValueListRequestBody();
        List<String> listbodyValues = new ArrayList<>();
        listbodyValues.add("/124");
        body.withDescription("demo");
        body.withValues(listbodyValues);
        body.withType(CreateValueListRequestBody.TypeEnum.fromValue("url"));
        body.withName("demo");
        request.withBody(body);
        try {
            CreateValueListResponse response = client.createValueList(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

在项目id为project_id的项目中创建引用表，引用表名称为“demo”，引用表类型为“url”，值为“/124”，描述为“demo”。

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = WafClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(WafRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = CreateValueListRequest()  
        listValuesbody = [  
            "/124"  
        ]  
        request.body = CreateValueListRequestBody(  
            description="demo",  
            values=listValuesbody,  
            type="url",  
            name="demo"  
        )  
        response = client.create_value_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

在项目id为project_id的项目中创建引用表，引用表名称为“demo”，引用表类型为“url”，值为“/124”，描述为“demo”。

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateValueListRequest{}
    var listValuesbody = []string{
        "/124",
    }
    descriptionCreateValueListRequestBody := "demo"
    request.Body = &model.CreateValueListRequestBody{
        Description: &descriptionCreateValueListRequestBody,
        Values: listValuesbody,
        Type: model.GetCreateValueListRequestBodyTypeEnum().URL,
        Name: "demo",
    }
    response, err := client.CreateValueList(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.55 查询引用表

功能介绍

查询引用表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/valuelist/{valuelistid}

表 4-947 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
valuelistid	是	String	引用表id，通过查询引用表列表（ListValueList）接口获取

表 4-948 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-949 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-950 响应 Body 参数

参数	参数类型	描述
id	String	引用表id
name	String	引用表名称
type	String	引用表类型
description	String	引用表描述
values	Array of strings	引用表的值
producer	Integer	引用表来源： <ul style="list-style-type: none">• 1:表示来源于用户手动创建• 2:表示来源于智能访问控制自动创建
timestamp	Long	创建规则的时间，格式为13位毫秒时间戳

状态码： 400

表 4-951 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-952 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-953 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-954 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-955 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-956 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，引用表id为valuelistid的引用表

```
GET https://{Endpoint}/v1/{project_id}/waf/valuelist/{valuelistid}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id": "63b1d9edf2594743bc7c6ee98527306c",
  "name": "RPmvp0m4",
  "type": "response_code",
  "values": [ "500" ],
  "description": "demo",
  "producer": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowValueListSolution {

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

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

```
        .withSk(sk);

    WafClient client = WafClient.newBuilder()
        .withCredential(auth)
        .withRegion(WafRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowValueListRequest request = new ShowValueListRequest();
    request.withValuelistid("{valuelistid}");
    try {
        ShowValueListResponse response = client.showValueList(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ShowValueListRequest()
        request.valuelistid = "{valuelistid}"
        response = client.show_value_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"
waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.56 修改引用表

功能介绍

修改引用表

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/valuelist/{valuelistid}

表 4-957 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
valuelistid	是	String	引用表id，通过查询引用表列表（ListValueList）接口获取

表 4-958 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-959 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)。
Content-Type	是	String	内容类型

表 4-960 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	引用表名称，2-32位字符串组成
type	是	String	引用表类型，参见枚举列表
values	否	Array of strings	引用表的值
description	否	String	引用表描述，最长128字符

响应参数

状态码： 200

表 4-961 响应 Body 参数

参数	参数类型	描述
id	String	引用表id
name	String	引用表名称
type	String	引用表类型
description	String	引用表描述
values	Array of strings	引用表的值
producer	Integer	引用表来源： <ul style="list-style-type: none"> 1:表示来源于用户手动创建 2:表示来源于智能访问控制自动创建

状态码： 400

表 4-962 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-963 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-964 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-965 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-966 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-967 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, 引用表id为valuelistid的引用表, 引用表名称为“RPMvp0m4”, 引用表类型为“response_coderl”, 值为“500”, 描述为“demo”。

```
PUT https://{Endpoint}/v1/{project_id}/waf/valuelist/{valuelistid}?enterprise_project_id=0
```

```
{
  "name": "RPMvp0m4",
  "type": "response_code",
  "values": [ "500" ],
  "description": "demo"
}
```

响应示例

状态码: 200

请求成功

```
{
  "id": "63b1d9edf2594743bc7c6ee98527306c",
  "name": "RPMvp0m4",
  "type": "response_code",
  "values": [ "500" ],
  "description": "demo",
  "producer": 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，引用表id为valuelistid的引用表，引用表名称为“Rpmvp0m4”，引用表类型为“response_coderl”，值为“500”，描述为“demo”。

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

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

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

public class UpdateValueListSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateValueListRequest request = new UpdateValueListRequest();
        request.withValuelistid("{valuelistid}");
        UpdateValueListRequestBody body = new UpdateValueListRequestBody();
        List<String> listbodyValues = new ArrayList<>();
        listbodyValues.add("500");
        body.withDescription("demo");
        body.withValues(listbodyValues);
        body.withType(UpdateValueListRequestBody.TypeEnum.fromValue("response_code"));
        body.withName("Rpmvp0m4");
        request.withBody(body);
        try {
            UpdateValueListResponse response = client.updateValueList(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
        }
    }
}
```

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

Python

更新项目id为project_id，引用表id为valuelistid的引用表，引用表名称为“RPmvp0m4”，引用表类型为“response_coderl”，值为“500”，描述为“demo”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateValueListRequest()
        request.valuelistid = "{valuelistid}"
        listValuesbody = [
            "500"
        ]
        request.body = UpdateValueListRequestBody(
            description="demo",
            values=listValuesbody,
            type="response_code",
            name="RPmvp0m4"
        )
        response = client.update_value_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

更新项目id为project_id，引用表id为valuelistid的引用表，引用表名称为“RPmvp0m4”，引用表类型为“response_coderl”，值为“500”，描述为“demo”。

```
package main

import (
    "fmt"
```

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateValueListRequest{}
    request.ValueListId = "{valueListId}"
    var listValuesbody = []string{
        "500",
    }
    descriptionUpdateValueListRequestBody := "demo"
    request.Body = &model.UpdateValueListRequestBody{
        Description: &descriptionUpdateValueListRequestBody,
        Values: &listValuesbody,
        Type: model.GetUpdateValueListRequestBodyTypeEnum().RESPONSE_CODE,
        Name: "Rpmvp0m4",
    }
    response, err := client.UpdateValueList(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.4.57 删除引用表

功能介绍

删除引用表

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/valuelist/{valuelistid}

表 4-968 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
valuelistid	是	String	引用表id，通过查询引用表列表（ListValueList）接口获取

表 4-969 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-970 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-971 响应 Body 参数

参数	参数类型	描述
id	String	引用表id
name	String	引用表名称
type	String	引用表类型
timestamp	Long	删除引用表的时间，时间为13位毫秒时间戳

状态码： 400

表 4-972 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-973 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-974 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-975 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-976 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-977 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id，引用表id为valuelistid的引用表

```
DELETE https://{Endpoint}/v1/{project_id}/waf/valuelist/{valuelistid}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "id": "63b1d9edf2594743bc7c6ee98527306c",
  "name": "RPmvp0m4",
  "type": "response_code",
  "timestamp": 1640938602391
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class DeleteValueListSolution {

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

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

```
WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
DeleteValueListRequest request = new DeleteValueListRequest();
request.withValuelistid("{valuelistid}");
try {
    DeleteValueListResponse response = client.deleteValueList(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteValueListRequest()
        request.valuelistid = "{valuelistid}"
        response = client.delete_value_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build()  
    )  
  
    request := &model.DeleteValueListRequest{}  
    request.ValueListId = "{valueListId}"  
    response, err := client.DeleteValueList(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.5 地址组管理

4.5.1 查询地址组列表

功能介绍

查询地址组列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/ip-groups

表 4-978 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-979 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
page	否	Integer	页码，默认值为1
pagesize	否	Integer	每页的条数，单页条数限制100，默认值为10
name	否	String	ip地址组名称，支持模糊查询
ip	否	String	ip地址或ip段，模糊匹配包含这个IP的IP地址或IP段地址组

请求参数

表 4-980 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)。

参数	是否必选	参数类型	描述
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-981 响应 Body 参数

参数	参数类型	描述
total	Integer	该用户当前企业项目下Ip地址组数量，只包含本地地址组
items	Array of IpGroupBody objects	地址组信息列表
cloudTotal	Integer	该用户总的Ip地址组数量，包含本地与共享地址组

表 4-982 IpGroupBody

参数	参数类型	描述
id	String	地址组id
name	String	地址组名称
ips	String	地址组ip（以逗号分隔的ip或ip段）
size	Integer	地址组长度
rules	Array of RuleInfo objects	ip地址组绑定的规则列表
share_info	ShareInfo object	共享信息
description	String	地址组描述

表 4-983 RuleInfo

参数	参数类型	描述
rule_id	String	规则id
rule_name	String	规则名称
policy_id	String	策略id

参数	参数类型	描述
policy_name	String	策略名称

表 4-984 ShareInfo

参数	参数类型	描述
share_count	Integer	共享用户数
accept_count	Integer	接受用户数
process_status	Integer	处理状态

状态码： 400

表 4-985 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-986 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-987 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-988 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-989 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-990 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的地址组列表

```
GET https://{Endpoint}/v1/{project_id}/waf/ip-groups?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "description": "",
    "id": "c36e896b18ee486a81026fce8e69fb1a",
    "ips": "xxx.xx.xx.xx",
    "name": "sfddf",
    "rules": [ ],
    "share_info": {
      "accept_count": 0,
      "process_status": 0,
      "share_count": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListIpGroupSolution {

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



```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListIpGroupRequest request = new ListIpGroupRequest();
try {
    ListIpGroupResponse response = client.listIpGroup(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListIpGroupRequest()
        response = client.list_ip_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.5.2 创建 ip 地址组

功能介绍

创建ip地址组

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/ip-groups

表 4-991 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户的project_id

表 4-992 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

请求参数

表 4-993 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)。
Content-Type	是	String	内容类型

表 4-994 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	地址组名称
ips	是	String	以逗号分隔的ip或ip段

参数	是否必选	参数类型	描述
description	否	String	地址组描述

响应参数

状态码： 200

表 4-995 响应 Body 参数

参数	参数类型	描述
id	String	地址组id
name	String	地址组名称
ips	String	地址组ip（以逗号分隔的ip或ip段）
size	Integer	地址组长度
rules	Array of RuleInfo objects	ip地址组绑定的规则列表
description	String	地址组描述
timestamp	Long	时间戳

表 4-996 RuleInfo

参数	参数类型	描述
rule_id	String	规则id
rule_name	String	规则名称
policy_id	String	策略id
policy_name	String	策略名称

状态码： 400

表 4-997 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-998 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-999 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1000 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1001 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1002 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中创建ip地址组。ip地址组的名称为"group3", 地址为"xx.xx.xx.xx", 地址组描述为 "demo"。

```
POST https://{Endpoint}/v1/{project_id}/waf/ip-groups?enterprise_project_id=0
```

```
{
  "name": "group3",
  "ips": "xx.xx.xx.xx",
  "description": "demo"
}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "c36e896b18ee486a81026fce8e69fb1a",
  "ips": "xx.xx.xx.xx",
  "name": "group3",
  "rules": [ ],
  "size": 1,
  "timestamp": 1666747418345,
  "description": "demo"
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建ip地址组。ip地址组的名称为"group3"，地址为"xx.xx.xx.xx"，地址组描述为“demo”。

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

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

public class CreateIpGroupSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateIpGroupRequest request = new CreateIpGroupRequest();
        CreateIpGroupRequestBody body = new CreateIpGroupRequestBody();
        body.withDescription("demo");
        body.withIps("xx.xx.xx.xx");
        body.withName("group3");
        request.withBody(body);
        try {
            CreateIpGroupResponse response = client.createIpGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

在项目id为project_id的项目中创建ip地址组。ip地址组的名称为"group3"，地址为"xx.xx.xx.xx"，地址组描述为“demo”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateIpGroupRequest()
        request.body = CreateIpGroupRequestBody(
            description="demo",
            ips="xx.xx.xx.xx",
            name="group3"
        )
        response = client.create_ip_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

在项目id为project_id的项目中创建ip地址组。ip地址组的名称为"group3"，地址为"xx.xx.xx.xx"，地址组描述为“demo”。

```
package main

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

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



```
projectId := "{project_id}"

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

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateIpGroupRequest{
    descriptionCreateIpGroupRequestBody:= "demo"
    request.Body = &model.CreateIpGroupRequestBody{
        Description: &descriptionCreateIpGroupRequestBody,
        Ips: "xx.xx.xx.xx",
        Name: "group3",
    }
}
response, err := client.CreateIpGroup(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.5.3 查询 ip 地址组明细

功能介绍

查询ip地址组明细

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/ip-group/{id}

表 4-1003 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
id	是	String	ip地址组id

表 4-1004 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1005 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1006 响应 Body 参数

参数	参数类型	描述
id	String	地址组id
name	String	地址组名称
ips	String	地址组ip（以逗号分隔的ip或ip段）

参数	参数类型	描述
size	Integer	地址组长度
rules	Array of RuleInfo objects	ip地址组绑定的规则列表
share_info	ShareInfo object	共享信息
description	String	地址组描述

表 4-1007 RuleInfo

参数	参数类型	描述
rule_id	String	规则id
rule_name	String	规则名称
policy_id	String	策略id
policy_name	String	策略名称

表 4-1008 ShareInfo

参数	参数类型	描述
share_count	Integer	共享用户数
accept_count	Integer	接受用户数
process_status	Integer	处理状态

状态码： 400

表 4-1009 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1010 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1011 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1012 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1013 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1014 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 地址组id为id的地址组。

```
GET https://{Endpoint}/v1/{project_id}/waf/ip-group/{id}?enterprise_project_id=0
```

响应示例

状态码: 200

请求成功

```
{
  "description": "",
  "id": "c36e896b18ee486a81026fce8e69fb1a",
  "ips": "xx.xx.xx.xx",
  "name": "sfddf",
  "rules": [ ],
  "size": 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;
import com.huaweicloud.sdk.core.auth.ICredential;
```

```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowIpGroupSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowIpGroupRequest request = new ShowIpGroupRequest();
        request.withId("{id}");
        try {
            ShowIpGroupResponse response = client.showIpGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

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

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

try:
    request = ShowIpGroupRequest()
    request.id = "{id}"
    response = client.show_ip_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

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

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.5.4 修改 ip 地址组

功能介绍

修改ip地址组

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/ip-group/{id}

表 4-1015 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
id	是	String	ip地址组id

表 4-1016 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
action	否	String	增量修改ip地址组时，此为必传字段，传入“add”；删除一个或者多个ip时传入“delete”

请求参数

表 4-1017 请求 Header 参数

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

表 4-1018 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	地址组名称
ips	是	String	以逗号分隔的ip或ip段
description	否	String	地址组描述

响应参数

状态码： 200

表 4-1019 响应 Body 参数

参数	参数类型	描述
id	String	地址组id
name	String	地址组名称
ips	String	地址组ip（以逗号分隔的ip或ip段）
size	Integer	地址组长度
rules	Array of RuleInfo objects	ip地址组绑定的规则列表
description	String	地址组描述

表 4-1020 RuleInfo

参数	参数类型	描述
rule_id	String	规则id
rule_name	String	规则名称
policy_id	String	策略id
policy_name	String	策略名称

状态码： 400

表 4-1021 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1022 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1023 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1024 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1025 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1026 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，地址组id为id的ip地址组。ip地址组的名称为"demo"，地址为"xx.xx.xx.xx"，地址组描述为“ ”。

```
PUT https://{Endpoint}/v1/{project_id}/waf/ip-group/{id}?enterprise_project_id=0
{
  "ips": "xx.xx.xx.xx",
  "name": "demo",
  "description": ""
}
```

响应示例

状态码： 200

请求成功

```
{
  "description": "",
  "id": "c36e896b18ee486a81026fce8e69fb1a",
  "ips": "xx.xx.xx.xx",
  "name": "demo",
  "size": 1,
  "rules": [ ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，地址组id为id的ip地址组。ip地址组的名称为"demo"，地址为"xx.xx.xx.xx"，地址组描述为“ ”。

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

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

public class UpdateIpGroupSolution {

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

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

```
WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateIpGroupRequest request = new UpdateIpGroupRequest();
request.withId("{id}");
UpdateIpGroupRequestBody body = new UpdateIpGroupRequestBody();
body.withDescription("");
body.withIps("xx.xx.xx.xx");
body.withName("demo");
request.withBody(body);
try {
    UpdateIpGroupResponse response = client.updateIpGroup(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新项目id为project_id，地址组id为id的ip地址组。ip地址组的名称为"demo"，地址为"xx.xx.xx.xx"，地址组描述为“ ”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateIpGroupRequest()
        request.id = "{id}"
        request.body = UpdateIpGroupRequestBody(
            description="",
            ips="xx.xx.xx.xx",
            name="demo"
        )
        response = client.update_ip_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

更新项目id为project_id，地址组id为id的ip地址组。ip地址组的名称为"demo"，地址为"xx.xx.xx.xx"，地址组描述为“ ”。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateIpGroupRequest{}
    request.Id = "{id}"
    descriptionUpdateIpGroupRequestBody := ""
    request.Body = &model.UpdateIpGroupRequestBody{
        Description: &descriptionUpdateIpGroupRequestBody,
        Ips: "xx.xx.xx.xx",
        Name: "demo",
    }
    response, err := client.UpdateIpGroup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.5.5 删除 ip 地址组

功能介绍

删除ip地址组

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/ip-group/{id}

表 4-1027 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
id	是	String	ip地址组id

表 4-1028 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1029 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1030 响应 Body 参数

参数	参数类型	描述
id	String	地址组id
name	String	地址组名称
ips	String	地址组ip（以逗号分隔的ip或ip段）
size	Integer	地址组长度
rules	Array of RuleInfo objects	ip地址组绑定的规则列表

表 4-1031 RuleInfo

参数	参数类型	描述
rule_id	String	规则id
rule_name	String	规则名称
policy_id	String	策略id
policy_name	String	策略名称

状态码： 400

表 4-1032 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1033 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1034 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1035 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1036 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1037 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 地址组id为id的地址组。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/ip-group/{id}?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{  
  "description": "",  
  "id": "c36e896b18ee486a81026fce8e69fb1a",  
  "ips": "xx.xx.xx.xx",
```

```
"name" : "demo",  
"size" : 1,  
"rules" : [ ]  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class DeletelpGroupSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeletelpGroupRequest request = new DeletelpGroupRequest();  
        request.withId("{id}");  
        try {  
            DeletelpGroupResponse response = client.deletelpGroup(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = DeleteIpGroupRequest()
        request.id = "{id}"
        response = client.delete_ip_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteIpGroupRequest{}
```

```
request.Id = "{id}"
response, err := client.DeletelpGroup(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

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

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6 证书管理

4.6.1 查询证书列表

功能介绍

查询证书列表

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/certificate

表 4-1038 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1039 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
page	否	Integer	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。
name	否	String	证书名称
host	否	Boolean	是否获取证书关联的域名，默认为false -true:获取已关联域名的证书 -false:获取未关联域名的证书
exp_status	否	Integer	证书过期状态，0-未过期，1-已过期，2-即将过期（证书将在一个月内过期）

请求参数

表 4-1040 请求 Header 参数

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

响应参数

状态码： 200

表 4-1041 响应 Body 参数

参数	参数类型	描述
items	Array of CertificateBody objects	证书列表
total	Integer	证书总数

表 4-1042 CertificateBody

参数	参数类型	描述
id	String	证书id
name	String	证书名
expire_time	Long	证书过期时间戳
exp_status	Integer	证书过期状态，0-未过期，1-已过期，2-即将过期（一个月内即将过期）
timestamp	Long	证书上传时间戳
bind_host	Array of BindHost objects	证书关联的域名信息

表 4-1043 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400

表 4-1044 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1045 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1046 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1047 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1048 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1049 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id证书列表。

```
GET https://{Endpoint}/v1/{project_id}/waf/certificate?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{  
  "total": 1,  
  "items": [{  
    "id": "dc443ca4f29c4f7e8d4adaf485be317b",
```

```
"name" : "demo",  
"timestamp" : 1643181401751,  
"expire_time" : 1650794100000,  
"bind_host" : [ ],  
"exp_status" : 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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class ListCertificatesSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListCertificatesRequest request = new ListCertificatesRequest();  
        try {  
            ListCertificatesResponse response = client.listCertificates(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = ListCertificatesRequest()
        response = client.list_certificates(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

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

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6.2 创建证书

功能介绍

创建证书

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/certificate

表 4-1050 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1051 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1052 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)。
Content-Type	是	String	内容类型

表 4-1053 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	证书名称，证书名称只能由数字、字母、中划线、下划线和英文句点组成，长度不能超过256位字符
content	是	String	证书文件，仅支持PEM格式的证书和私钥文件，且文件中的换行符应以\n替换，如请求示例所示
key	是	String	证书私钥，仅支持PEM格式的证书和私钥文件，且文件中的换行符应以\n替换，如请求示例所示

响应参数

状态码： 200

表 4-1054 响应 Body 参数

参数	参数类型	描述
id	String	证书ID

参数	参数类型	描述
name	String	证书名
content	String	证书文件，PEM编码
key	String	证书私钥，PEM编码
expire_time	Long	证书过期时间戳
exp_status	Integer	证书过期状态，0-未过期，1-已过期，2-即将过期
timestamp	Long	证书上传时间戳
bind_host	Array of BindHost objects	证书关联的域名信息

表 4-1055 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400

表 4-1056 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1057 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1058 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1059 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1060 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的 decode-authorization-message 接口进行解码, 可参考 STS5 联调自验证。IAM5 鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5 鉴权错误时才会返回此字段。

表 4-1061 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目 id 为 project_id 的项目中创建证书, 证书名称为 "demo", 证书内容为 "-----BEGIN CERTIFICATE----- ", 证书密钥为 "-----BEGIN PRIVATE KEY-----"。

POST https://{Endpoint}/v1/{project_id}/waf/certificate?enterprise_project_id=0

```
{
  "name": "demo",
  "content": "-----BEGIN CERTIFICATE----- \
\nMIIIDyzCCArOgAwIBAgIJAN5U0Z4Bh5ccMA0GCSqGSIb3DQEBCwUAMHwx CzAJBgNV
BAYTAplMIRIwEAYDVQQIDAIHVUFOR0RPTkcxETAPBgNVBACMCERPTkdHVUFOMQ0w
CwYDVQQKDARERUtFMQswCQYDVQQLEDAJESzELMAkGA1UEAwwCT0QxHTAbBgkqhkiG
9w0BCQEWDk8IZC5odWF3Z3WkuY29tMB4XDTIxMTEwNTA4MTk0MVoXDTIyMTEwNTA4
MTk0MVoYfDELMkGA1UEBhMCWkgx EJAQBgNVBAGMCUdVQU5HRE9ORzERMA8GA1UE
BwY*****VQQD
DAJPRDEdMBsGCSqGSIb3DQEJARYOTwhkLmh1YXdlaS5jb20wggEiMA0GCSqGSIb3
DQEBAQUAA4IBDwAwggEKAoIBAQCdcoLfk62//r0RHFyweYBj97S4Nsj8Qj0RG+Y02
OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn1PcN2Pj2vPJD6NEk4I6VdOWr /
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2OyBxT+ocTLRgEIB8ZbvJyPNseg
feLE*****SusXo FQ/WRbBRH7DrQmxGiXsq4VELER9Nnc/Kywyq+9pYi8L
+mKeRL+lcmMbxXC/3k6OfMB tVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE
+tmJZTPH6lYtAgMBAAGjUDBOMB0G A1UdDgQWBQBprUUFxw
+gIkpzXdrYlsWjSahWjAfbGvNVHSMEDAWgBQprUUFxw+g lkpzXdrYlsWjSahWjAMBgNVHRMEBTADAQH/
MA0GCSqGSIb3DQEBCwUAA4IBAQA2 603K*****iOhU o/
kVwkiUlCwx4t7RwP0hVms0OZw59MuqKd3oCSWkYO4vEHs3t40JDWnGDnmQ4sol
RkOWJwL4w8tnPe3qY9JSupjlsu6Y1hlvKtEfN2vEKFnsuMhidkUpUAJWodHhWBQH
wglDo4/6yTnWZNGK8JDal86Dm5lchXea1EoYBjsHxiJb7HeWQlkre+MCYi1RHOin 4mIXTr0t4/jWlglSz6/
ZhGRq+7W7tl7cvzCe+4XsVZlenAcYoNd/WLfo91PD4 yAsRxoJw1so1Bj0BkDz\
\n -----END CERTIFICATE-----",
  "key": "-----BEGIN PRIVATE KEY----- \
\nMIIEVwIBADANBgkqhkiG9w0BAQEFAASCBKkwggSlAgEAAoIBAQCdcoLfk62//r0RH FyweYBj97S4Nsj8Qj0RG
+Y02OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn 1PcN2Pj2vPJD6NEk4I6VdOWr/
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2Oy BxT
+ocTLRgEIB8ZbvJyPNsegfeLEUuPYRpQ5kXLgJH2/3NwZFogBHVv/b07l4fr+ sWJMniA2yijSBQ0DEAOSusXoFQ/
WRbBRH7DrQmxGiXsq4VELER9Nnc/Kywyq+9pYi 8L+mKeRL+lcmMbxXC/
3k6OfMBtVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE+tmJ ZTPH6lYtAgMBAEcgEBAL+xZxm/QoqXT
+2stoqV2GEYaMFASpRqxl0cZMmEE/9 jZa+cBWljHhVpsjRqYFBDChEebu0JwlrjciAvgnlVnO5XgXm1A9Q
+WbscokmcX1 xCvpHgc+MDVn+uWdCd4KW5kEk4EnSsFN5iNSf+1VxNURN+gwSSp/0E+muwA5IISO
G6*****lcGO1n 4urPBHuNly04GgGw
```



```
+vkaqjqOvZrNLVOMMaFWBxsDWBehgSSBQTj+f3NCxneGYtt8 3SCTZQI5nlkb+r/  
M455EwKTSXuEsNHolwx7L6GEPbQECgYEA8lxgK2fYyklolCoh  
TFJaRAvyjyKa2+Aza4qT9SGY9Y30VPClPjBB1vUu5M9KrFufzlv06nGEcHmpEwOe  
8vbRu7nLAQTGYFi8VK63q8w6FIldAyCG6Sx+BWCfWxJzXsZLAJTfklwi8HsOSlqh  
6QN*****fvx mTbLG52Z21OyocNq3Tf/  
b1Zwolc1ik6cyBzY6z1blrbSzArCqm0sb2iD+kjL81O0 /qqdXjBxZUkKiVAMNNp7xjGZHHFKWUxT2+UX/  
tlyx4tT4dZrFlkdDXkcMmqfsRxd 1NEVaAaT8SECgYaoU7Bptplun43YTpUfr3pSIN6oZeKoxSbw9i4MMNC  
+4fSDRPC+ 80ImcmZRL7taF+Y7p0jxAOTulkdJC8NbAiv5J9WzrwQ+5MF2BPP/2bYnRa6tNofH  
kZD*****QC9 ihjZTj/bTHtRiHZppzCvyYm/lgd  
+Uwtsy0uXR1n0G1SQENgrTBD/J6AzdJae6tE P0U8YIM5Oqxf2i/as9ay+IPRecMl4eSxz7jJWAGx6Yx/3AZ  
+hAB1ZbNbnqniCLYNk d0MvjwmA25ATO+ro4OZ7AdEpQbk3l9aG/WFyYBz9AQKBgQCucFPA1l5eslL8196V  
WMr2Qo0tqzL7CGSoWQk2Sa2HZtZdfofXAaaqo+zvJ6RPHUj0jgJtx536DvV3egI  
37YrdQyJbCPZXQ3SPgqWCORUnXBWq/nxS06uwu6JBxUfC57ijmMU4fWYNrvkkmWb 7keAg/  
r5Uy1joMAvBN1l6lB8pg==\n -----END PRIVATE KEY-----"  
}
```

响应示例

状态码： 200

OK

```
{  
  "id" : "64af92e2087d49cbabc233e9bdc761b7",  
  "name" : "testly",  
  "timestamp" : 1658994431596,  
  "expire_time" : 1682394560000  
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建证书，证书名称为 "demo"，证书内容为 "-----BEGIN CERTIFICATE----- "，证书密钥为"-----BEGIN PRIVATE KEY-----"。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class CreateCertificateSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);
```

```

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
CreateCertificateRequest request = new CreateCertificateRequest();
CreateCertificateRequestBody body = new CreateCertificateRequestBody();
body.withKey("-----BEGIN PRIVATE KEY-----
\nMIIEvwIBADANBgkqhkiG9w0BAQEFAASCCKkwggSlAgEAAoIBAQCdcoLFK62//r0RH FyweYBj97S4NsJ8Qj0RG
+Y02OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn 1PcN2Pj2vPJD6NEk4I6VdOWr/
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2Oy BxT
+ocTLRgElB8ZbvJyPNsegfeLEUuPYRpQ5kXLgJH2/3NwZFOgBHvV/b07l4fR+ sWJMnIA2yljSBQ0DEAOSusXoFQ/
WRbBRH7DrQmxGiXsq4VELer9Nnc/Kywq+9pYi 8L+mKeRL+lcMMbXC/
3k6OfMBtVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE+tmJ ZTPH6lYtAgMBAEcgEBAL+xZxm/QoqxT
+2stoqV2GEYaMFASpRqxlqjZMmEE/9 jZa+cBWlJHhVPsjRqYFBDcHEebu0JwlrjclAvgnIvn05XgXm1A9Q
+WbscokmcX1 xCvpHgc+MDVn+uWdCd4KW5kEk4EnSsFN5iNSf+1VxNURN+gwSSp/0E+muwA5IISO
G6*****lcGO1n 4urPBHuNly04GgGw
+vkaqjQvZrNLVOMMaFWBxsDWBehgSSBQTj+f3NCxneGYtt8 3SCTZQI5nlkb+r/
M455EwKTSXuESNHolwx7L6GEPbQECgYEA8lxgK2fYyKlclCoh
TFJaRAVvyjKa2+Aza4qT9SGY9Y30VPClPjBB1vUu5M9KrFufzlv06nGEcHmpEwOe
8vbRu7nLAQTGYFi8VK63q8w6FIEdAyCG6Sx+BWCfWxJzXSLAJTfklwi8HsOSlqh
6QN*****fvx mTbLG5Z2Z1OyocNq3Tf/
b1Zwolc1ik6cyBzY6z1blrbSzArCqm0sb2iD+kjL8100 /qqdXjBxZUkKiVAMNNp7xjGZHHFKWUxT2+UX/
tlyx4T4dZrFlkdDXkcMmqfsRxd 1NEVaAaT8SECgYAOu7BPTplun43YTpUfr3pSIN6oZeKoxSbw9i4MNC
+4fSDRPC+ 80lmcmZRL7taF+Y7p0jxAOTulkdJC8NbAiv5J9WzrwQ+5MF2BPB/2bYnRa6tNofH
kZD*****QC9 ihjZTj/bThtRiHzppzCvyYm/lgd
+Uwtsy0uXR1n0G1SQENgrTBD/J6AzdFJae6tE P0U8YIM5Oqx2f/as9ay+IPRecMl4eSxz7jWAGx6Yx/3AZ
+hAB1ZbNbnqniCLYNk d0MvjwmA25ATO+ro4OZ7AdEpQbk3l9aG/WfYBz9AQKBgQCucFPA115eslL8196V
WMr2Q0tqzL7CGSoWQk2Sa2HZtZdfofXAaaqo+zvJ6RPHUj0jgJtx536DvV3egI
37YrdQyJbCPZXQ3SPgqWCORUnXBWq/nxS06uwu6JBxUfC57ijmMU4fWYNrvkkmWb 7keAg/
r5Uy1joMAvBN1I6lB8pg==\n -----END PRIVATE KEY-----");
body.withContent("-----BEGIN CERTIFICATE-----
\nMIIDyzCCARogAwIBAgIJAN5U0Z4Bh5ccMA0GCSqGSIb3DQEBCwUAMHwxZAJBgNV
BAYTA1RlMRlWAEYDVQQLIDALHVUFOR0RPTkcxETAPBgNVBACMCERPTkdHVUJFOMQ0w
CwYDVQQKDARERUUtFMQswCQYDVQQLDAJESzELMAkGA1UEAwwCT0QxHTAbBgkqhkiG
9w0BQCQEWdK8IZCSodWF3ZwkuY29tMB4XDlXMTExNTA4MTk0MVoXDTIyMTExNTA4
MTk0MVoVowfDELMakGA1UEBhMCWkgxEjAQBgNVBAgMCUdVQU5HRE9ORzERMA8GA1UE
Bw*****VQQD
DAJPRDEdMBsGCSqGSIb3DQEJARYOTwhkLmh1YXdlas5jb2wggEiMA0GCSqGSIb3
DQEBAQUAA4IBDwAwggEKAoIBAQCdcoLFK62//r0RHFyweYBj97S4NsJ8Qj0RG+Y02
OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn1PcN2Pj2vPJD6NEk4I6VdOWr /
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2OyBxT+ocTLRgElB8ZbvJyPNseg
feLE*****SusXo FQ/WRbBRH7DrQmxGiXsq4VELer9Nnc/Kywq+9pYi8L
+mKeRL+lcMMbXC/3k6OfMB tVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE
+tmJZTPH6lYtAgMBAAGjUDBOMB0G A1UdDgQWBBQprUUFxW
+gIkpzXdrYlsWjSahWjAfBgNVHSMEGDAWgBQprUUFxW+g IkpzXdrYlsWjSahWjAMBgNVHRMEBTADAQH/
MA0GCSqGSIb3DQEBCwUAA4IBAQA2 603K*****iOhU o/
kVvkiUlcxw4t7RwP0hVms0OZw59MuqKd3oCSWkYO4vEHs3t40JDWnGDnmQ4sol
RkOWJwL4w8tnPe3qY9JSupjlsu6Y1hlvKtEfN2vEKFnsuMhidkUpUAJWodHhWBQH
wglDo4/6yTnWZNGK8JDal86Dm5IchXea1EoYBjsHxiJb7HeWQlkre+MCYi1RHOin 4mlXTr0oT4/jWlglkSz6/
ZhGRq+7W7tl7cvzCe+4XsVZlenAcYoNd/WLfo91PD4 yAsRXrOjW1so1Bj0BkDzn -----END CERTIFICATE-----");
body.withName("demo");
request.withBody(body);
try {
    CreateCertificateResponse response = client.createCertificate(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
}
}

```

Python

在项目id为project_id的项目中创建证书，证书名称为 "demo"，证书内容为 "-----BEGIN CERTIFICATE----- "，证书密钥为"-----BEGIN PRIVATE KEY-----"。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = CreateCertificateRequest()
        request.body = CreateCertificateRequestBody(
            key="-----BEGIN PRIVATE KEY-----
\nMIIIEvwlBADANBgkqhkiG9w0BAQEFAASCBKkwggSIAgEAAoIBAQCdcoLfk62//r0RH FyweYBj97S4NsJ8Qj0RG
+Y02OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn 1PcN2Pj2vPJD6NEk4I6VdOWr/
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2Oy BxT
+ocTLRgElB8ZbvJyPNsegfeLEUuPYRqQ5kXlgJH2/3NwZFOgBHVv/b07l4fR+ sWJMnIA2yljSBQ0DEAOSusXoFQ/
WRbBRH7DrQmxGiXsq4VELEr9Nnc/Kywq+9pYi 8L+mKeRL+lCMMbXC/
3k6OfMBVTiwcms1Mkr3iG03i8u6H7RSvRwyBz9G9sE+tmJ ZTPH6lYtAgMBAEECggEBAL+xZxm/QoqXT
+2stoqV2GEyAMFASpRqxlzCZmEE/9 jZa+cBWlJHhVPsjRqYFBDcHEebu0JwlrjclAvgnlVnO5XgM1A9Q
+WbscokmC1 xCvpHgc+MDVn+uWdCd4KW5kEk4EnSsFN5iNSf+1VxNURN+gwSSp/OE+muwA5IISO
G6*****lcGO1n 4urPBHuNly04GgGw
+vkaqjQvZrNLVOMMaFWBxsDWBehgSSBQTj+f3NCxneGYtt8 3SCTZQI5nlkbr/
M455EwKTSXuEsNHolwx7L6GEPbQECgYEA8lxgK2fYykolCoh
TFJaRAVvyjKa2+Aza4qT9SGY9Y30VPClPjBB1vUu5M9KrFufzlv06nGEcHmpEwOe
6vbRu7nLAQTGYFi8VK63q8w6FlFdAyCG6Sx+BWCfWxJzXsZLAJTfklwi8HsOslqH
8QN*****fvx mTbLG5Z2Z1OyocNq3Tf/
b1Zwolc1ik6cyBzY6z1blrbSzArCqm0sb2iD+kjL8100 /qqdXjBxZUkKiVAMNnp7xjGZHHFKWUxT2+UX/
tlyx4t4dzrFlkdDXkcMmqfsRxd 1NEVaAaT8SECgYAOu7Bptplun43YtPfUfr3pSIN6oZeKoxSbw9i4MNC
+4fSDRPC+ 80ImcmZRL7taF+Y7p0jxAOtUlkdJC8NbAiv5J9WzrwQ+5MF2BPB/2bYnRa6tNofH
kZD*****QC9 ihZTj/bTHtRiHZppzCvyYm/lgd
+Uwtsy0uXR1n0G1SQENgrTBD/J6AzdfJae6tE P0U8YIM5Oqxf2i/as9ay+IPRecMl4eSxz7jJWAGx6Yx/3AZ
+hAB1ZbNbnqicLYNk d0MvjwmA25ATO+ro4OZ7AdEpQbk3l9aG/WFyYBz9AQKBgQCucFPA1l5eslL8196V
WMr2Qo0tqzL7CGSoWQk2Sa2HZtZdfXAaaqo+zvJ6RPHJh0jgJtx536DvV3egl
37YrdQyJbCPZXQ3SPgqWCORUnXBwq/nxS06uwu6JBxUfC57ijmMU4fWYnrVkkMwB 7keAg/
r5Uy1joMAvBN1i6lB8pg=
\n -----END PRIVATE KEY-----",
            content="-----BEGIN CERTIFICATE-----
\nMIIIDyzCCArOgAwIBAgIJAN5U0Z4Bh5ccMA0GCSqGSIb3DQEBCwUAMHwxZAJBgNV
BAYTAkRlEAYDVQQIDAIHVUFOR0RPTkcxETAPBgNVBACMCEPRTkdHVUFOMQ0w
CwYDVQQKDARERUtuFMQswCQYDVQQLDAJESZELMAkGA1UEAwwCT0xHTAbGkqhkiG
9w0BCQEWDk8IZC5odWF3Z3ZwkuY29tMB4XDTEyMTExMTA4MTk0MVoXDTIyMTExNTA4
MTk0MVoVowfDELMakGA1UEBhMCWkxgEjAQBgNVBAgMCUdVQU5HRE9ORzERMA8GA1UE
Bw*****VQQD
DAJPRDEdMBsGCSqGSIb3DQEJARYOTwhkLmh1YXdlas5jb20wggEiMA0GCSqGSIb3
DQEBAQUAA4IBDwAwggEKAoIBAQCdcoLfk62//r0RHFyweYBj97S4NsJ8Qj0RG+Y02
OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn1PcN2Pj2vPJD6NEk4I6VdOWr /
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2OyBxT+ocTLRgElB8ZbvJyPNseg
feLE*****SusXo FQ/WRbBRH7DrQmxGiXsq4VELEr9Nnc/Kywq+9pYi8L
```

```
+mKeRL+lcMMbXC/3k6OfMB tVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE
+tmJZTPH6lYtAgMBAAGjUDBOMB0G A1UdDgQWBBQprUUFxW
+gIkpzXdrYlsWjfsahWjAfbgNVHSMEDAWgBQprUUFxW+g IkpzXdrYlsWjfsahWjAMBgNVHMRMEBTADAQH/
MA0GCSqGSIb3DQEBCwUAA4IBAQA2 603K*****iOhU o/
kVwkiUlCw4t7RwP0hVms0OZw59MuqKd3oCSWkYO4vEHs3t40JDWnGDnmQ4sol
RkOWJwL4w8tnPe3qY9JSupjlsu6Y1hlvKtEfN2vEKFnsuMhidkUpUAJWodHhWBQH
wglDo4/6yTnWZNGK8JDal86Dm5lchXea1EoYBjsHxib7HeWQlkre+MCYi1RHOin 4mIXTr0oT4/jWlglkSz6/
ZhGRq+7W7tl7cvzCe+4XsvZlenAcYoNd/WLfo91PD4 yAsRxoJw1so1Bj0BkDz\n -----END CERTIFICATE-----",
    name="demo"
)
response = client.create_certificate(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在项目id为project_id的项目中创建证书，证书名称为 "demo"，证书内容为 "-----BEGIN CERTIFICATE----- …… “，证书密钥为"-----BEGIN PRIVATE KEY-----”。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateCertificateRequest{}
    request.Body = &model.CreateCertificateRequestBody{
        Key: "-----BEGIN PRIVATE KEY-----
\nMIIIEvwiBADANBgkqhkiG9w0BAQEFAASCBBkwggSlAgEAAoIBAQDcolFK62//r0RH FyweYBj97S4NsJ8Qj0RG
+Y02OgwhQmRiNNjubJwP8Nqqyd86zr+fsSQxKBaBCosn 1PcN2Pj2vPJ6NEK4i6VdOWR/
kFYMIocimhSfW4wt6VakniOKIYGrCxxvQe1X2Oy BxT
+ocTLRgEiB8zbvJyPNsefLEUuPYRpQ5kXLgJH2/3NwZFOgBHVv/b07l4fR+ sWJMniA2yljSBQ0DEAOSusXoFQ/
WRbBRH7DrQmxGiXsq4VELEr9Nnc/Kywwq+9pYi 8L+mKeRL+lcMMbXC/
3k6OfMBtVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE+tmJ ZTPH6lYtAgMBAEAggEBAL+xZxm/QoqXT
+2stoqV2GEYaMFASpRqxlocjZMmEE/9 jZa+cBWjHhVPsjRqYFBDcHEebu0JwlrjciAvgnlVnO5XgXm1A9Q
+WbScokmX1 xCvpHgc+MDVn+uWdCd4KW5kEk4EnSsFN5iNSf+1VxNURN+gwSSp/0E+muwA5IISO
G6*****lcGO1n 4urPBHuNly04GgGw
+vkaqjQovZrNLVOMMaFWBxsDWBehgSSBQTj+f3NCxneGytt8 3SCTZQl5nkb+r/
M455EwKTSXuEsNHolwx7L6GEPbQECgYEA8lxgK2fykloIcOH
```

```

TFJaRAvyjyKa2+Aza4qT9SGY9Y30VPCIPjBB1vUu5M9KrFufzlv06nGEcHmpEwOe
8vbRu7nLAQTGYFi8VK63q8w6FlFdAyCG6Sx+BWCFWxjzXsZLAJTFklwi8HsOSlqh
6QN*****fvx mTbLG52Z21OyocNq3Tf/
b1Zwolic1ik6cyBzY6z1blrbSzArCqm0sb2iD+kjL81O0 /qqdXjBxZUKiVAMNnp7xjGZHHFKWUxT2+UX/
tlyx4tT4dZrFlkdDXkcMmqfsRxd 1NEVaAaT8SECgYAOu7Bptplun43YtpfUfr3pSiN6oZeKoxSbw9i4MNC
+4fSDRPC+ 80lmcmZRL7taF+Y7p0jxAOTulkdJC8NbAiv5J9WzrwQ+5MF2BPB/2bYnRa6tNofH
kZD*****QC9 ihjZTj/bTHtRiHZppzCvyYm/lgd
+Uwtsy0uXR1n0G1SQENgrTBD/J6AzdfJae6tE P0U8YIM5Oqxf2i/as9ay+IPRecMl4eSxz7jJWAGx6Yx/3AZ
+hAB1ZbNbnqniCLYNk d0MjywmA25ATO+ro4OZ7AdEpQbk3l9aG/WFyYBz9AQKBgQCucFPA115eslL8196V
WMr2Q0otqzl7CGSoWQk2Sa2HZtZdfofXAaaqo+zvJ6RPHtJh0jgJtx536DvV3egI
37YrdQyJbCPZXQ3SPgqWCorUnXBWq/nxS06uwu6JBxUFc57ijmMU4fWYNrvkkmWb 7keAg/
r5Uy1joMAvBN1I6lB8pg==\n -----END PRIVATE KEY-----",
Content: "-----BEGIN CERTIFICATE-----
\nMIIDyzCCArOgAwIBAgIJAN5U0Z4Bh5ccMA0GCSqGSIb3DQEBCwUAMHwxCzAJBgNV
BAYTAlpIMRIwEAYDVQQLDAIHVUFOR0RPTkcxETAPBgNVBAMCERPTkdHVUFOMQ0w
CwYDVQQKDARERUUtFMQswCQYDVQQQLDAJESzELMAKGA1UEAwwCT0QxHTAbBgkqhkiG
9w0BCQEWDk8lZC5odWF3Z3WkuY29tMB4XDTEwMTExNTA4MTk0MVoXDTIyMTE5NTA4
MTk0MVoVowfDElMAKGA1UEBhMCWkgxYjAQBgNVBAMCUCUdVQU5HRE9ORzERMA8GA1UE
Bjw*****VQQD
DAJPRDEdMBsGCSqGSIb3DQEJARYOTwhkLmh1YXdlas5j20wggEiMA0GCSqGSIb3
DQEBAQUAA4IBDwAwggEKAoIBAQCdcoLfk62//r0RHFyweYBj97S4NsJ8Qj0RG+Y02
OgwhQmRiNnjubJwP8Nqqy86zr+fsSQxKBaBCosn1PcN2Pj2vPJD6NEk4l6VdOWr /
kFYMI0cimhSfW4wt6VakniOKIYGrCxxvQe1X2OyBxT+ocTLRgEiB8ZbvJyPNseg
feLE*****SusXo FQ/WRbBRH7DrQmxGiXsq4VELEr9Nnc/Kywq+9pYi8L
+mKeRL+lCMMbXC/3k6OfMB tVTiwcmS1Mkr3iG03i8u6H7RSvRwyBz9G9sE
+tmJZTPH6lYtAgMBAAGjUDBOMB0G A1UdDgQWBBQprUUFxw
+gIkpzXdrYlsWjSahWjAfBgNVHSMEGDAWgBQprUUFxw+g IkpzXdrYlsWjSahWjAMBgNVHRMEBTADAQH/
MA0GCSqGSIb3DQEBCwUAA4IBAQA2 603K*****iOhU o/
kVwkiUlCw4t7RwP0hVms0OZw59MuqKd3oCSWkYO4vEHs3t40JDWnGDnmQ4sol
RkOWJwL4w8tnPe3qY9JSupjlsu6Y1hlvKtEfN2vEKFnsuMhidkUpUAJWodHhWBQH
wglDo4/6yTnWZNGK8JDal86Dm5lchXea1EoYBjsHxiJb7HeWQlkre+Mcyi1RHOin 4miXTr0oT4/jWlGklSz6/
ZhGRq+7W7tIl7cvzCe+4XsvZlenAcYoNd/WLfo91PD4 yAsRxoJw1so1Bj0BkDzn -----END CERTIFICATE-----",
Name: "demo",
}
response, err := client.CreateCertificate(request)
if err == nil {
fmt.Printf("%v\n", response)
} else {
fmt.Println(err)
}
}

```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6.3 查询证书

功能介绍

查询证书

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/certificate/{certificate_id}

表 4-1062 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
certificate_id	是	String	https证书id，您可以通过调用查询证书列表（ListCertificates）接口获取证书id

表 4-1063 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1064 请求 Header 参数

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

响应参数

状态码： 200

表 4-1065 响应 Body 参数

参数	参数类型	描述
id	String	证书ID
name	String	证书名
content	String	证书文件， PEM编码
key	String	证书私钥， PEM编码
expire_time	Long	证书过期时间戳
exp_status	Integer	证书过期状态， 0-未过期， 1-已过期， 2-即将过期
timestamp	Long	证书上传时间戳
bind_host	Array of BindHost objects	证书关联的域名信息

表 4-1066 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式： cloud（云模式） / premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400

表 4-1067 响应 Body 参数

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

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1068 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1069 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1070 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1071 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1072 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 证书id为certificate_id的证书。

```
GET https://{Endpoint}/v1/{project_id}/waf/certificate/{certificate_id}?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{
  "id": "6e2be127b79f4a418414952ad5d8c59f",
  "name": "certificatename94319",
  "content": "-----BEGIN CERTIFICATE-----\nMIIB
+TCCAaOgAwIbAgIUJP9I8OupQ77w0bGL2yWOQXreM4kwDQYJKoZIhvcNAQELBQAwUjELMAkGA1UEBhMC
QVUxEzARBgNVBAGMClNvbWUtU3RhdGUxZDZhbnRvbnQxZDZhbnRvbnQxZDZhbnRvbnQxZDZhbnRvbnQx
aW*****MRMwEQYDVQIDApTb21lLVN0YXRIMQ8wDQYDVQQKDAZidWF3Z3ZlZDZlZDZlZDZlZDZlZDZl
DAaBgNVBAMME3dhZi5odWF3Z3ZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZlZDZl
TKrKcDUw9xjFqxM7BaQFM3SLsQlmd5hkzygyL1ra
+cWajPJlTCxz9Ph6qldna2+Orl*****wYDV
R0jBBgwFoAUE7ZQNcgI3lmryx1s5gy9mnc1rsYwDwYDVR0TAQH/BAUwAwEB/
zANBgkqhkiG9w0BAQsFAANBAM5wGi88jYWLgOnGbae5hH3I9lMBKxGqv17Cbm1tjWuUogVINz86lqvCpuhZL
D/vzJAqPluDwgM8uvzjgRfZs8=\n-----END CERTIFICATE-----",
  "key": "-----BEGIN RSA PRIVATE KEY-----
\nMIIBOQIBAAJBANFBGz2M274DiUyqna1MPcYxasTOWwKbTNOi7EJZg+YZM8oMi9a2vnFmozyZUwsc/
T4eqpXZ2tvjqyLk3bwnKY8CAwEAQAIBAI7LMPaH/HQk/b/bVmY0qsr
+me9nb9BqFLuqzwKbx0hSmWPOWfSd3rOFISopyHqgYtAsPfvPumEdGbdnCyU8zAECIQD71768K1ejb
```

```
+ei2lqZqHaczqdUNQxMh54yot9F2yVWjwIhANS1Y1Jv89WEU/ZvMS9a4638Msv2c4GGp08RtXNYn0BAiA0H4b
+cwoEbZjHf+HYg6Fo+uxu5TvSaw8287a6Qo0LyQlfVZSIYYWpIT6oiX5rdLzBiap4N0gJWdsa2ihmV59LAQlgK8N
+j1daq63b0bJ9k4HruhQtpgxI6U9nFBemH4zTRYM=\n-----END RSA PRIVATE KEY-----",
  "timestamp" : 1650595334578,
  "expire_time" : 1596865564000,
  "bind_host" : [ {
    "id" : "978b411657624c2db069cd5484195d1c",
    "hostname" : "www.demo.com",
    "waf_type" : "cloud"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowCertificateSolution {

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

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

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowCertificateRequest request = new ShowCertificateRequest();
        request.withCertificateId("{certificate_id}");
        try {
            ShowCertificateResponse response = client.showCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = WafClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(WafRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ShowCertificateRequest()  
        request.certificate_id = "{certificate_id}"  
        response = client.show_certificate(request)  
        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"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()
```

```
client := waf.NewWafClient(  
    waf.WafClientBuilder().  
        WithRegion(region.ValueOf("<YOUR REGION>")).  
        WithCredential(auth).  
        Build())  
  
request := &model.ShowCertificateRequest{}  
request.CertificateId = "{certificate_id}"  
response, err := client.ShowCertificate(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6.4 修改证书

功能介绍

修改证书

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/certificate/{certificate_id}

表 4-1073 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
certificate_id	是	String	https证书id, 您可以通过调用查询证书列表 (ListCertificates) 接口获取证书id

表 4-1074 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-1075 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)。
Content-Type	是	String	内容类型

表 4-1076 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	证书名称, 证书名称只能由数字、字母、中划线、下划线和英文句点组成, 长度不能超过64位字符
content	否	String	证书文件, 仅支持PEM格式的证书和私钥文件, 且文件中的换行符应以\n替换

参数	是否必选	参数类型	描述
key	否	String	证书私钥，仅支持PEM格式的证书和私钥文件，且文件中的换行符应以\n替换

响应参数

状态码： 200

表 4-1077 响应 Body 参数

参数	参数类型	描述
id	String	证书ID
name	String	证书名
expire_time	Long	证书过期时间戳
timestamp	Long	时间戳

状态码： 400

表 4-1078 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1079 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-1080** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1081 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1082** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1083 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，证书id为certificate_id的证书的证书名称为 "demo"。

```
PUT https://{Endpoint}/v1/{project_id}/waf/certificate/{certificate_id}?enterprise_project_id=0
{
  "name" : "demo"
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "360f992501a64de0a65c50a64d1ca7b3",
  "name" : "demo",
  "timestamp" : 1650593797892,
  "expire_time" : 1596865564000
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，证书id为certificate_id的证书的证书名称为 "demo"。

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

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

public class UpdateCertificateSolution {

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



```
String projectId = "{project_id}";

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

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateCertificateRequest request = new UpdateCertificateRequest();
request.withCertificateId("{certificate_id}");
UpdateCertificateRequestBody body = new UpdateCertificateRequestBody();
body.withName("demo");
request.withBody(body);
try {
    UpdateCertificateResponse response = client.updateCertificate(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新项目id为project_id，证书id为certificate_id的证书的证书名称为 "demo"。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

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

    try:
        request = UpdateCertificateRequest()
        request.certificate_id = "{certificate_id}"
        request.body = UpdateCertificateRequestBody(
            name="demo"
        )
        response = client.update_certificate(request)
```

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

Go

更新项目id为project_id，证书id为certificate_id的证书的证书名称为 "demo"。

```
package main

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

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

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

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateCertificateRequest{}
    request.CertificateId = "{certificate_id}"
    request.Body = &model.UpdateCertificateRequestBody{
        Name: "demo",
    }
    response, err := client.UpdateCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

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

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6.5 删除证书

功能介绍

删除证书

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/waf/certificate/{certificate_id}

表 4-1084 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
certificate_id	是	String	https证书id，您可以通过调用查询证书列表（ListCertificates）接口获取证书id

表 4-1085 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1086 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1087 响应 Body 参数

参数	参数类型	描述
id	String	证书ID
name	String	证书名
content	String	证书文件，PEM编码
key	String	证书私钥，PEM编码
expire_time	Long	证书过期时间戳
exp_status	Integer	证书过期状态，0-未过期，1-已过期，2-即将过期
timestamp	Long	证书上传时间戳
bind_host	Array of BindHost objects	证书关联的域名信息

表 4-1088 BindHost

参数	参数类型	描述
id	String	域名ID
hostname	String	域名
waf_type	String	域名对应模式：cloud（云模式）/ premium（独享模式）
mode	String	仅独享模式涉及特殊域名模式

状态码： 400**表 4-1089** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1090 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-1091** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1092 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1093 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1094 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 证书id为certificate_id的证书。

```
DELETE https://{Endpoint}/v1/{project_id}/waf/certificate/{certificate_id}?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{  
  "id": "e1d87ba2d88d4ee4a3b0c829e935e5e0",  
  "name": "certificatename29556",  
  "timestamp": 1650594410630,  
}
```

```
"expire_time": 1596865564000  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class DeleteCertificateSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeleteCertificateRequest request = new DeleteCertificateRequest();  
        request.withCertificateId("{certificate_id}");  
        try {  
            DeleteCertificateResponse response = client.deleteCertificate(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

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

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteCertificateRequest()
        request.certificate_id = "{certificate_id}"
        response = client.delete_certificate(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteCertificateRequest{}
    request.CertificateId = "{certificate_id}"
    response, err := client.DeleteCertificate(request)
```



```
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.6.6 绑定证书到域名

功能介绍

绑定证书到域名

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/certificate/{certificate_id}/apply-to-hosts

表 4-1095 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
certificate_id	是	String	https证书id，您可以通过调用查询证书列表（ListCertificates）接口获取证书id

表 4-1096 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1097 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）。
Content-Type	是	String	内容类型

表 4-1098 请求 Body 参数

参数	是否必选	参数类型	描述
cloud_host_ids	否	Array of strings	云模式HTTPS域名id，通过查询云模式防护域名列表（ListHost）接口获取
premium_host_ids	否	Array of strings	独享模式HTTPS域名id，通过独享模式域名列表（ListPremiumHost）接口获取

响应参数

状态码：200

表 4-1099 响应 Body 参数

参数	参数类型	描述
id	String	证书id
name	String	证书名
timestamp	Long	时间戳
expire_time	Long	过期时间

参数	参数类型	描述
bind_host	Array of CertificateBundlingHostBody objects	绑定域名列表

表 4-1100 CertificateBundlingHostBody

参数	参数类型	描述
id	String	域名id
hostname	String	域名
waf_type	String	waf模式（分为云模式：cloud,独享模式：premium）

状态码： 400

表 4-1101 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1102 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1103 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1104 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1105 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1106 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

绑定项目id为project_id和证书id为certificate_id的证书到两个域名，分别为域名id为“85e554189d494c0f97789e93531c9f90”的云模式域名和域名id为“4e9e97c425fc463c8f374b90124e8392”的独享模式域名。

```
POST https://{Endpoint}/v1/{project_id}/waf/certificate/{certificate_id}/apply-to-hosts?
enterprise_project_id=0
```

```
{
  "cloud_host_ids" : [ "85e554189d494c0f97789e93531c9f90" ],
  "premium_host_ids" : [ "4e9e97c425fc463c8f374b90124e8392" ]
}
```

响应示例

状态码： 200

OK

```
{
  "id" : "3ac1402300374a63a05be68c641e92c8",
  "name" : "www.abc.com",
  "timestamp" : 1636343349139,
  "expire_time" : 1650794100000,
  "bind_host" : [ {
    "id" : "e350cf556da34adab1f017523d1c05ed",
    "hostname" : "www.demo.com",
    "waf_type" : "cloud"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

绑定项目id为project_id和证书id为certificate_id的证书到两个域名，分别为域名id为“85e554189d494c0f97789e93531c9f90”的云模式域名和域名id为“4e9e97c425fc463c8f374b90124e8392”的独享模式域名。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;
```

```
import java.util.List;
import java.util.ArrayList;

public class ApplyCertificateToHostSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ApplyCertificateToHostRequest request = new ApplyCertificateToHostRequest();
        request.withCertificateId("{certificate_id}");
        ApplyCertificateToHostRequestBody body = new ApplyCertificateToHostRequestBody();
        List<String> listbodyPremiumHostIds = new ArrayList<>();
        listbodyPremiumHostIds.add("4e9e97c425fc463c8f374b90124e8392");
        List<String> listbodyCloudHostIds = new ArrayList<>();
        listbodyCloudHostIds.add("85e554189d494c0f97789e93531c9f90");
        body.withPremiumHostIds(listbodyPremiumHostIds);
        body.withCloudHostIds(listbodyCloudHostIds);
        request.withBody(body);
        try {
            ApplyCertificateToHostResponse response = client.applyCertificateToHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

绑定项目id为project_id和证书id为certificate_id的证书到两个域名，分别为域名id为“85e554189d494c0f97789e93531c9f90”的云模式域名和域名id为“4e9e97c425fc463c8f374b90124e8392”的独享模式域名。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ApplyCertificateToHostRequest()
    request.certificate_id = "{certificate_id}"
    listPremiumHostIdsbody = [
        "4e9e97c425fc463c8f374b90124e8392"
    ]
    listCloudHostIdsbody = [
        "85e554189d494c0f97789e93531c9f90"
    ]
    request.body = ApplyCertificateToHostRequestBody(
        premium_host_ids=listPremiumHostIdsbody,
        cloud_host_ids=listCloudHostIdsbody
    )
    response = client.apply_certificate_to_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

绑定项目id为project_id和证书id为certificate_id的证书到两个域名，分别为域名id为“85e554189d494c0f97789e93531c9f90”的云模式域名和域名id为“4e9e97c425fc463c8f374b90124e8392”的独享模式域名。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
}
```

```
client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ApplyCertificateToHostRequest{}
request.CertificateId = "{certificate_id}"
var listPremiumHostIdsbody = []string{
    "4e9e97c425fc463c8f374b90124e8392",
}
var listCloudHostIdsbody = []string{
    "85e554189d494c0f97789e93531c9f90",
}
request.Body = &model.ApplyCertificateToHostRequestBody{
    PremiumHostIds: &listPremiumHostIdsbody,
    CloudHostIds: &listCloudHostIdsbody,
}
response, err := client.ApplyCertificateToHost(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.7 防护事件管理

4.7.1 查询指定事件 id 的防护事件详情

功能介绍

查询指定事件id的防护事件详情

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/event/{eventid}

表 4-1107 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
eventid	是	String	防护事件id,通过调用查询攻击事件列表(ListEvent)接口获取防护事件id

表 4-1108 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-1109 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)。
Content-Type	是	String	内容类型
X-Language	否	String	语言, 默认值为en-us。zh-cn (中文)/en-us (英文)

响应参数

状态码: 200

表 4-1110 响应 Body 参数

参数	参数类型	描述
total	Integer	攻击事件数量
items	Array of ShowEventItems objects	攻击事件详情

表 4-1111 ShowEventItems

参数	参数类型	描述
time	Long	攻击发生时的时间戳（毫秒）
policyid	String	策略id
sip	String	源ip，Web访问者的IP地址（攻击者IP地址）
host	String	域名
url	String	攻击的url链接
attack	String	攻击类型
rule	String	命中的规则id
action	String	防护动作
cookie	String	攻击请求的cookie
headers	Object	攻击请求的headers
host_id	String	被攻击的域名id
id	String	防护事件id
payload	String	恶意负载
payload_location	String	恶意负载位置
region	String	源ip地理位置
process_time	Integer	处理时长
request_line	String	攻击请求的请求行
response_size	Integer	返回大小（字节）
response_time	Long	响应时间（毫秒）
status	String	响应码
request_body	String	请求体

状态码： 400**表 4-1112 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1113 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-1114 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1115 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1116 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1117 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 防护事件id为event_id的的防护事件详情

GET https://{Endpoint}/v1/{project_id}/waf/event/{event_id}?enterprise_project_id=0

响应示例

状态码： 200

ok

```
{
  "total": 1,
  "items": [{
    "id": "09-0000-0000-0000-12120220421093806-a60a6166",
```

```
"time" : 1650505086000,
"policyid" : "173ed802272a4b0798049d7edffeff03",
"host" : "x.x.x.x:xxxxxx-xxx-xxx-xxx-xxxxxxx",
"url" : "/mobile/DBconfigReader.jsp",
"attack" : "vuln",
"rule" : "091004",
"payload" : " /mobile/dbconfigreader.jsp",
"payload_location" : "uri",
"sip" : "x.x.x.x",
"action" : "block",
"request_line" : "GET /mobile/DBconfigReader.jsp",
"headers" : {
  "ls-id" : "c0d957e6-26a8-4f2e-8216-7fc9332a250f",
  "host" : "x.x.x.x:81",
  "lb-id" : "68d3c435-2607-45e0-a5e2-38980544dd45",
  "accept-encoding" : "gzip",
  "user-agent" : "Mozilla/5.0 (Windows NT 10.0; rv:78.0) Gecko/20100101 CSIRTx/2022"
},
"cookie" : "HWWAFSESID=2a0bf76a111c93926d; HWWAFSESTIME=1650505086260",
"status" : "418",
"region" : "Reserved IP",
"host_id" : "e093a352fd3a4ddd994c585e2e1dda59",
"response_time" : 0,
"response_size" : 3318,
"process_time" : 0,
"request_body" : "{}"
}]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowEventSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
    }
}
```

```
ShowEventRequest request = new ShowEventRequest();
request.withEventid("{eventid}");
try {
    ShowEventResponse response = client.showEvent(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowEventRequest()
        request.eventid = "{eventid}"
        response = client.show_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

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowEventRequest{}
    request.EventId = "{eventid}"
    response, err := client.ShowEvent(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.7.2 查询攻击事件列表

功能介绍

查询攻击事件列表，该API暂时不支持查询全部防护事件，pagesize参数不可设为-1，由于性能原因，数据量越大消耗的内存越大，后端最多限制查询10000条数据，例如：

自定义时间段内的数据超过了10000条，就无法查出page为101，pagesize为100之后的数据，需要调整时间区间，再进行查询

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/event

表 4-1118 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1119 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
recent	否	String	查询日志的时间范围（不能和from、to同时使用，同时使用以recent为准），且recent参数与from、to必须使用其中一个。当同时使用recent参数与from、to时，以recent参数为准
from	否	Long	起始时间(13位时间戳)，需要和to同时使用，不能和recent参数同时使用
to	否	Long	结束时间(13位时间戳)，需要和from同时使用，不能和recent参数同时使用

参数	是否必选	参数类型	描述
attacks	否	Array of strings	攻击类型: <ul style="list-style-type: none"> • vuln: 其它攻击类型 • sqli: sql注入攻击 • lfi: 本地文件包含 • cmdi: 命令注入攻击 • xss: XSS攻击 • robot: 恶意爬虫 • rfi: 远程文件包含 • custom_custom: 精准防护 • cc: cc攻击 • webshell: 网站木马 • custom_whiteblackip: 黑白名单拦截 • custom_geoop: 地理访问控制拦截 • antitamper: 防篡改 • anticrawler: 反爬虫 • leakage: 网站信息防泄漏 • illegal: 非法请求 • antiscan_high_freq_scan: 高频扫描封禁 • antiscan_dir_traversal: 目录遍历防护
hosts	否	Array of strings	域名id, 从获取防护网站列表 (ListHost) 接口获取域名id
page	否	Integer	分页查询时, 返回第几页数据。默认值为1, 表示返回第1页数据。
pagesize	否	Integer	分页查询时, 每页包含多少条结果。默认值为10, 表示每页包含10条结果。

请求参数

表 4-1120 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)。
Content-Type	是	String	内容类型
X-Language	否	String	语言，默认值为en-us。zh-cn (中文)/en-us (英文)

响应参数

状态码： 200

表 4-1121 响应 Body 参数

参数	参数类型	描述
total	Integer	攻击事件数量
items	Array of ListEventItems objects	攻击事件详情

表 4-1122 ListEventItems

参数	参数类型	描述
id	String	事件id
time	Long	攻击发生时的时间戳(毫秒)
policyid	String	策略id
sip	String	源ip，Web访问者的IP地址（攻击者IP地址）
host	String	域名
url	String	攻击的url链接

参数	参数类型	描述
attack	String	攻击类型: <ul style="list-style-type: none"> vuln: 其它攻击类型 sqli: sql注入攻击 lfi: 本地文件包含 cmdi: 命令注入攻击 xss: XSS攻击 robot: 恶意爬虫 rfi: 远程文件包含 custom_custom: 精准防护 webshell: 网站木马 custom_blacklist: 黑白名单拦截 custom_geoip: 地理访问控制拦截 antitamper: 防篡改 anticrawler: 反爬虫 leakage: 网站信息防泄漏 illegal: 非法请求 antiscan_high_freq_scan: 高频扫描封禁 antiscan_dir_traversal: 目录遍历防护
rule	String	命中的规则id
payload	String	命中的载荷
payload_location	String	命中的载荷位置
action	String	防护动作
request_line	String	请求方法和路径
headers	Object	http请求header
cookie	String	请求cookie
status	String	响应码状态
process_time	Integer	处理时长
region	String	地理位置
host_id	String	域名id
response_time	Long	响应时长
response_size	Integer	响应体大小
response_body	String	响应体

参数	参数类型	描述
request_body	String	请求体

状态码： 400

表 4-1123 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1124 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1125 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1126 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1127 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1128 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询今天项目id为project_id的防护事件列表

```
GET https://{Endpoint}/v1/{project_id}/waf/event?
enterprise_project_id=0&page=1&pagesize=10&recent=today
```

响应示例

状态码： 200

ok

```
{
  "total" : 1,
  "items" : [ {
    "id" : "04-0000-0000-0000-21120220421152601-2f7a5ceb",
    "time" : 1650525961000,
    "policyid" : "25f1d179896e4e3d87ceac0598f48d00",
    "host" : "x.x.x.x:xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "url" : "/osclass/oc-admin/index.php",
    "attack" : "lfi",
    "rule" : "040002",
    "payload" : " file=../../../../../../../../etc/passwd",
    "payload_location" : "params",
    "sip" : "x.x.x.x",
    "action" : "block",
    "request_line" : "GET /osclass/oc-admin/index.php?
page=appearance&action=render&file=../../../../../../../../etc/passwd",
    "headers" : {
      "accept-language" : "en",
      "ls-id" : "xxxxx-xxxxx-xxxx-xxxx-9c302cb7c54a",
      "host" : "x.x.x.x",
      "lb-id" : "2f5f15ce-08f4-4df0-9899-ec0cc1fcdc52",
      "accept-encoding" : "gzip",
      "accept" : "*/*",
      "user-agent" : "Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
35.0.2309.372 Safari/537.36"
    },
    "cookie" : "HWWAFSESID=2a1d773f9199d40a53; HWWAFSESTIME=1650525961805",
    "status" : "418",
    "host_id" : "6fbe595e7b874dbbb1505da3e8579b54",
    "response_time" : 0,
    "response_size" : 3318,
    "response_body" : "",
    "process_time" : 2,
    "request_body" : "{}"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListEventSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListEventRequest request = new ListEventRequest();
try {
    ListEventResponse response = client.listEvent(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListEventRequest()
        response = client.list_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

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListEventRequest{}
    response, err := client.ListEvent(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8 安全总览

4.8.1 查询安全统计 qps 次数

功能介绍

查询安全统计qps次数。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/qps/timeline

表 4-1129 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1130 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	通过企业项目管理服务的查询企业项目列表接口 ListEnterpriseProject查询通过企业项目管理服务的查询企业项目列表接口 ListEnterpriseProject查询企业项目ID
from	是	Long	起始时间（13位毫秒时间戳），需要和to同时使用
to	是	Long	结束时间（13位毫秒时间戳），需要和from同时使用
hosts	否	String	域名id，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id
instances	否	String	要查询引擎实例id（仅独享或者ELB实例化模式涉及）

参数	是否必选	参数类型	描述
group_by	否	String	展示维度，按天展示时传 "DAY"；默认不传，按照分钟展示

请求参数

表 4-1131 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1132 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of StatisticsTimelineItem objects	安全总览的Qps时间线统计数据

表 4-1133 StatisticsTimelineItem

参数	参数类型	描述
key	String	键值，包括请求总量（ACCESS）、Bot攻击防护（CRAWLER）、攻击总量（ATTACK）、Web基础防护（WEB_ATTACK）、精准防护（PRECISE）、CC攻击防护（CC）
timeline	Array of TimeLineItem objects	对应键值的时间线统计数据

表 4-1134 TimeLineItem

参数	参数类型	描述
time	Long	时间点
num	Integer	time对应时间点与前一时间点间隔内的统计数值。

状态码： 400

表 4-1135 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1136 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1137 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1138 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1139 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1140 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，时间从2022-04-21 00:00:00到2022-04-21 14:35:36的安全统计qps次数。

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/qps/timeline?
enterprise_project_id=0&from=1650470400196&to=1650522936196
```

响应示例

状态码： 200

ok

```
[ {
  "key": "ACCESS",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "PRECISE",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "CRAWLER",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "CC",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "ATTACK",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "WEB_ATTACK",
  "timeline": [ {
    "time": 1650470400000,
    "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.waf.v1.region.WafRegion;
```

```
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListQpsTimelineSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListQpsTimelineRequest request = new ListQpsTimelineRequest();
        try {
            ListQpsTimelineResponse response = client.listQpsTimeline(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
try:
    request = ListQpsTimelineRequest()
    response = client.list_qps_timeline(request)
    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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListQpsTimelineRequest{}
    response, err := client.ListQpsTimeline(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok

状态码	描述
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8.2 查询安全总览请求与攻击数量

功能介绍

查询安全总览请求与攻击数量。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/statistics

表 4-1141 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1142 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
from	是	Long	起始时间(13位时间戳)，需要和to同时使用
to	是	Long	结束时间(13位时间戳)，需要和from同时使用

参数	是否必选	参数类型	描述
hosts	否	String	域名id，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id
instances	否	String	要查询引擎实例id

请求参数

表 4-1143 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1144 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of CountItem objects	安全总览请求与攻击数据

表 4-1145 CountItem

参数	参数类型	描述
key	String	类型，包括请求总量（ACCESS）、Bot攻击防护（CRAWLER）、攻击总量（ATTACK）、Web基础防护（WEB_ATTACK）、精准防护（PRECISE）以及CC攻击防护（CC）
num	Long	数量

状态码： 400**表 4-1146** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1147 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-1148** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1149 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 403

表 4-1150 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1151 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1152 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1153 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 时间从2022-04-21 00:00:00到2022-04-21 14:35:36的安全总览请求与攻击数量

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/statistics?enterprise_project_id=0&from=1650470400196&to=1650522936196
```

响应示例

状态码: 200

成功

```
[{
  "key": "ACCESS",
  "num": 1190
}, {
  "key": "PRECISE",
  "num": 0
}, {
  "key": "CRAWLER",
  "num": 10
}, {
  "key": "WEB_ATTACK",
  "num": 22
}, {
  "key": "CC",
  "num": 0
}, {
  "key": "ATTACK",
  "num": 32
}]
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListStatisticsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListStatisticsRequest request = new ListStatisticsRequest();
        try {
            ListStatisticsResponse response = client.listStatistics(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListStatisticsRequest()
    response = client.list_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListStatisticsRequest{}
    response, err := client.ListStatistics(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功
400	请求失败
401	token权限不足
403	资源配额不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8.3 查询安全统计带宽数据

功能介绍

查询安全统计带宽数据，统计的带宽数据为平均值，单位为bit/s。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/bandwidth/timeline

表 4-1154 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1155 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
from	是	Long	查询的带宽统计数据的起始时间（13位毫秒时间戳），需要和to同时使用

参数	是否必选	参数类型	描述
to	是	Long	查询的带宽统计数据的结束时间（13位毫秒时间戳），需要和from同时使用
hosts	否	String	域名id，用于查询指定的防护域名在from到to这段时间内的带宽数据。通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id
instances	否	String	引擎实例id，用于查询指定的独享引擎实例所防护的域名在from到to这段时间内的带宽数据。
group_by	否	String	展示维度，按天展示时传"DAY"；默认不传，按照分钟展示。
display_option	否	Integer	发送/接受字节数，查看峰值请输入1，查看平均值请输入0

请求参数

表 4-1156 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1157 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of BandwidthStatisticsTimelineItem objects	带宽时间线统计数据，包括带宽（BANDWIDTH）、入带宽（IN_BANDWIDTH）以及出带宽（OUT_BANDWIDTH）统计数据。

表 4-1158 BandwidthStatisticsTimelineItem

参数	参数类型	描述
key	String	键值，其值包括带宽（BANDWIDTH）、入带宽（IN_BANDWIDTH）以及出带宽（OUT_BANDWIDTH）
timeline	Array of TimelineItem objects	对应键值的时间线统计数据，包含两个字段，time字段值为时间点；num字段为time对应时间点与前一时间点间隔内的统计数值

表 4-1159 TimelineItem

参数	参数类型	描述
time	Long	时间点
num	Integer	time对应时间点与前一时间点间隔内的统计数值。

状态码： 400

表 4-1160 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。

参数	参数类型	描述
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1161 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1162 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1163 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1164 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1165 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 时间从2022-04-21 00:00:00到2022-04-21 14:35:36的安全统计带宽数据

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/bandwidth/timeline?enterprise_project_id=0&from=1650470400196&to=1650522936196
```

响应示例

状态码: 200

ok

```
[{
  "key": "IN_BANDWIDTH",
  "timeline": [{
    "time": 1650470400000,
    "num": 0
  }]
}, {
  "key": "OUT_BANDWIDTH",
  "timeline": [{
    "time": 1650470400000,
    "num": 0
  }]
}, {
  "key": "BANDWIDTH",
  "timeline": [{
    "time": 1650470400000,
    "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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class ListBandwidthTimelineSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListBandwidthTimelineRequest request = new ListBandwidthTimelineRequest();  
        try {  
            ListBandwidthTimelineResponse response = client.listBandwidthTimeline(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListBandwidthTimelineRequest()
        response = client.list_bandwidth_timeline(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListBandwidthTimelineRequest{}
    response, err := client.ListBandwidthTimeline(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
```

```
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8.4 查询业务异常数量

功能介绍

查询业务异常TOP统计信息。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/abnormal

表 4-1166 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1167 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
from	是	Long	起始时间（13位毫秒时间戳），需要和to同时使用
to	是	Long	结束时间（13位毫秒时间戳），需要和from同时使用
top	否	Integer	要查询的前几的结果，默认值为5，最大值为10。
code	否	Integer	要查询的异常状态码，目前支持查询的异常状态码包括404、500以及502。不传该参数默认查询404的状态码。
hosts	否	String	域名id，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id。默认不传，查询该项目下所有防护域名的top业务异常统计信息。
instances	否	String	要查询引擎实例id

请求参数

表 4-1168 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1169 响应 Body 参数

参数	参数类型	描述
total	Integer	异常请求数量
items	Array of UrlCountItem objects	异常请求信息数组

表 4-1170 UrlCountItem

参数	参数类型	描述
key	String	攻击类型
num	Integer	数量
host	String	防护域名

状态码： 400

表 4-1171 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1172 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1173 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1174 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1175 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1176 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，时间从2022-04-21 00:00:00到2022-04-21 14:35:36，异常状态码为“404”的业务异常TOP统计信息

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/abnormal?
enterprise_project_id=0&from=1650470400089&to=1650523520089&top=10&code=404
```

响应示例

状态码： 200

ok

```
{
  "total": 2,
  "items": [ {
    "key": "/",
    "num": 6,
    "host": "hkh4.test.418lab.cn"
  }, {
    "key": "/",
    "num": 6,
    "host": "ces_after.test.418lab.cn"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListTopAbnormalSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListTopAbnormalRequest request = new ListTopAbnormalRequest();
try {
    ListTopAbnormalResponse response = client.listTopAbnormal(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTopAbnormalRequest()
        response = client.list_top_abnormal(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListTopAbnormalRequest{}
    response, err := client.ListTopAbnormal(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8.5 查询安全总览分类统计 top 信息

功能介绍

查询安全总览分类统计TOP信息，包含受攻击域名、攻击源ip、受攻击URL、攻击来源区域、攻击事件分布。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/classification

表 4-1177 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1178 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
from	是	Long	起始时间（13位毫秒时间戳），需要和to同时使用
to	是	Long	结束时间（13位毫秒时间戳），需要和from同时使用
top	否	Integer	要查询的前几的结果，最大值为10，默认值为5。
hosts	否	String	域名id，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id。默认不传，查询该项目下所有防护域名的top业务异常统计信息。
instances	否	String	要查询引擎实例id

请求参数

表 4-1179 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)
Content-Type	是	String	内容类型
X-Language	否	String	语言，默认值为en-us。zh-cn (中文)/en-us (英文)

响应参数

状态码： 200

表 4-1180 响应 Body 参数

参数	参数类型	描述
domain	DomainClassificationItem object	受攻击域名
attack_type	AttackTypeClassificationItem object	攻击事件分布
ip	IpClassificationItem object	攻击源ip
url	UrlClassificationItem object	受攻击url
geo	GeoClassificationItem object	攻击来源区域

表 4-1181 DomainClassificationItem

参数	参数类型	描述
total	Integer	DomainItem的总数量
items	Array of DomainItem objects	DomainItem详细信息

表 4-1182 DomainItem

参数	参数类型	描述
key	String	域名
num	Integer	数量
web_tag	String	网站名称，对应WAF控制台域名详情中的网站名称

表 4-1183 AttackTypeClassificationItem

参数	参数类型	描述
total	Integer	AttackTypeItem的总数量
items	Array of AttackTypeItem objects	AttackTypeItem详细信息

表 4-1184 AttackTypeItem

参数	参数类型	描述
key	String	攻击事件类型
num	Integer	数量

表 4-1185 IpClassificationItem

参数	参数类型	描述
total	Integer	IpItem的总数量
items	Array of IpItem objects	IpItem详细信息

表 4-1186 IpItem

参数	参数类型	描述
key	String	ip地址
num	Integer	数量

表 4-1187 UrlClassificationItem

参数	参数类型	描述
total	Integer	UrlItem的总数量
items	Array of UrlItem objects	UrlItem详细信息

表 4-1188 UrlItem

参数	参数类型	描述
key	String	url路径
num	Integer	数量
host	String	域名

表 4-1189 GeoClassificationItem

参数	参数类型	描述
total	Integer	GeoItem的总数量
items	Array of GeoItem objects	GeoItem详细信息

表 4-1190 GeoItem

参数	参数类型	描述
key	String	攻击来源区域
num	Integer	数量

状态码： 400

表 4-1191 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1192 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1193 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1194 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1195 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1196 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 时间从2022-05-19 00:00:00到2022-06-17 11:14:41, 域名id为“1bac09440a814aa98ed08302c580a48b”, 引擎实例id为“5a532f83a2fb476ba51ca1de7b1ebc43”的top10安全总览分类统计信息

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/classification?enterprise_project_id=0&from=1652889600354&to=1655435681354&top=10&hosts=1bac09440a814aa98ed08302c580a48b&instances=5a532f83a2fb476ba51ca1de7b1ebc43
```

响应示例

状态码： 200

ok

```
{
  "attack_type": {
    "total": 1,
    "items": [ {
      "key": "custom_custom",
      "num": 2
    } ]
  },
  "domain": {
    "total": 2,
    "items": [ {
      "key": "www.whitelist.com",
```

```
"num" : 2,
"web_tag" : "www.whitelist.com"
}, {
"key" : "zbx002.apayaduo.cn",
"num" : 2,
"web_tag" : ""
}]
},
"geo" : {
"total" : 1,
"items" : [ {
"key" : "上海",
"num" : 2
} ]
},
"ip" : {
"total" : 1,
"items" : [ {
"key" : "10.142.4.15",
"num" : 2
} ]
},
"url" : {
"total" : 1,
"items" : [ {
"key" : "/attack",
"num" : 2,
"host" : "www.whitelist.com"
} ]
}
}
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListOverviewsClassificationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
```

```
        .withCredential(auth)
        .withRegion(WafRegion.valueOf("<YOUR REGION>"))
        .build();
ListOverviewsClassificationRequest request = new ListOverviewsClassificationRequest();
try {
    ListOverviewsClassificationResponse response = client.listOverviewsClassification(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListOverviewsClassificationRequest()
        response = client.list_overviews_classification(request)
        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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListOverviewsClassificationRequest{}
    response, err := client.ListOverviewsClassification(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.8.6 查询安全总览中请求次数时间线统计数据

功能介绍

查询安全总览中请求次数时间线统计数据。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/overviews/request/timeline

表 4-1197 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1198 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id
from	是	Long	起始时间（13位毫秒时间戳），需要和to同时使用
to	是	Long	结束时间（13位毫秒时间戳），需要和from同时使用
hosts	否	Array of strings	域名id列表，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id。默认不传，查询该项目下所有防护域名的top业务异常统计信息。
instances	否	Array of strings	要查询引擎实例id列表
group_by	否	String	展示维度，按天展示时传"DAY"；默认不传，按照分钟展示

请求参数

表 4-1199 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)
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1200 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of StatisticsTimelineltem objects	安全统计的请求时间线数据

表 4-1201 StatisticsTimelineltem

参数	参数类型	描述
key	String	键值，包括请求总量（ACCESS）、Bot攻击防护（CRAWLER）、攻击总量（ATTACK）、Web基础防护（WEB_ATTACK）、精准防护（PRECISE）、CC攻击防护（CC）
timeline	Array of TimeLineltem objects	对应键值的时间线统计数据

表 4-1202 TimeLineltem

参数	参数类型	描述
time	Long	时间点
num	Integer	time对应时间点与前一时间点间隔内的统计数值。

状态码： 400

表 4-1203 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1204 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1205 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1206 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1207 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1208 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id, 时间从2022-04-21 00:00:00到2022-04-21 00:00:50的安全总览中请求次数时间线统计数据

```
GET https://{Endpoint}/v1/{project_id}/waf/overviews/request/timeline?enterprise_project_id=0&from=1650470400196&to=1650470450000
```

响应示例

状态码： 200

ok

```
[{
  "key": "ACCESS",
```

```
"timeline": [ {
  "time": 1650470400196,
  "num": 0
} ]
}, {
  "key": "PRECISE",
  "timeline": [ {
    "time": 1650470400196,
    "num": 0
  } ]
}, {
  "key": "CRAWLER",
  "timeline": [ {
    "time": 1650470400196,
    "num": 0
  } ]
}, {
  "key": "CC",
  "timeline": [ {
    "time": 1650470400196,
    "num": 0
  } ]
}, {
  "key": "ATTACK",
  "timeline": [ {
    "time": 1650470400000,
    "num": 0
  } ]
}, {
  "key": "WEB_ATTACK",
  "timeline": [ {
    "time": 1650470400196,
    "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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListRequestTimelineSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
```

```
.withAk(ak)
.withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListRequestTimelineRequest request = new ListRequestTimelineRequest();
try {
    ListRequestTimelineResponse response = client.listRequestTimeline(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListRequestTimelineRequest()
        response = client.list_request_timeline(request)
        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"
```

```
waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListRequestTimelineRequest{}  
    response, err := client.ListRequestTimeline(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.9 局点支持特性查询

4.9.1 局点支持特性查询

功能介绍

局点支持特性查询

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/config/console

表 4-1209 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

请求参数

表 4-1210 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1211 响应 Body 参数

参数	参数类型	描述
eps	Boolean	是否支持EPS，false：不支持；true：支持
tls	Boolean	是否支持的TLS版本（TLS v1.0/TLS v1.1/TLS v1.2），默认为TLS v1.0版本，false：不支持；true：支持

参数	参数类型	描述
ipv6	Boolean	是否支持IPV6, false: 不支持; true: 支持
alert	Boolean	是否支持告警, false: 不支持; true: 支持
custom	Boolean	是否支持精准防护, false: 不支持; true: 支持
elb_mode	Boolean	是否支持ELB模式, false: 不支持; true: 支持
event_lts	Boolean	是否支持LTS全量日志, false: 不支持; true: 支持
multi_dns	Boolean	是否支持多DNS解析, false: 不支持; true: 支持
search_ip	Boolean	是否支持搜索IP, false: 不支持; true: 支持
cc_enhance	Boolean	是否支持CC增强, false: 不支持; true: 支持
cname_switch	Boolean	是否支持cname切换, false: 不支持; true: 支持
custom_block	Boolean	是否支持自定义拦截页面, false: 不支持, true: 支持
advanced_ignore	Boolean	是否支持误报屏蔽, false: 不支持; true: 支持
js_crawler_enable	Boolean	是否支持js反爬虫, false: 不支持; true: 支持
deep_decode_enable	Boolean	是否支持web基础防护深度检测, false: 不支持; true: 支持
overview_bandwidth	Boolean	是否支持安全总览带宽统计, false: 不支持; true: 支持
proxy_use_oldcname	Boolean	是否支持使用旧cname解析, false: 不支持; true: 支持
check_all_headers_enable	Boolean	是否支持检查所有的header, false: 不支持; true: 支持
geoip_enable	Boolean	是否支持地理位置访问控制, false: 不支持; true: 支持
load_balance_enable	Boolean	是否支持域名访问负载均衡配置, false: 不支持; true: 支持
ipv6_protection_enable	Boolean	是否支持ipv6防护, false: 不支持; true: 支持

参数	参数类型	描述
policy_sharing_enable	Boolean	是否支持策略共享, false: 不支持; true: 支持
ip_group	Boolean	是否支持ip地址组, false: 不支持; true: 支持
robot_action_enable	Boolean	是否支持网站反爬虫, false: 不支持; true: 支持
http2_enable	Boolean	是否支持http2, false: 不支持; true: 支持
timeout_config_enable	Boolean	是否支持超时配置, false: 不支持; true: 支持

状态码: 400

表 4-1212 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1213 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1214 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1215 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1216 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1217 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，时间从2022-04-21 00:00:00到2022-04-21 14:45:20的局点支持特性

```
GET https://{Endpoint}/v1/{project_id}/waf/config/console?
enterprise_project_id=0&from=1650470400089&to=1650523520089
```

响应示例

状态码： 200

OK

```
{
  "eps" : true,
  "tls" : true,
  "ipv6" : true,
  "alert" : true,
  "custom" : true,
  "ip_group" : true,
  "event_lts" : true,
  "multi_dns" : true,
  "search_ip" : true,
  "cc_enhance" : true,
  "cname_switch" : true,
  "custom_block" : true,
  "geoip_enable" : true,
  "http2_enable" : true,
  "advanced_ignore" : true,
  "js_crawler_enable" : true,
  "deep_decode_enable" : true,
  "overview_bandwidth" : true,
  "proxy_use_oldcname" : false,
  "load_balance_enable" : true,
  "robot_action_enable" : true,
  "policy_sharing_enable" : true,
  "timeout_config_enable" : true,
  "ipv6_protection_enable" : true,
  "check_all_headers_enable" : true,
  "elb_mode" : true
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowConsoleConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowConsoleConfigRequest request = new ShowConsoleConfigRequest();
        try {
            ShowConsoleConfigResponse response = client.showConsoleConfig(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
```

```
.with_credentials(credentials) \  
.with_region(WafRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = ShowConsoleConfigRequest()  
    response = client.show_console_config(request)  
    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"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ShowConsoleConfigRequest{}  
    response, err := client.ShowConsoleConfig(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.10 独享实例管理

4.10.1 查询 WAF 独享引擎列表

功能介绍

查询WAF独享引擎列表。独享模式只在部分局点支持，包括：华北-北京四、华东-上海一、华南-广州、华南-深圳、中国-香港、亚太-曼谷、亚太-新加坡。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/premium-waf/instance

表 4-1218 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1219 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

参数	是否必选	参数类型	描述
page	否	Integer	分页查询参数，第page页，默认值为1
pagesize	否	Integer	分页查询参数，每页pagesize条记录，默认值为10
instancename	否	String	模糊查询，独享引擎名称

请求参数

表 4-1220 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1221 响应 Body 参数

参数	参数类型	描述
total	Integer	独享引擎实例数量
purchased	Boolean	是否曾经购买过独享引擎
items	Array of ListInstance objects	详细的独享引擎信息列表

表 4-1222 ListInstance

参数	参数类型	描述
id	String	独享引擎实例ID
instancename	String	独享引擎实例名称
region	String	独享引擎实例Region ID

参数	参数类型	描述
zone	String	可用区ID
arch	String	CPU架构
cpu_flavor	String	ECS规格
vpc_id	String	独享引擎实例所在VPC ID
subnet_id	String	独享引擎实例所在VPC的子网ID
service_ip	String	独享引擎实例的业务面IP
security_group_ids	Array of strings	独享引擎绑定的安全组
status	Integer	独享引擎计费状态 <ul style="list-style-type: none">0: 正常计费1: 冻结,资源和数据会保留,但租户无法再正常使用云服务2: 终止,资源和数据将清除
run_status	Integer	独享引擎运行状态 <ul style="list-style-type: none">0: 创建中1: 运行中2: 删除中3: 已删除4: 创建失败5: 已冻结6: 异常7: 更新中8: 更新失败
access_status	Integer	独享引擎接入状态(0: 未接入, 1: 已接入)
upgradable	Integer	独享引擎是否可升级(0: 不可升级, 1: 可升级)
cloudServiceType	String	云服务代码。仅作为标记,用户可忽略。
resourceType	String	云服务资源类型,仅作为标记,用户可忽略。
resourceSpecCode	String	云服务资源代码。仅作为标记,用户可忽略。
specification	String	独享引擎ECS规格,如"8vCPUs 16GB"

参数	参数类型	描述
hosts	Array of IdHostnameEntry objects	独享引擎防护的域名
serverId	String	独享引擎ECS ID
create_time	Long	引擎实例创建时间
instance_name	String	独享引擎实例名称

表 4-1223 IdHostnameEntry

参数	参数类型	描述
id	String	防护域名ID
hostname	String	防护域名

状态码： 400

表 4-1224 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1225 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1226 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1227 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1228 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1229 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的WAF独享引擎列表。

```
GET https://{endpoint}/v1/{project_id}/premium-waf/instance
```

响应示例

状态码： 200

独享引擎信息列表

```
{
  "purchased": true,
  "total": 1,
  "items": [ {
    "id": "0619871acb764d48a112695e8f7cbb10",
    "region": "region-01-7",
    "zone": "region-01-7a",
    "specification": "8vCPUs | 16GB",
    "arch": "x86",
    "upgradable": 0,
    "status": 0,
    "serverId": "477353dc-8687-4bf4-b45b-1d7fee74fa63",
    "cloudServiceType": "hws.service.type.waf",
    "resourceType": "hws.resource.type.waf.instance",
    "resourceSpecCode": "waf.instance.enterprise",
    "vpc_id": "13718074-a3f9-408d-82aa-3c41ef55e589",
    "subnet_id": "74d1b5a6-c7eb-4e9a-8372-181212552fcc",
    "service_ip": "192.168.10.68",
    "security_group_ids": [ "34287bdb-7aba-471a-b041-27427f1af76a" ],
    "cpu_flavor": "Si2.2xlarge.2",
    "run_status": 2,
    "access_status": 1,
    "hosts": [ {
      "id": "c3be17bbe3a641c7a1ded6019c377402",
      "hostname": "demo.www.com"
    } ],
    "instance_name": "0412elb",
    "create_time": 1649217360674
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ListInstanceRequest request = new ListInstanceRequest();
        try {
            ListInstanceResponse response = client.listInstance(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
```

```
.with_credentials(credentials) \  
.with_region(WafRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = ListInstanceRequest()  
    response = client.list_instance(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListInstanceRequest{}  
    response, err := client.ListInstance(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	独享引擎信息列表
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.10.2 创建 WAF 独享引擎实例

功能介绍

创建WAF独享引擎实例。独享模式支持的局点包括：华东-青岛、中东-利雅得、华北-北京一、华北-北京四、华北-乌兰察布一、华东-上海一、华东-上海二、华南-广州、华南-深圳、中国-香港、西南-贵阳一、亚太-曼谷、亚太-新加坡、非洲-约翰内斯堡、土耳其-伊斯坦布尔；普通租户类独享支持的局点：华北-北京四、华东-上海一、华南-广州、中国-香港、亚太-曼谷、亚太-新加坡。

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/premium-waf/instance

表 4-1230 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1231 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1232 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)。
Content-Type	是	String	内容类型

表 4-1233 请求 Body 参数

参数	是否必选	参数类型	描述
chargemode	否	Integer	收费模式，当前仅支持按需收费（30）
region	是	String	需要创建独享引擎的局点，例如：北京四（cn-north-4）
available_zone	是	String	需要创建独享引擎的可用区，例如：北京四可用区1（cn-north-4a）
arch	是	String	独享引擎CPU架构，仅支持x86
instancetype	是	String	独享引擎名称前缀
specification	是	String	独享引擎版本规格 <ul style="list-style-type: none">waf.instance.enterprise：企业版，对应控制台WI-500规格waf.instance.professional：专业版，对应控制台WI-100规格
cpu_flavor	否	String	独享引擎ECS规格，实例规格。创建资源租户类独享引擎可不填该参数，创建普通资源租户独享引擎必填该参数。普通租户类独享引擎具体支持的规格以waf控制台上支持的规格为准。
vpc_id	是	String	独享引擎所在VPC的ID
subnet_id	是	String	独享引擎所在VPC内的子网ID
security_group	是	Array of strings	独享引擎需要绑定的安全组ID
count	是	Integer	申请的独享引擎数量

参数	是否必选	参数类型	描述
res_tenant	否	Boolean	是否为资源租户类，默认值为 false。 <ul style="list-style-type: none"> • true: 资源租户类 • false: 普通租户类
anti_affinity	否	Boolean	是否开启反亲和。仅资源租户独享实例支持该特性。
tags	否	Array of TmsResource Tag objects	TMS标签信息

表 4-1234 TmsResourceTag

参数	是否必选	参数类型	描述
key	否	String	标签键
value	否	String	标签值

响应参数

状态码： 200

表 4-1235 响应 Body 参数

参数	参数类型	描述
instances	Array of instanceInfo objects	instances

表 4-1236 instanceInfo

参数	参数类型	描述
id	String	引擎实例ID
name	String	引擎实例名称

状态码： 400

表 4-1237 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1238 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1239 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1240 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1241 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1242 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为project_id的项目中创建WAF独享引擎实例, WAF独享引擎实例的收费模式为按需收费, 需要创建独享引擎的局点为“region-01-4”, 创建独享引擎的可用区为“region-01-4a”, 独享引擎CPU架构为“x86”, 独享引擎名称前缀为“demo”, 独享引擎版本规格为企业版, 独享引擎所在VPC的ID为“d7b6a5ff-6c53-4cd4-9d57-f20ee8753056”, 独享引擎所在VPC内的子网ID为“e59ccd18-7e15-4588-b689-04b856f4e78b”, 独享引擎需要绑定的安全组ID为“09b156a2-f0f0-41fd-9891-60e594601cfd”, 数量为一个, 是 资源租户类。

```
POST https://{endpoint}/v1/{project_id}/premium-waf/instance
```

```
{
  "chargemode": 30,
  "region": "region-01-4",
  "available_zone": "region-01-4a",
```



```
"arch": "x86",
"instancetype": "demo",
"specification": "waf.instance.enterprise",
"vpc_id": "d7b6a5ff-6c53-4cd4-9d57-f20ee8753056",
"subnet_id": "e59ccd18-7e15-4588-b689-04b856f4e78b",
"security_group": [ "09b156a2-f0f0-41fd-9891-60e594601cfd" ],
"count": 1,
"res_tenant": true
}
```

响应示例

状态码： 200

独享引擎信息

```
{
  "instances": [ {
    "id": "50a6b6c9bdb643f9a8038976fc58ad02",
    "name": "demo-6wvl"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为project_id的项目中创建WAF独享引擎实例，WAF独享引擎实例的收费模式为按需收费，需要创建独享引擎的局点为“region-01-4”，创建独享引擎的可用区为“region-01-4a”，独享引擎CPU架构为“x86”，独享引擎名称前缀为“demo”，独享引擎版本规格为企业版，独享引擎所在VPC的ID为“d7b6a5ff-6c53-4cd4-9d57-f20ee8753056”，独享引擎所在VPC内的子网ID为“e59ccd18-7e15-4588-b689-04b856f4e78b”，独享引擎需要绑定的安全组ID为“09b156a2-f0f0-41fd-9891-60e594601cfd”，数量为一个，是资源租户类。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
```

```
.withProjectId(projectId)
.withAk(ak)
.withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
CreateInstanceRequest request = new CreateInstanceRequest();
CreateInstanceRequestBody body = new CreateInstanceRequestBody();
List<String> listbodySecurityGroup = new ArrayList<>();
listbodySecurityGroup.add("09b156a2-f0f0-41fd-9891-60e594601cfd");
body.withResTenant(true);
body.withCount(1);
body.withSecurityGroup(listbodySecurityGroup);
body.withSubnetId("e59ccd18-7e15-4588-b689-04b856f4e78b");
body.withVpcId("d7b6a5ff-6c53-4cd4-9d57-f20ee8753056");
body.withSpecification("waf.instance.enterprise");
body.withInstancename("demo");
body.withArch("x86");
body.withAvailableZone("region-01-4a");
body.withRegion("region-01-4");
body.withChargemode(30);
request.withBody(body);
try {
    CreateInstanceResponse response = client.createInstance(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为project_id的项目中创建WAF独享引擎实例，WAF独享引擎实例的收费模式为按需收费，需要创建独享引擎的局点为“region-01-4”，创建独享引擎的可用区为“region-01-4a”，独享引擎CPU架构为“x86”，独享引擎名称前缀为“demo”，独享引擎版本规格为企业版，独享引擎所在VPC的ID为“d7b6a5ff-6c53-4cd4-9d57-f20ee8753056”，独享引擎所在VPC内的子网ID为“e59ccd18-7e15-4588-b689-04b856f4e78b”，独享引擎需要绑定的安全组ID为“09b156a2-f0f0-41fd-9891-60e594601cfd”，数量为一个，是资源租户类。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
```

```
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateInstanceRequest()
    listSecurityGroupbody = [
        "09b156a2-f0f0-41fd-9891-60e594601cfd"
    ]
    request.body = CreateInstanceRequestBody(
        res_tenant=True,
        count=1,
        security_group=listSecurityGroupbody,
        subnet_id="e59ccd18-7e15-4588-b689-04b856f4e78b",
        vpc_id="d7b6a5ff-6c53-4cd4-9d57-f20ee8753056",
        specification="waf.instance.enterprise",
        instancename="demo",
        arch="x86",
        available_zone="region-01-4a",
        region="region-01-4",
        chargemode=30
    )
    response = client.create_instance(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

在项目id为project_id的项目中创建WAF独享引擎实例，WAF独享引擎实例的收费模式为按需收费，需要创建独享引擎的局点为“region-01-4”，创建独享引擎的可用区为“region-01-4a”，独享引擎CPU架构为“x86”，独享引擎名称前缀为“demo”，独享引擎版本规格为企业版，独享引擎所在VPC的ID为“d7b6a5ff-6c53-4cd4-9d57-f20ee8753056”，独享引擎所在VPC内的子网ID为“e59ccd18-7e15-4588-b689-04b856f4e78b”，独享引擎需要绑定的安全组ID为“09b156a2-f0f0-41fd-9891-60e594601cfd”，数量为一个，是资源租户类。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

```
WithSk(sk).
WithProjectId(projectId).
Build()

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateInstanceRequest{}
var listSecurityGroupbody = []string{
    "09b156a2-f0f0-41fd-9891-60e594601cfd",
}
resTenantCreateInstanceRequestBody:= true
chargemodeCreateInstanceRequestBody:= int32(30)
request.Body = &model.CreateInstanceRequestBody{
    ResTenant: &resTenantCreateInstanceRequestBody,
    Count: int32(1),
    SecurityGroup: listSecurityGroupbody,
    SubnetId: "e59ccd18-7e15-4588-b689-04b856f4e78b",
    VpcId: "d7b6a5ff-6c53-4cd4-9d57-f20ee8753056",
    Specification: "waf.instance.enterprise",
    Instancename: "demo",
    Arch: "x86",
    AvailableZone: "region-01-4a",
    Region: "region-01-4",
    Chargemode: &chargemodeCreateInstanceRequestBody,
}
response, err := client.CreateInstance(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	独享引擎信息
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.10.3 查询 WAF 独享引擎信息

功能介绍

查询WAF独享引擎信息。独享模式只在部分局点支持，包括：华北-北京四、华东-上海一、华南-广州、华南-深圳、中国-香港、亚太-曼谷、亚太-新加坡。

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/premium-waf/instance/{instance_id}

表 4-1243 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	独享引擎ID（通过调用WAF的ListInstance接口获取所有独享引擎信息查询独享引擎ID）

表 4-1244 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1245 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1246 响应 Body 参数

参数	参数类型	描述
id	String	独享引擎实例ID
instancetype	String	独享引擎实例名称
region	String	独享引擎实例Region ID
zone	String	可用区ID
arch	String	CPU架构
cpu_flavor	String	ECS规格
vpc_id	String	独享引擎实例所在VPC ID
subnet_id	String	独享引擎实例所在VPC的子网ID
service_ip	String	独享引擎实例的业务面IP
security_group_ids	Array of strings	独享引擎绑定的安全组
status	Integer	独享引擎计费状态 <ul style="list-style-type: none">0: 正常计费1: 冻结,资源和数据会保留,但租户无法再正常使用云服务2: 终止,资源和数据将清除
run_status	Integer	独享引擎运行状态 <ul style="list-style-type: none">0: 创建中1: 运行中2: 删除中3: 已删除4: 创建失败5: 已冻结6: 异常7: 更新中8: 更新失败
access_status	Integer	独享引擎接入状态 (0: 未接入, 1: 已接入)
upgradable	Integer	独享引擎是否可升级 (0: 不可升级, 1: 可升级)
cloudServiceType	String	云服务代码。仅作为标记, 用户可忽略。

参数	参数类型	描述
resourceType	String	云服务资源类型，仅作为标记，用户可忽略。
resourceSpecCode	String	云服务资源代码。仅作为标记，用户可忽略。
specification	String	独享引擎ECS规格，如"8vCPUs 16GB"
serverId	String	独享引擎ECS ID
create_time	Long	引擎实例创建时间

状态码： 400

表 4-1247 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1248 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1249 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1250 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1251 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1252 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，独享引擎ID为instance_id的WAF独享引擎信息。

```
GET https://{endpoint}/v1/{project_id}/premium-waf/instance/{instance_id}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "0619871acb764d48a112695e8f7cbb10",
  "region": "region-01-7",
  "zone": "region-01-7a",
  "specification": "8vCPUs | 16GB",
  "arch": "x86",
  "upgradable": 0,
  "status": 0,
  "serverId": "477353dc-8687-4bf4-b45b-1d7fee74fa63",
  "cloudServiceType": "hws.service.type.waf",
  "resourceType": "hws.resource.type.waf.instance",
  "resourceSpecCode": "waf.instance.enterprise",
  "vpc_id": "13718074-a3f9-408d-82aa-3c41ef55e589",
  "subnet_id": "74d1b5a6-c7eb-4e9a-8372-181212552fcc",
  "service_ip": "192.168.10.68",
  "security_group_ids": [ "34287bdb-7aba-471a-b041-27427f1af76a" ],
  "cpu_flavor": "Si2.2xlarge.2",
  "run_status": 2,
  "access_status": 1,
  "instancename": "0412elb",
  "create_time": 1649217360674
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
```

```
.withProjectId(projectId)
.withAk(ak)
.withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ShowInstanceRequest request = new ShowInstanceRequest();
request.withInstanceId("{instance_id}");
try {
    ShowInstanceResponse response = client.showInstance(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceRequest()
        request.instance_id = "{instance_id}"
        response = client.show_instance(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main
```

```
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := waf.NewWafClient(  
        waf.WafClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ShowInstanceRequest{}  
    request.InstanceId = "{instance_id}"  
    response, err := client.ShowInstance(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.10.4 重命名 WAF 独享引擎

功能介绍

重命名WAF独享引擎。独享模式只在部分局点支持，包括：华北-北京四、华东-上海一、华南-广州、华南-深圳、中国-香港、亚太-曼谷、亚太-新加坡。

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/premium-waf/instance/{instance_id}

表 4-1253 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	独享引擎ID（通过调用WAF的ListInstance接口获取所有独享引擎信息查询独享引擎ID）

表 4-1254 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1255 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)。
Content-Type	是	String	内容类型

表 4-1256 请求 Body 参数

参数	是否必选	参数类型	描述
instancetype	是	String	独享引擎新名称

响应参数

状态码： 200

表 4-1257 响应 Body 参数

参数	参数类型	描述
id	String	独享引擎实例ID
instancetype	String	独享引擎实例名称
region	String	独享引擎实例Region ID
zone	String	可用区ID
arch	String	CPU架构
cpu_flavor	String	ECS规格
vpc_id	String	独享引擎实例所在VPC ID
subnet_id	String	独享引擎实例所在VPC的子网ID
service_ip	String	独享引擎实例的业务面IP
security_group_ids	Array of strings	独享引擎绑定的安全组
status	Integer	独享引擎计费状态 <ul style="list-style-type: none">0: 正常计费1: 冻结,资源和数据会保留,但租户无法再正常使用云服务2: 终止,资源和数据将清除
run_status	Integer	独享引擎运行状态 <ul style="list-style-type: none">0: 创建中1: 运行中2: 删除中3: 已删除4: 创建失败5: 已冻结6: 异常7: 更新中8: 更新失败

参数	参数类型	描述
access_status	Integer	独享引擎接入状态（0：未接入，1：已接入）
upgradable	Integer	独享引擎是否可升级（0：不可升级，1：可升级）
cloudServiceType	String	云服务代码。仅作为标记，用户可忽略。
resourceType	String	云服务资源类型，仅作为标记，用户可忽略。
resourceSpecCode	String	云服务资源代码。仅作为标记，用户可忽略。
specification	String	独享引擎ECS规格，如"8vCPUs 16GB"
serverId	String	独享引擎ECS ID
create_time	Long	引擎实例创建时间

状态码： 400

表 4-1258 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码（加密）后的详细拒绝原因，用户可以自行调用 STS 服务的decode-authorization-message接口进行解码，可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合，IAM5鉴权错误时才会返回此字段。

表 4-1259 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401**表 4-1260 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1261 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1262 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1263 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id，独享引擎ID为instance_id的WAF独享引擎的独享引擎新名称为“0412elb”。

```
PUT https://{endpoint}/v1/{project_id}/premium-waf/instance/{instance_id}
{
  "instancename": "0412elb"
}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "0619871acb764d48a112695e8f7cbb10",
  "region": "region-01-7",
  "zone": "region-01-7a",
  "specification": "8vCPUs | 16GB",
  "arch": "x86",
  "upgradable": 0,
  "status": 0,
  "serverId": "477353dc-8687-4bf4-b45b-1d7fee74fa63",
  "cloudServiceType": "hws.service.type.waf",
  "resourceType": "hws.resource.type.waf.instance",
  "resourceSpecCode": "waf.instance.enterprise",
  "vpc_id": "13718074-a3f9-408d-82aa-3c41ef55e589",
  "subnet_id": "74d1b5a6-c7eb-4e9a-8372-181212552fcc",
  "service_ip": "192.168.10.68",
  "security_group_ids": [ "34287bdb-7aba-471a-b041-27427f1af76a" ],
  "cpu_flavor": "Si2.2xlarge.2",
  "run_status": 2,
  "access_status": 1,
  "instancename": "0412elb",
  "create_time": 1649217360674
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id，独享引擎ID为instance_id的WAF独享引擎的独享引擎新名称为“0412elb”。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
```



```
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class RenameInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        RenameInstanceRequest request = new RenameInstanceRequest();
        request.withInstanceId("{instance_id}");
        RenameInstanceRequestBody body = new RenameInstanceRequestBody();
        body.withInstancename("0412elb");
        request.withBody(body);
        try {
            RenameInstanceResponse response = client.renameInstance(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新项目id为project_id，独享引擎ID为instance_id的WAF独享引擎的独享引擎新名称为“0412elb”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = RenameInstanceRequest()
    request.instance_id = "{instance_id}"
    request.body = RenameInstanceRequestBody(
        instancename="0412elb"
    )
    response = client.rename_instance(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

更新项目id为project_id，独享引擎ID为instance_id的WAF独享引擎的独享引擎新名称为“0412elb”。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RenameInstanceRequest{}
    request.InstanceId = "{instance_id}"
    request.Body = &model.RenameInstanceRequestBody{
        Instancename: "0412elb",
    }
    response, err := client.RenameInstance(request)
```

```
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.10.5 删除 WAF 独享引擎信息

功能介绍

删除WAF独享引擎信息。独享模式只在部分局点支持，包括：华北-北京四、华东-上海一、华南-广州、华南-深圳、中国-香港、亚太-曼谷、亚太-新加坡。

调用方法

请参见[如何调用API](#)。

URI

DELETE /v1/{project_id}/premium-waf/instance/{instance_id}

表 4-1264 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
instance_id	是	String	独享引擎ID（通过调用WAF的ListInstance接口获取所有独享引擎信息查询独享引擎ID）

表 4-1265 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1266 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1267 响应 Body 参数

参数	参数类型	描述
id	String	独享引擎实例ID
instancetype	String	独享引擎实例名称
region	String	独享引擎实例Region ID
zone	String	可用区ID
arch	String	CPU架构
cpu_flavor	String	ECS规格
vpc_id	String	独享引擎实例所在VPC ID
subnet_id	String	独享引擎实例所在VPC的子网ID
service_ip	String	独享引擎实例的业务面IP
security_group_ids	Array of strings	独享引擎绑定的安全组

参数	参数类型	描述
status	Integer	独享引擎计费状态 <ul style="list-style-type: none">0: 正常计费1: 冻结,资源和数据会保留,但租户无法再正常使用云服务2: 终止,资源和数据将清除
run_status	Integer	独享引擎运行状态 <ul style="list-style-type: none">0: 创建中1: 运行中2: 删除中3: 已删除4: 创建失败5: 已冻结6: 异常7: 更新中8: 更新失败
access_status	Integer	独享引擎接入状态 (0: 未接入, 1: 已接入)
upgradable	Integer	独享引擎是否可升级 (0: 不可升级, 1: 可升级)
cloudServiceType	String	云服务代码。仅作为标记,用户可忽略。
resourceType	String	云服务资源类型,仅作为标记,用户可忽略。
resourceSpecCode	String	云服务资源代码。仅作为标记,用户可忽略。
specification	String	独享引擎ECS规格,如"8vCPUs 16GB"
serverId	String	独享引擎ECS ID
create_time	Long	引擎实例创建时间

状态码: 400

表 4-1268 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1269 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1270 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1271 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1272 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1273 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

删除项目id为project_id, 独享引擎ID为instance_id的WAF独享引擎信息。

```
DELETE https://{endpoint}z/v1/{project_id}/premium-waf/instance/{instance_id}
```

响应示例

状态码： 200

请求成功

```
{
  "id": "0619871acb764d48a112695e8f7cbb10",
  "region": "region-01-7",
  "zone": "region-01-7a",
  "specification": "8vCPUs | 16GB",
  "arch": "x86",
  "upgradable": 0,
  "status": 0,
  "serverId": "477353dc-8687-4bf4-b45b-1d7fee74fa63",
  "cloudServiceType": "hws.service.type.waf",
  "resourceType": "hws.resource.type.waf.instance",
  "resourceSpecCode": "waf.instance.enterprise",
  "vpc_id": "13718074-a3f9-408d-82aa-3c41ef55e589",
  "subnet_id": "74d1b5a6-c7eb-4e9a-8372-181212552fcc",
  "service_ip": "192.168.10.68",
  "security_group_ids": [ "34287bdb-7aba-471a-b041-27427f1af76a" ],
  "cpu_flavor": "Si2.2xlarge.2",
```

```
"run_status" : 2,  
"access_status" : 1,  
"instancename" : "0412elb"  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class DeleteInstanceSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeleteInstanceRequest request = new DeleteInstanceRequest();  
        request.withInstanceId("{instance_id}");  
        try {  
            DeleteInstanceResponse response = client.deleteInstance(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8
```



```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteInstanceRequest()
        request.instance_id = "{instance_id}"
        response = client.delete_instance(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteInstanceRequest{}
```

```
request.InstanceId = "{instance_id}"
response, err := client.DeleteInstance(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.11 日志配置管理

4.11.1 配置全量日志 lts

功能介绍

配置全量日志lts，该接口可用来开启与关闭waf全量日志以及配置日志组和日志流。日志组id和日志流id可前往云日志服务获取。配置的日志流id要属于所配置的日志组。

调用方法

请参见[如何调用API](#)。

URI

PUT /v1/{project_id}/waf/config/lts/{ltsconfig_id}

表 4-1274 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 对应控制台用户名->我的凭证->项目列表->项目ID
ltsconfig_id	是	String	lts配置信息id, 通过 ShowLtsInfoConfig获取

表 4-1275 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务 (EPS) 的查询企业项目列表接口 (ListEnterpriseProject) 查询企业项目id

请求参数

表 4-1276 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)。
Content-Type	是	String	内容类型

表 4-1277 请求 Body 参数

参数	是否必选	参数类型	描述
enabled	否	Boolean	是否开启全量日志 <ul style="list-style-type: none">• false: 不开启• true: 开启
ltsIdInfo	否	LtsIdInfo object	日志组与日志流id信息
enabale	否	Boolean	该参数废弃, 请忽略

表 4-1278 LtsIdInfo

参数	是否必选	参数类型	描述
ltsGroupId	否	String	日志组id
ltsAccessStreamID	否	String	访问日志流id
ltsAttackStreamID	否	String	攻击日志流id

响应参数

状态码： 200

表 4-1279 响应 Body 参数

参数	参数类型	描述
id	String	lts配置信息id
enabled	Boolean	是否开启全量日志 <ul style="list-style-type: none"> • false: 不开启 • true: 开启
ltsIdInfo	LtsIdInfo object	日志组与日志流id信息
enabale	Boolean	该参数已废弃，请忽略

表 4-1280 LtsIdInfo

参数	参数类型	描述
ltsGroupId	String	日志组id
ltsAccessStreamID	String	访问日志流id
ltsAttackStreamID	String	攻击日志流id

状态码： 400

表 4-1281 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1282 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1283 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1284 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1285 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1286 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id, lts配置信息id为c89a667487734f6a95e9967d1f373c77的全量日志lts信息。全量日志lts开启全量日志, 日志组id为"4bcff74d-f649-41c8-8325-1b0a264ff683", 访问日志流id为:"0a7ef713-cc3e-418d-abda-85df04db1a3c", 攻击日志流id "f4fa07f6-277b-4e4a-a257-26508ece81e6"。

```
PUT https://{Endpoint}/v1/{project_id}/waf/config/lts/c89a667487734f6a95e9967d1f373c77?enterprise_project_id=0
```

```
{
  "enabled": true,
  "ltsIdInfo": {
    "ltsAccessStreamID": "4bcff74d-f649-41c8-8325-1b0a264ff683",
    "ltsAttackStreamID": "0a7ef713-cc3e-418d-abda-85df04db1a3c",
    "ltsGroupID": "f4fa07f6-277b-4e4a-a257-26508ece81e6"
  }
}
```

响应示例

状态码： 200

请求成功

```
{
  "enabled": true,
  "id": "c89a667487734f6a95e9967d1f373c77",
}
```

```
"ltsIdInfo" : {  
  "ltsAccessStreamID" : "4bcff74d-f649-41c8-8325-1b0a264ff683",  
  "ltsAttackStreamID" : "0a7ef713-cc3e-418d-abda-85df04db1a3c",  
  "ltsGroupId" : "f4fa07f6-277b-4e4a-a257-26508ece81e6"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id, lts配置信息id为c89a667487734f6a95e9967d1f373c77的全量日志lts信息。全量日志lts开启全量日志, 日志组id为"4bcff74d-f649-41c8-8325-1b0a264ff683", 访问日志流id为:"0a7ef713-cc3e-418d-abda-85df04db1a3c", 攻击日志流id "f4fa07f6-277b-4e4a-a257-26508ece81e6"。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class UpdateLtsInfoConfigSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateLtsInfoConfigRequest request = new UpdateLtsInfoConfigRequest();  
        request.withLtsconfigId("{ltsconfig_id}");  
        UpdateLtsInfoConfigRequestBody body = new UpdateLtsInfoConfigRequestBody();  
        LtsIdInfo ltsIdInfo = new LtsIdInfo();  
        ltsIdInfo.withLtsGroupId("f4fa07f6-277b-4e4a-a257-26508ece81e6")  
            .withLtsAccessStreamID("4bcff74d-f649-41c8-8325-1b0a264ff683")  
            .withLtsAttackStreamID("0a7ef713-cc3e-418d-abda-85df04db1a3c");  
        body.withLtsIdInfo(ltsIdInfo);  
        body.withEnabled(true);  
        request.withBody(body);  
        try {  
            UpdateLtsInfoConfigResponse response = client.updateLtsInfoConfig(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {
```

```
e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新项目id为project_id, lts配置信息id为c89a667487734f6a95e9967d1f373c77的全量日志lts信息。全量日志lts开启全量日志, 日志组id为"4bcff74d-f649-41c8-8325-1b0a264ff683", 访问日志流id为:"0a7ef713-cc3e-418d-abda-85df04db1a3c", 攻击日志流id "f4fa07f6-277b-4e4a-a257-26508ece81e6"。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateLtsInfoConfigRequest()
        request.ltsconfig_id = "{ltsconfig_id}"
        ltsIdInfoBody = LtsIdInfo(
            lts_group_id="f4fa07f6-277b-4e4a-a257-26508ece81e6",
            lts_access_stream_id="4bcff74d-f649-41c8-8325-1b0a264ff683",
            lts_attack_stream_id="0a7ef713-cc3e-418d-abda-85df04db1a3c"
        )
        request.body = UpdateLtsInfoConfigRequestBody(
            lts_id_info=ltsIdInfoBody,
            enabled=True
        )
        response = client.update_lts_info_config(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新项目id为project_id, lts配置信息id为c89a667487734f6a95e9967d1f373c77的全量日志lts信息。全量日志lts开启全量日志, 日志组id为"4bcff74d-

f649-41c8-8325-1b0a264ff683", 访问日志流id为:"0a7ef713-cc3e-418d-abda-85df04db1a3c", 攻击日志流id "f4fa07f6-277b-4e4a-a257-26508ece81e6"。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateLtsInfoConfigRequest{}
    request.LtsconfigId = "{ltsconfig_id}"
    ltsGroupIdLtsIdInfo := "f4fa07f6-277b-4e4a-a257-26508ece81e6"
    ltsAccessStreamIDLtsIdInfo := "4bcff74d-f649-41c8-8325-1b0a264ff683"
    ltsAttackStreamIDLtsIdInfo := "0a7ef713-cc3e-418d-abda-85df04db1a3c"
    ltsIdInfoBody := &model.LtsIdInfo{
        LtsGroupId: &ltsGroupIdLtsIdInfo,
        LtsAccessStreamID: &ltsAccessStreamIDLtsIdInfo,
        LtsAttackStreamID: &ltsAttackStreamIDLtsIdInfo,
    }
    enabledUpdateLtsInfoConfigRequestBody := true
    request.Body = &model.UpdateLtsInfoConfigRequestBody{
        LtsIdInfo: ltsIdInfoBody,
        Enabled: &enabledUpdateLtsInfoConfigRequestBody,
    }
    response, err := client.UpdateLtsInfoConfig(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.11.2 查询 lts 配置信息

功能介绍

查询lts配置信息

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/config/lts

表 4-1287 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1288 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1289 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1290 响应 Body 参数

参数	参数类型	描述
id	String	lts配置信息id，每个企业项目对应唯一id
enabled	Boolean	是否开启全量日志 <ul style="list-style-type: none">• false: 不开启• true: 开启
ltsIdInfo	LtsIdInfo object	日志组与日志流id信息
enabale	Boolean	该参数已废弃，请忽略

表 4-1291 LtsIdInfo

参数	参数类型	描述
ltsGroupId	String	日志组id
ltsAccessStreamID	String	访问日志流id
ltsAttackStreamID	String	攻击日志流id

状态码： 400

表 4-1292 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1293 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1294 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1295 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1296 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1297 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的lts配置信息

```
GET https://{Endpoint}/v1/{project_id}/waf/config/lts?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "enabled": true,
  "id": "c89a667487734f6a95e9967d1f373c77",
  "ltsIdInfo": {
    "ltsAccessStreamID": "4bcff74d-f649-41c8-8325-1b0a264ff683",
    "ltsAttackStreamID": "0a7ef713-cc3e-418d-abda-85df04db1a3c",
    "ltsGroupId": "f4fa07f6-277b-4e4a-a257-26508ece81e6"
  }
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowLtsInfoConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowLtsInfoConfigRequest request = new ShowLtsInfoConfigRequest();
        try {
            ShowLtsInfoConfigResponse response = client.showLtsInfoConfig(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ShowLtsInfoConfigRequest()
    response = client.show_lts_info_config(request)
    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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowLtsInfoConfigRequest{}
    response, err := client.ShowLtsInfoConfig(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.12 租户订购管理

4.12.1 购买包周期云模式 waf

功能介绍

购买包周期云模式waf。入门版waf不支持购买扩展包

接口约束

如果付款方式无法完成支付，系统会自动生成一个待支付订单，您可以手动在控制台上选择支付方式进行支付。（仅在北京四、北京一、上海二、上海一、广州以及广州友好用户环境支持入门版）

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/subscription/purchase/prepaid-cloud-waf

表 4-1298 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户的project_id，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1299 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1300 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)。
Content-Type	是	String	内容类型

表 4-1301 请求 Body 参数

参数	是否必选	参数类型	描述
project_id	是	String	项目id
is_auto_pay	是	Boolean	是否自动支付 <ul style="list-style-type: none">• false: 否（需要客户手动去支付）• true: 是（自动支付）
is_auto_renew	是	Boolean	是否自动续订 <ul style="list-style-type: none">• true: 自动续订• false: 不自动续订
region_id	是	String	region Id
waf_product_info	否	WafProductInfo object	购买waf信息
domain_expand_product_info	否	ExpandProductInfo object	购买域名扩展包信息
bandwidth_expand_product_info	否	ExpandProductInfo object	购买带宽扩展包信息

参数	是否必选	参数类型	描述
rule_expack_product_info	否	ExpackProductInfo object	购买规则扩展包信息

表 4-1302 WafProductInfo

参数	是否必选	参数类型	描述
resource_spec_code	否	String	waf规格 <ul style="list-style-type: none">• detection: 入门（仅在北京四、北京一、上海二、上海一、广州以及广州-友好用户环境支持入门版）• professional: 标准• enterprise: 专业• ultimate: 铂金版
period_type	否	String	订购周期类型 month: 月; year: 年
period_num	否	Integer	订购周期数

表 4-1303 ExpackProductInfo

参数	是否必选	参数类型	描述
resource_size	否	Integer	扩展包数量

响应参数

状态码： 200

表 4-1304 响应 Body 参数

参数	参数类型	描述
orderId	String	订单id

状态码： 400

表 4-1305 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1306 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1307 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1308 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1309 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1310 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

在项目id为“550500b49078408682d0d4f7d923f3e1”的项目中购买包周期云模式waf, waf不自动续订, 不自动支付, region Id为“cn-north-7”, 带宽扩展包的扩展包数量为“1”。

```
POST https://{Endpoint}/v1/{project_id}/waf/subscription/purchase/prepaid-cloud-waf?
enterprise_project_id=0
{
  "project_id": "550500b49078408682d0d4f7d923f3e1",
  "is_auto_renew": false,
  "is_auto_pay": false,
  "region_id": "cn-north-7",
  "bandwidth_expack_product_info": {
    "resource_size": 1
  }
}
```

```
}  
}
```

响应示例

状态码： 200

请求成功

```
{  
  "orderId" : "38ff0cb9a10e4d5293c642bc0350fa6d"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

在项目id为“550500b49078408682d0d4f7d923f3e1”的项目中购买包周期云模式waf，waf不自动续订，不自动支付，regionId为“cn-north-7”，带宽扩展包的扩展包数量为“1”。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class CreatePrepaidCloudWafSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
  
        CreatePrepaidCloudWafRequest request = new CreatePrepaidCloudWafRequest();  
        CreatePrepaidCloudWafRequestBody body = new CreatePrepaidCloudWafRequestBody();  
        ExpackProductInfo bandwidthExpackProductInfo = new ExpackProductInfo();  
        bandwidthExpackProductInfo.withResourceSize(1);  
        body.withBandwidthExpackProductInfo(bandwidthExpackProductInfo);  
        body.withRegionId("cn-north-7");  
        body.withIsAutoRenew(false);  
        body.withIsAutoPay(false);  
        body.withProjectId("550500b49078408682d0d4f7d923f3e1");  
        request.withBody(body);  
    }  
}
```

```
try {
    CreatePrepaidCloudWafResponse response = client.createPrepaidCloudWaf(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

在项目id为“550500b49078408682d0d4f7d923f3e1”的项目中购买包周期云模式waf，waf不自动续订，不自动支付，regionId为“cn-north-7”，带宽扩展包的扩展包数量为“1”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreatePrepaidCloudWafRequest()
        bandwidthExpackProductInfobody = ExpackProductInfo(
            resource_size=1
        )
        request.body = CreatePrepaidCloudWafRequestBody(
            bandwidth_expack_product_info=bandwidthExpackProductInfobody,
            region_id="cn-north-7",
            is_auto_renew=False,
            is_auto_pay=False,
            project_id="550500b49078408682d0d4f7d923f3e1"
        )
        response = client.create_prepaid_cloud_waf(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在项目id为“550500b49078408682d0d4f7d923f3e1”的项目中购买包周期云模式waf，waf不自动续订，不自动支付，region Id为“cn-north-7”，带宽扩展包的扩展包数量为“1”。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreatePrepaidCloudWafRequest{}
    resourceSizeBandwidthExpackProductInfo := int32(1)
    bandwidthExpackProductInfoBody := &model.ExpackProductInfo{
        ResourceSize: &resourceSizeBandwidthExpackProductInfo,
    }
    request.Body = &model.CreatePrepaidCloudWafRequestBody{
        BandwidthExpackProductInfo: bandwidthExpackProductInfoBody,
        RegionId: "cn-north-7",
        IsAutoRenew: false,
        IsAutoPay: false,
        ProjectId: "550500b49078408682d0d4f7d923f3e1",
    }
    response, err := client.CreatePrepaidCloudWaf(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.12.2 变更包周期云模式 waf 规格

功能介绍

变更包周期云模式waf规格。

- 变更某产品规格的前提是必须已购买该产品
- waf版本只支持升配，不支持降配；扩展包数量可以增加或者减少，但不支持数量减少为0
- 不支持同时升降配，如增加域名扩展包数量，同时减少规则扩展包数量

接口约束

如果付款方式无法完成支付，系统会自动生成一个待支付订单，您可以手动在控制台上选择支付方式进行支付。

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/waf/subscription/batchalter/prepaid-cloud-waf

表 4-1311 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户的project_id，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1312 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1313 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)。
Content-Type	是	String	内容类型

表 4-1314 请求 Body 参数

参数	是否必选	参数类型	描述
project_id	是	String	项目id
is_auto_pay	是	Boolean	是否自动支付 <ul style="list-style-type: none">• false: 否（需要客户手动去支付）• true: 是（自动支付）
waf_product_info	否	AlterWafProductInfo object	变更waf版本信息
domain_expand_product_info	否	ExpandProductInfo object	变更域名扩展包信息
bandwidth_expand_product_info	否	ExpandProductInfo object	变更带宽扩展包信息
rule_expand_product_info	否	ExpandProductInfo object	变更规则扩展包信息

表 4-1315 AlterWafProductInfo

参数	是否必选	参数类型	描述
resource_spec_code	否	String	waf规格 <ul style="list-style-type: none">• detection: 入门版• professional: 标准版• enterprise: 专业版• ultimate: 铂金版

表 4-1316 ExpackProductInfo

参数	是否必选	参数类型	描述
resource_size	否	Integer	扩展包数量

响应参数

状态码： 200

表 4-1317 响应 Body 参数

参数	参数类型	描述
orderId	String	订单id

状态码： 400

表 4-1318 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1319 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1320 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1321 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1322 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1323 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

变更项目id为“550500b49078408682d0d4f7d923f3e1”的包周期云模式waf规格, 修改为可以自动支付, 带宽扩展包的扩展包数量为“2”。

```
POST https://{Endpoint}/v1/{project_id}/waf/subscription/batchalter/prepaid-cloud-waf?
enterprise_project_id=0

{
  "project_id": "550500b49078408682d0d4f7d923f3e1",
  "is_auto_pay": true,
  "domain_expack_product_info": {
    "resource_size": 2
  }
}
```

响应示例

状态码: 200

请求成功

```
{
  "orderId": "38ff0cb9a10e4d5293c642bc0350fa6d"
}
```

SDK 代码示例

SDK代码示例如下。

Java

变更项目id为“550500b49078408682d0d4f7d923f3e1”的包周期云模式waf规格, 修改为可以自动支付, 带宽扩展包的扩展包数量为“2”。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ChangePrepaidCloudWafSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ChangePrepaidCloudWafRequest request = new ChangePrepaidCloudWafRequest();
        ChangePrepaidCloudWafRequestBody body = new ChangePrepaidCloudWafRequestBody();
        ExpackProductInfo domainExpackProductInfobody = new ExpackProductInfo();
        domainExpackProductInfobody.withResourceSize(2);
        body.withDomainExpackProductInfo(domainExpackProductInfobody);
        body.withIsAutoPay(true);
        body.withProjectId("550500b49078408682d0d4f7d923f3e1");
        request.withBody(body);
        try {
            ChangePrepaidCloudWafResponse response = client.changePrepaidCloudWaf(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

变更项目id为“550500b49078408682d0d4f7d923f3e1”的包周期云模式waf规格，修改为可以自动支付，带宽扩展包的扩展包数量为“2”。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
```

```
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ChangePrepaidCloudWafRequest()
        domainExpackProductInfoBody = ExpackProductInfo(
            resource_size=2
        )
        request.body = ChangePrepaidCloudWafRequestBody(
            domain_expack_product_info=domainExpackProductInfoBody,
            is_auto_pay=True,
            project_id="550500b49078408682d0d4f7d923f3e1"
        )
        response = client.change_prepaid_cloud_waf(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

变更项目id为“550500b49078408682d0d4f7d923f3e1”的包周期云模式waf规格，修改为可以自动支付，带宽扩展包的扩展包数量为“2”。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
```

```
waf.WafClientBuilder().
  WithRegion(region.ValueOf("<YOUR REGION>")).
  WithCredential(auth).
  Build())

request := &model.ChangePrepaidCloudWafRequest{
  resourceSizeDomainExpackProductInfo:= int32(2)
  domainExpackProductInfobody := &model.ExpactProductInfo{
    ResourceSize: &resourceSizeDomainExpackProductInfo,
  }
}
request.Body = &model.ChangePrepaidCloudWafRequestBody{
  DomainExpactProductInfo: domainExpactProductInfobody,
  IsAutoPay: true,
  ProjectId: "550500b49078408682d0d4f7d923f3e1",
}
response, err := client.ChangePrepaidCloudWaf(request)
if err == nil {
  fmt.Printf("%+v\n", response)
} else {
  fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.12.3 查询租户订购信息

功能介绍

查询租户订购信息，包括云模式包周期、按需计费、独享模式

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/subscription

表 4-1324 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户的project_id, 对应控制台用户名->我的凭证->项目列表->项目ID

请求参数

表 4-1325 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-1326 响应 Body 参数

参数	参数类型	描述
type	Integer	云模式版本 <ul style="list-style-type: none">-2: 已冻结-1: 未订购2: 标准版3: 专业版4: 铂金版7: 入门版22: 按需版本
resources	Array of ResourceResponse objects	资源列表
isNewUser	Boolean	是否为新用户
premium	Premium object	独享模式订购信息

表 4-1327 ResourceResponse

参数	参数类型	描述
resourceId	String	资源id
cloudServiceType	String	云服务产品对应的云服务类型
resourceType	String	云服务产品的资源类型 <ul style="list-style-type: none">hws.resource.type.waf: 云模式包周期WAFhws.resource.type.waf.domain: 云模式包周期WAF域名扩展包hws.resource.type.waf.bandwidth: 云模式包周期WAF带宽扩展包hws.resource.type.waf.rule: 云模式包周期WAF规则扩展包hws.resource.type.waf.payperuserrequest: Web应用防火墙按需请求hws.resource.type.waf.payperusedomain: Web应用防火墙按需域名hws.resource.type.waf.payperuserule: Web应用防火墙按需规则
resourceSpecCode	String	云服务产品的资源规格
status	Integer	资源状态 <ul style="list-style-type: none">0: 解冻/正常1: 冻结2: 删除
expireTime	String	资源到期时间
resourceSize	Integer	资源数量

表 4-1328 Premium

参数	参数类型	描述
purchased	Boolean	是否开通独享模式
total	Integer	独享实例数量, 包括elb
elb	Integer	elb实例数量
dedicated	Integer	独享WAF实例数量

状态码: 400

表 4-1329 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1330 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1331 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1332 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500

表 4-1333 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1334 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的租户订购信息

```
GET https://{Endpoint}/v1/{project_id}/waf/subscription?
```

响应示例

状态码： 200

请求成功

```
{  
  "type" : 3,  
  "resources" : [ {  
    "resourceId" : "d2759a06ed844b9c9837bb76326ca656",
```

```
"cloudServiceType" : "hws.service.type.waf",
"resourceType" : "hws.resource.type.waf",
"resourceSpecCode" : "waf.enterprise",
"resourceSize" : null,
"expireTime" : "2022-12-07T15:59:59Z",
"status" : 0
}, {
"resourceId" : "6a5a4b06dbcd4cc5be6ff88bcd988046",
"cloudServiceType" : "hws.service.type.waf",
"resourceType" : "hws.resource.type.waf.rule",
"resourceSpecCode" : "waf.expack.rule.enterprise",
"resourceSize" : 5,
"expireTime" : "2022-12-07T15:59:59Z",
"status" : 0
}, {
"resourceId" : "a9202ca8704740b6a1e0481c80bd4255",
"cloudServiceType" : "hws.service.type.waf",
"resourceType" : "hws.resource.type.waf.domain",
"resourceSpecCode" : "waf.expack.domain.enterprise",
"resourceSize" : 10,
"expireTime" : "2022-12-07T15:59:59Z",
"status" : 0
}
}],
"isNewUser" : false,
"premium" : {
"purchased" : true,
"total" : 8,
"elb" : 0,
"dedicated" : 8
}
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowSubscriptionInfoSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
```

```
        .withCredential(auth)
        .withRegion(WafRegion.valueOf("<YOUR REGION>"))
        .build();
ShowSubscriptionInfoRequest request = new ShowSubscriptionInfoRequest();
try {
    ShowSubscriptionInfoResponse response = client.showSubscriptionInfo(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowSubscriptionInfoRequest()
        response = client.show_subscription_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowSubscriptionInfoRequest{}
    response, err := client.ShowSubscriptionInfo(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.13 租户域名查询

4.13.1 查询全部防护域名列表

功能介绍

查询全部防护域名列表，包括云模式和独享模式

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/composite-waf/host

表 4-1335 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1336 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id。默认值为0，表示默认企业项目。 缺省值：0
page	否	Integer	分页查询时，返回第几页数据。默认值为1，表示返回第1页数据。
pagesize	否	Integer	分页查询时，每页包含多少条结果。范围1-100，默认值为10，表示每页包含10条结果。如果需要一次查全部域名，该参数值填-1。
hostname	否	String	域名名称
policyname	否	String	防护策略名称

参数	是否必选	参数类型	描述
protect_status	否	Integer	域名防护状态： <ul style="list-style-type: none">-1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测1: 开启防护, WAF根据您配置的策略进行攻击检测
waf_type	否	String	域名所属WAF模式
is_https	否	Boolean	域名是否使用HTTPS

请求参数

表 4-1337 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)。
Content-Type	是	String	内容类型

响应参数

状态码: 200

表 4-1338 响应 Body 参数

参数	参数类型	描述
total	Integer	所有防护域名的数量
cloud_total	Integer	云模式防护域名的数量
premium_total	Integer	独享模式防护域名的数量
items	Array of CompositeHostResponse objects	详细的防护域名信息

表 4-1339 CompositeHostResponse

参数	参数类型	描述
id	String	域名id
hostid	String	域名id
hostname	String	创建的云模式防护域名
policyid	String	策略id
access_code	String	cname前缀
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">• -1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF• 0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测• 1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">• false: 不使用代理• true: 使用代理
timestamp	Long	创建防护域名的时间
paid_type	String	套餐付费模式, 默认值为prePaid。 prePaid: 包周期款模式; postPaid: 按需模式。
flag	Flag object	特殊标识, 用于前端使用
waf_type	String	域名所属WAF模式, cloud为云模式, premium为独享模式
web_tag	String	网站名称, 对应WAF控制台域名详情中的网站名称
access_progress	Array of Access_progress objects	接入进度, 仅用于新版console(前端)使用
premium_waf_instances	Array of Premium_waf_instances objects	租户引擎实例信息列表
description	String	域名描述

参数	参数类型	描述
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip, 该参数为预留参数, 用于后续功能扩展, 用户可忽略 <ul style="list-style-type: none">• true: 使用独享ip• false: 不使用独享ip
region	String	区域ID, 控制台创建的域名会携带此参数, api调用创建的域名此参数为空, 可以通过地区和终端节点文档查询区域ID对应的中文名称
server	Array of WafServer objects	防护域名的源站服务器配置信息, 只有独享模式域名才返回vpc_id
enterprise_project_id	String	企业项目id

表 4-1340 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

表 4-1341 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none">• 1: 指回源IP加白• 2: 指本地验证• 3: 指修改DNS解析
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

表 4-1342 Premium_waf_instances

参数	参数类型	描述
id	String	引擎实例id
name	String	引擎实例名
accessed	Boolean	引擎实例是否已接入, false: 未接入; true: 已接入

表 4-1343 WafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重, 负载均衡算法将按该权重将请求分配给源站, 默认值是1, 云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

参数	参数类型	描述
vpc_id	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none">1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

状态码: 400

表 4-1344 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1345 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1346 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1347 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1348 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1349 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的全部防护域名列表。

```
GET https://{Endpoint}/v1/{project_id}/composite-waf/host?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{
  "items": [ {
    "id": "31af669f567246c289771694f2112289",
    "hostid": "31af669f567246c289771694f2112289",
    "description": "",
    "proxy": false,
    "flag": {
      "pci_3ds": "false",
      "pci_dss": "false",
      "ipv6": "false",
      "cname": "new",
      "is_dual_az": "true"
    },
    "region": "cn-north-4",
    "hostname": "www.demo.com",
    "access_code": "1b18879b9d064f8bbcbf8abce7294cac",
    "policyid": "41cba8aee2e94bcdbf57460874205494",
    "timestamp": 1650527546454,
    "protect_status": 0,
    "access_status": 0,
    "exclusive_ip": false,
    "web_tag": "",
    "paid_type": "prePaid",
    "waf_type": "cloud"
  } ],
  "total": 1,
  "cloud_total": 1,
  "premium_total": 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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ListCompositeHostsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
ListCompositeHostsRequest request = new ListCompositeHostsRequest();
try {
    ListCompositeHostsResponse response = client.listCompositeHosts(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListCompositeHostsRequest()
        response = client.list_composite_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListCompositeHostsRequest{}
    response, err := client.ListCompositeHosts(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.13.2 根据 Id 查询防护域名

功能介绍

根据Id查询防护域名

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/composite-waf/host/{host_id}

表 4-1350 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID
host_id	是	String	域名id，通过查询云模式防护域名列表（ListHost）获取域名id或者通过独享模式域名列表（ListPremiumHost）获取域名id

表 4-1351 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1352 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1353 响应 Body 参数

参数	参数类型	描述
id	String	域名id
hostid	String	域名id
hostname	String	创建的云模式防护域名
policyid	String	策略id
access_code	String	cname前缀
protect_status	Integer	域名防护状态： <ul style="list-style-type: none">-1: bypass, 该域名的请求直接到达其后端服务器, 不再经过WAF0: 暂停防护, WAF只转发该域名的请求, 不做攻击检测1: 开启防护, WAF根据您配置的策略进行攻击检测
access_status	Integer	域名接入状态, 0表示未接入, 1表示已接入
proxy	Boolean	防护域名是否使用代理 <ul style="list-style-type: none">false: 不使用代理true: 使用代理
timestamp	Long	创建防护域名的时间
paid_type	String	套餐付费模式, 默认值为prePaid。 prePaid: 包周期款模式; postPaid: 按需模式。
flag	Flag object	特殊标识, 用于前端使用
waf_type	String	域名所属WAF模式, cloud为云模式, premium为独享模式
web_tag	String	网站名称, 对应WAF控制台域名详情中的网站名称
access_progress	Array of Access_progress objects	接入进度, 仅用于新版console(前端)使用
premium_waf_instances	Array of Premium_waf_instances objects	租户引擎实例信息列表
description	String	域名描述

参数	参数类型	描述
exclusive_ip	Boolean	是否使用用户独享的引擎公网ip, 该参数为预留参数, 用于后续功能扩展, 用户可忽略 <ul style="list-style-type: none">• true: 使用独享ip• false: 不使用独享ip
region	String	区域ID, 控制台创建的域名会携带此参数, api调用创建的域名此参数为空, 可以通过地区和终端节点文档查询区域ID对应的中文名称
server	Array of WafServer objects	防护域名的源站服务器配置信息, 只有独享模式域名才返回vpc_id
enterprise_project_id	String	企业项目id

表 4-1354 Flag

参数	参数类型	描述
pci_3ds	String	是否开启pci_3ds合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
pci_dss	String	是否开启pci_dss合规认证 <ul style="list-style-type: none">• true: 开启• false: 不开启
cname	String	old: 代表域名使用的老的cname, new: 代表域名使用新的cname
is_dual_az	String	是否支持双AZ模式 <ul style="list-style-type: none">• true: 支持• false: 不支持
ipv6	String	域名是否开启ipv6 <ul style="list-style-type: none">• true: 支持• false: 不支持

表 4-1355 Access_progress

参数	参数类型	描述
step	Integer	步骤 <ul style="list-style-type: none">• 1: 指回源IP加白• 2: 指本地验证• 3: 指修改DNS解析
status	Integer	状态, 0: 未完成这个步骤; 1: 已完成这个状态”

表 4-1356 Premium_waf_instances

参数	参数类型	描述
id	String	引擎实例id
name	String	引擎实例名
accessed	Boolean	引擎实例是否已接入, false: 未接入; true: 已接入

表 4-1357 WafServer

参数	参数类型	描述
front_protocol	String	客户端请求访问防护域名源站服务器的协议
back_protocol	String	WAF转发客户端请求到防护域名源站服务器的协议
weight	Integer	源站权重, 负载均衡算法将按该权重将请求分配给源站, 默认值是1, 云模式的冗余字段
address	String	客户端访问的源站服务器的IP地址
port	Integer	WAF转发客户端请求到源站服务的业务端口
type	String	源站地址为ipv4或ipv6

参数	参数类型	描述
vpc_id	String	VPC id,通过以下步骤获取VPC id: <ul style="list-style-type: none">1.找到独享引擎所在的虚拟私有云名称, VPC\子网这一列就是VPC的名称: 登录WAF的控制台->单击系统管理->独享引擎->VPC\子网2.登录虚拟私有云 VPC控制台->虚拟私有云->单击虚拟私有云的名称->基本信息的ID

状态码: 400

表 4-1358 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1359 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1360 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1361 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1362 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1363 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id，域名id为host_id的域名信息。

```
GET https://{Endpoint}/v1/{project_id}/composite-waf/host/{host_id}?enterprise_project_id=0
```

响应示例

状态码： 200

OK

```
{
  "id": "31af669f567246c289771694f2112289",
  "hostid": "31af669f567246c289771694f2112289",
  "description": "",
  "proxy": false,
  "flag": {
    "pci_3ds": "false",
    "pci_dss": "false",
    "ipv6": "false",
    "cname": "new",
    "is_dual_az": "true"
  },
  "region": "cn-north-4",
  "hostname": "www.demo.com",
  "access_code": "1b18879b9d064f8bbcbf8abce7294cac",
  "policyid": "41cba8aee2e94bcdbf57460874205494",
  "timestamp": 1650527546454,
  "protect_status": 0,
  "access_status": 0,
  "exclusive_ip": false,
  "web_tag": "",
  "paid_type": "prePaid",
  "waf_type": "cloud",
  "server": [ {
    "address": "1.2.3.4",
    "port": 443,
    "type": "ipv4",
    "weight": 1,
    "front_protocol": "HTTPS",
    "back_protocol": "HTTPS",
    "vpc_id": "ebfc553a-386d-4746-b0c2-18ff3f0e903d"
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowCompositeHostSolution {
```

```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");
    String projectId = "{project_id}";

    ICredential auth = new BasicCredentials()
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

    WafClient client = WafClient.newBuilder()
        .withCredential(auth)
        .withRegion(WafRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowCompositeHostRequest request = new ShowCompositeHostRequest();
    request.withHostId("{host_id}");
    try {
        ShowCompositeHostResponse response = client.showCompositeHost(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowCompositeHostRequest()
        request.host_id = "{host_id}"
        response = client.show_composite_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowCompositeHostRequest{}
    request.HostId = "{host_id}"
    response, err := client.ShowCompositeHost(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	请求失败
401	token权限不足

状态码	描述
500	服务器内部错误

错误码

请参见[错误码](#)。

4.14 租户防护域名管理

4.14.1 按企业项目迁移防护域名

功能介绍

按企业项目迁移防护域名，仅专业版与独享版支持该功能

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/composite-waf/hosts/migration

表 4-1364 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1365 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	是	String	当前企业项目ID
target_enterprise_project_id	是	String	迁移的目标企业项目ID

请求参数

表 4-1366 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)。
Content-Type	是	String	内容类型

表 4-1367 请求 Body 参数

参数	是否必选	参数类型	描述
host_ids	是	Array of strings	host_id列表
policy_id	是	String	策略ID（目标企业项目下的策略ID）
certificate_id	否	String	证书ID（目标企业项目下的证书ID）

响应参数

状态码： 200

表 4-1368 响应 Body 参数

参数	参数类型	描述
host_ids	Array of strings	host_id列表
policy_id	String	策略ID（目标企业项目下的策略ID）
certificate_id	String	证书ID（目标企业项目下的证书ID）

状态码： 400

表 4-1369 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1370 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1371 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1372 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1373 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1374 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

将项目id为project_id的项目下的域名id为“6cc26af071de46479840b41c143bb79b”的域名和防护策略id为“b97ce5d4e38d4eaf94d65be7372974a8”的防护策略从企业id为0的企业, 迁移到企业id为“c341b6bd-0935-462b-abb5-28a63bd68021”的企业。

```
POST https://{Endpoint}/v1/{project_id}/composite-waf/hosts/migration?
enterprise_project_id=0&target_enterprise_project_id=c341b6bd-0935-462b-abb5-28a63bd68021

{
  "host_ids": [ "6cc26af071de46479840b41c143bb79b" ],
  "policy_id": "b97ce5d4e38d4eaf94d65be7372974a8"
}
```

响应示例

状态码： 200

请求成功

```
{
  "host_ids": [ "6cc26af071de46479840b41c143bb79b" ],
  "policy_id": "b97ce5d4e38d4eaf94d65be7372974a8"
}
```

SDK 代码示例

SDK代码示例如下。

Java

将项目id为project_id的项目下的域名id为“6cc26af071de46479840b41c143bb79b”的域名和防护策略id为“b97ce5d4e38d4eaf94d65be7372974a8”的防护策略从企业id为0的企业，迁移到企业id为“c341b6bd-0935-462b-abb5-28a63bd68021”的企业。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class MigrateCompositeHostsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();

        MigrateCompositeHostsRequest request = new MigrateCompositeHostsRequest();
        MigrateCompositeHostsRequestBody body = new MigrateCompositeHostsRequestBody();
        List<String> listbodyHostIds = new ArrayList<>();
        listbodyHostIds.add("6cc26af071de46479840b41c143bb79b");
        body.withPolicyId("b97ce5d4e38d4eaf94d65be7372974a8");
        body.withHostIds(listbodyHostIds);
        request.withBody(body);
        try {
            MigrateCompositeHostsResponse response = client.migrateCompositeHosts(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

将项目id为project_id的项目下的域名id为“6cc26af071de46479840b41c143bb79b”的域名和防护策略id为“b97ce5d4e38d4eaf94d65be7372974a8”的防护策略从企业id为0的企业，迁移到企业id为“c341b6bd-0935-462b-abb5-28a63bd68021”的企业。

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkwaf.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = WafClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(WafRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = MigrateCompositeHostsRequest()  
        listHostIdsbody = [  
            "6cc26af071de46479840b41c143bb79b"  
        ]  
        request.body = MigrateCompositeHostsRequestBody(  
            policy_id="b97ce5d4e38d4eaf94d65be7372974a8",  
            host_ids=listHostIdsbody  
        )  
        response = client.migrate_composite_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

将项目id为project_id的项目下的域名id为“6cc26af071de46479840b41c143bb79b”的域名和防护策略id为“b97ce5d4e38d4eaf94d65be7372974a8”的防护策略从企业id为0的企业，迁移到企业id为“c341b6bd-0935-462b-abb5-28a63bd68021”的企业。

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
  )

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.MigrateCompositeHostsRequest{}
    var listHostIdsbody = []string{
        "6cc26af071de46479840b41c143bb79b",
    }
    request.Body = &model.MigrateCompositeHostsRequestBody{
        PolicyId: "b97ce5d4e38d4eaf94d65be7372974a8",
        HostIds: listHostIdsbody,
    }
    response, err := client.MigrateCompositeHosts(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.15 系统管理

4.15.1 查询 WAF 回源 IP 信息

功能介绍

查询WAF回源IP信息

调用方法

请参见[如何调用API](#)。

URI

GET /v1/{project_id}/waf/config/source-ip

表 4-1375 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

请求参数

表 4-1376 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1377 响应 Body 参数

参数	参数类型	描述
source_ip	Array of IpsItem objects	源站信息列表

参数	参数类型	描述
last_modify	Long	回源IP最后更新时间

表 4-1378 IpsItem

参数	参数类型	描述
ips	Array of strings	waf回源IP
update_time	Long	回源IP更新时间

状态码： 400

表 4-1379 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1380 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1381 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1382 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1383 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1384 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的WAF回源IP信息。

```
GET https://{endpoint}/v1/{project_id}/waf/config/source-ip
```

响应示例

状态码： 200

WAF回源IP信息

```
{
  "source_ip": [ {
    "ips": [ "122.112.208.32/28", "49.4.56.64/27", "2407:c080:804::/48" ],
    "update_time": 1573779840000
  } ]
}
```

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.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class ShowSourceIpSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        WafClient client = WafClient.newBuilder()
            .withCredential(auth)
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowSourceIpRequest request = new ShowSourceIpRequest();
        try {
            ShowSourceIpResponse response = client.showSourceIp(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowSourceIpRequest()
        response = client.show_source_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := waf.NewWafClient(
    waf.WafClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowSourceIpRequest{}
response, err := client.ShowSourceIp(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	WAF回源IP信息
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.16 告警管理

4.16.1 查询告警通知配置

功能介绍

查询告警通知配置

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/waf/alerts

表 4-1385 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，对应控制台用户名->我的凭证->项目列表->项目ID

表 4-1386 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	您可以通过调用企业项目管理服务（EPS）的查询企业项目列表接口（ListEnterpriseProject）查询企业项目id

请求参数

表 4-1387 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)。
Content-Type	是	String	内容类型

响应参数

状态码： 200

表 4-1388 响应 Body 参数

参数	参数类型	描述
total	Integer	配置的告警通知总数量
items	Array of AlertNoticeConfigResponse objects	配置的告警通知

表 4-1389 AlertNoticeConfigResponse

参数	参数类型	描述
id	String	ID
name	String	告警通知名称
enabled	Boolean	是否开启 <ul style="list-style-type: none">false: 不开启true: 开启
topic_urn	String	主题
sendfreq	Integer	时间间隔，单位为分钟。当通知类型为防护事件时，该参数表示在该时间间隔内，攻击次数等于或者大于设定阈值时，将发送告警通知，支持的值：5、15、30、60、120、360、720、1440；当通知类型为证书到期时，该参数表示每隔多长时间发送一次告警通知，支持的值1天、1周（需要转换成分钟）。
locale	String	语言
times	Integer	当通知类型为防护事件时，需要填写该参数。在该时间间隔内，当攻击次数大于或等于您设置的阈值时才会发送告警通知
threat	Array of strings	事件类型
prefer_html	Boolean	预留参数，可忽略
notice_class	String	通知类型
nearly_expired_time	String	提前通知天数
is_all_enterprise_project	Boolean	是否是所有企业项目
enterprise_project_id	String	企业项目ID
update_time	Long	更新时间

状态码： 400

表 4-1390 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码

参数	参数类型	描述
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1391 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 401

表 4-1392 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1393 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 500**表 4-1394 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1395 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

查询项目id为project_id的告警通知配置。

```
GET https://{Endpoint}/v2/{project_id}/waf/alerts?enterprise_project_id=0
```

响应示例

状态码： 200

请求成功

```
{
  "total": 1,
  "items": [ {
    "enabled": true,
    "enterprise_project_id": "0",
    "id": "753231205d474fa78655760c8dbd9e6f",
    "is_all_enterprise_project": true,
    "locale": "zh-cn",
    "name": "test-demo33",
    "nearly_expired_time": 60,
    "notice_class": "cert_alert_notice",
    "prefer_html": false,
    "sendfreq": 10080,
    "threat": [ ],
    "times": 1,
    "topic_urn": "urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test",
    "update_time": 1664347553944
  }
]
```

```
    }  
  }  
}
```

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.waf.v1.region.WafRegion;  
import com.huaweicloud.sdk.waf.v1.*;  
import com.huaweicloud.sdk.waf.v1.model.*;  
  
public class ListNoticeConfigsSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        WafClient client = WafClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(WafRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListNoticeConfigsRequest request = new ListNoticeConfigsRequest();  
        try {  
            ListNoticeConfigsResponse response = client.listNoticeConfigs(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException 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 huaweicloudsdkwaf.v1.region.waf_region import WafRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = WafClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(WafRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListNoticeConfigsRequest()
        response = client.list_notice_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"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListNoticeConfigsRequest{}
    response, err := client.ListNoticeConfigs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
```

```
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

4.16.2 更新告警通知配置

功能介绍

更新告警通知配置

调用方法

请参见[如何调用API](#)。

URI

PUT /v2/{project_id}/waf/alert/{alert_id}

表 4-1396 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户项目ID
alert_id	是	String	告警通知id

请求参数

表 4-1397 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	租户token
Content-Type	是	String	内容类型
X-Language	是	String	zh-cn/en-us

表 4-1398 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	告警通知名称
enabled	否	Boolean	是否开启 <ul style="list-style-type: none">• false: 不开启• true: 开启
topic_urn	是	String	主题URN, 通过“消息通知服务”获取
sendfreq	否	Integer	时间间隔, 单位为分钟。当通知类型为防护事件时, 该参数表示在该时间间隔内, 攻击次数等于或者大于设定阈值时, 将发送告警通知, 支持的值: 5、15、30、60、120、360、720、1440; 当通知类型为证书到期时, 该参数表示每隔多长时间发送一次告警通知, 支持的值为1440、10080 (单位为分钟)。
locale	否	String	语言 <ul style="list-style-type: none">• zh-cn: 中文• en-us: 英文
times	否	Integer	当通知类型为防护事件时, 需要填写该参数。在该时间间隔内, 当攻击次数大于或等于您设置的阈值时才会发送告警通知
threat	否	Array of strings	事件类型

参数	是否必选	参数类型	描述
notice_class	是	String	通知类型 <ul style="list-style-type: none">threat_alert_notice: 防护事件cert_alert_notice: 证书到期
nearly_expire_d_time	否	String	提前通知天数，通知类型为证书到期通知需要填写该参数
is_all_enterprise_project	否	Boolean	是否是所有企业项目

响应参数

状态码： 200

表 4-1399 响应 Body 参数

参数	参数类型	描述
id	String	ID
name	String	告警通知名称
enabled	Boolean	是否开启 <ul style="list-style-type: none">false: 不开启true: 开启
topic_urn	String	主题
sendfreq	Integer	时间间隔，单位为分钟。当通知类型为防护事件时，该参数表示在该时间间隔内，攻击次数等于或者大于设定阈值时，将发送告警通知，支持的值为：5、15、30、60、120、360、720、1440；当通知类型为证书到期时，该参数表示每隔多长时间发送一次告警通知，支持的值为1440、10080（单位为分钟）。
locale	String	语言
times	Integer	当通知类型为防护事件时，需要填写该参数。在该时间间隔内，当攻击次数大于或等于您设置的阈值时才会发送告警通知
threat	Array of strings	事件类型
prefer_html	Boolean	预留参数，可忽略
notice_class	String	通知类型

参数	参数类型	描述
nearly_expired_time	String	提前通知天数
is_all_enterprise_project	Boolean	是否是所有企业项目
enterprise_project_id	String	企业项目ID
update_time	Long	更新时间

状态码： 400

表 4-1400 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1401 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码： 401

表 4-1402 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息

参数	参数类型	描述
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1403 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

状态码: 500

表 4-1404 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码
error_msg	String	错误信息
encoded_authorization_message	String	编码 (加密) 后的详细拒绝原因, 用户可以自行调用 STS 服务的decode-authorization-message接口进行解码, 可参考STS5联调自验证。IAM5鉴权错误时才会返回此字段。
details	Array of IAM5ErrorDetails objects	调用下游服务的报错信息集合, IAM5鉴权错误时才会返回此字段。

表 4-1405 IAM5ErrorDetails

参数	参数类型	描述
error_code	String	下游服务错误码
error_msg	String	下游服务报错信息

请求示例

更新项目id为project_id的告警通知配置，更新通知类型为证书到期，主题URN为"urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test"，告警通知名称为“test”，提前60天通知，时间间隔10080分钟一次。

```
PUT https://{Endpoint}/v2/{project_id}/waf/alert/{alert_id}?enterprise_project_id=0
{
  "notice_class": "cert_alert_notice",
  "topic_urn": "urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test",
  "name": "test",
  "nearly_expired_time": 60,
  "sendfreq": 10080
}
```

响应示例

状态码： 200

请求成功

```
{
  "enabled": true,
  "enterprise_project_id": "0",
  "id": "7a19ee86a7dc43f0b12093decb795096",
  "is_all_enterprise_project": true,
  "locale": "zh-cn",
  "name": "demo",
  "nearly_expired_time": 60,
  "notice_class": "cert_alert_notice",
  "prefer_html": false,
  "sendfreq": 10080,
  "threat": [],
  "times": 1,
  "topic_urn": "urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test"
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新项目id为project_id的告警通知配置，更新通知类型为证书到期，主题URN为"urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test"，告警通知名称为“test”，提前60天通知，时间间隔10080分钟一次。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.waf.v1.region.WafRegion;
import com.huaweicloud.sdk.waf.v1.*;
import com.huaweicloud.sdk.waf.v1.model.*;

public class UpdateAlertNoticeConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

WafClient client = WafClient.newBuilder()
    .withCredential(auth)
    .withRegion(WafRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateAlertNoticeConfigRequest request = new UpdateAlertNoticeConfigRequest();
request.withAlertId("{alert_id}");
UpdateAlertNoticeConfigRequestBody body = new UpdateAlertNoticeConfigRequestBody();
body.withNearlyExpiredTime("60");
body.withNoticeClass("cert_alert_notice");
body.withSendfreq(10080);
body.withTopicUrn("urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test");
body.withName("test");
request.withBody(body);
try {
    UpdateAlertNoticeConfigResponse response = client.updateAlertNoticeConfig(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新项目id为project_id的告警通知配置，更新通知类型为证书到期，主题URN为"urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test"，告警通知名称为“test”，提前60天通知，时间间隔10080分钟一次。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkwaf.v1.region.waf_region import WafRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkwaf.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
```

```
client = WafClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(WafRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateAlertNoticeConfigRequest()
    request.alert_id = "{alert_id}"
    request.body = UpdateAlertNoticeConfigRequestBody(
        nearly_expired_time="60",
        notice_class="cert_alert_notice",
        sendfreq=10080,
        topic_urn="urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test",
        name="test"
    )
    response = client.update_alert_notice_config(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

更新项目id为project_id的告警通知配置，更新通知类型为证书到期，主题URN为"urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test"，告警通知名称为“test”，提前60天通知，时间间隔10080分钟一次。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    waf "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/waf/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := waf.NewWafClient(
        waf.WafClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAlertNoticeConfigRequest{}
    request.AlertId = "{alert_id}"
    nearlyExpiredTimeUpdateAlertNoticeConfigRequestBody := "60"
    sendfreqUpdateAlertNoticeConfigRequestBody := int32(10080)
    request.Body = &model.UpdateAlertNoticeConfigRequestBody{
        NearlyExpiredTime: &nearlyExpiredTimeUpdateAlertNoticeConfigRequestBody,
```

```
NoticeClass: "cert_alert_notice",
Sendfreq: &sendfreqUpdateAlertNoticeConfigRequestBody,
TopicUrn: "urn:smn:cn-north-7:550500b49078408682d0d4f7d923f3e1:ces_zyh_test",
Name: "test",
}
response, err := client.UpdateAlertNoticeConfig(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求成功
400	请求失败
401	token权限不足
500	服务器内部错误

错误码

请参见[错误码](#)。

A 附录

A.1 状态码

- 正常

返回值	说明
200	请求成功。

- 异常

状态码	编码	说明
400	Bad Request	服务器未能处理请求。
401	Unauthorized	被请求的页面需要用户名和密码。
403	Forbidden	对被请求页面的访问被禁止。
404	Not Found	服务器无法找到被请求的页面。
405	Method Not Allowed	请求中指定的方法不被允许。
406	Not Acceptable	服务器生成的响应无法被客户端所接受。
407	Proxy Authentication Required	用户必须首先使用代理服务器进行验证，这样请求才会被处理。
408	Request Timeout	请求超出了服务器的等待时间。
409	Conflict	由于冲突，请求无法被完成。
500	Internal Server Error	请求未完成，服务异常。
501	Not Implemented	请求未完成，服务器不支持所请求的功能。

状态码	编码	说明
502	Bad Gateway	请求未完成，服务器从上游服务器收到一个无效的响应。
503	Service Unavailable	请求未完成，系统暂时异常。
504	Gateway Timeout	网关超时。

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	WAF.00011001	bad.request	非法请求	检查入参
400	WAF.00011002	url.param.illegal	URL格式不正确	检查url格式
400	WAF.00011003	request.body.illegal	请求Body格式错误：Body中的参数缺失、值非法	检查请求body
400	WAF.00011004	id.illegal	ID非法	检查ID
400	WAF.00011005	name.illegal	Name非法	检查name
400	WAF.00011006	host.illegal	域名非法	检查域名
400	WAF.00011007	port.illegal	端口非法	检查端口
400	WAF.00011007	ip.illegal	IP非法	检查IP地址
400	WAF.00011008	protect.status.illegal	防护状态非法	检查防护状态是否在枚举值范围
400	WAF.00011009	access.status.illegal	接入状态非法	检查接入状态是否在枚举值范围
400	WAF.00011010	offsetOrLimit.illegal	起始行或限制条数非法	检查起始行或者限制条数是否在范围内
400	WAF.00011011	pageOrPageSize.illegal	页码或每页条数非法	检查页码或者每页条数是否在范围内

状态码	错误码	错误信息	描述	处理措施
400	WAF.00011012	standard.violated	参数非法	检查参数
400	WAF.00011013	description.illegal	描述格式非法	检查描述格式
400	WAF.00011014	request.header.illegal	请求Header格式错误: Header中的参数缺失、值非法	检查header必填参数
400	WAF.00011015	spec.code.illegal	WAF规格不合法	替换合法WAF规格
400	WAF.00011016	name.duplicate	名称重复	修改名称
400	WAF.00011017	ipv6.switch.illegal	ipv6 防护不可关闭	开启ipv6防护
400	WAF.00011018	action.type.illegal	非法的动作类型	替换合法的防护动作类型
400	WAF.00011019	cert.illegal	证书非法	使用合法的证书
400	WAF.00011020	cve.num.illegal	CVE编号非法	使用合法的CVE编号
400	WAF.00011021	cert.expired	证书过期	替换未过期证书
400	WAF.00011022	gocm.action.illegal	非法的gocm action	检查GOCM单中的action, 使用正确合法的action规则
400	WAF.00011023	repeat.purchases	不允许重复购买	不允许重复购买
400	WAF.00012001	invalid.token	token非法	检查token是否正确
400	WAF.00012002	invalid.project	project_id和token中的一致	检查project_id和token的一致性
400	WAF.00013004	protocol.not.support	协议不支持	通过ELB转换协议
400	WAF.00013010	custom.rule.set.does.not.support.for.shared.policy	自定义规则集不支持共享策略	无

状态码	错误码	错误信息	描述	处理措施
400	WAF.00014002	resource.already.exists	资源已存在	资源已存在
400	WAF.00014003	open.protect.failed	开启防护失败	检查域名防护状态
400	WAF.00014004	access.failed	接入WAF失败	修改DNS解析
400	WAF.00014005	bypass.failed	BypassWAF失败	检查防护状态后进行重试
400	WAF.00014006	proxy.config.error	代理配置错误	重新正确配置代理后重试
400	WAF.00014007	host.conflict	域名冲突	检查网站配置中是否已存在域名
400	WAF.00014008	cert.inconsistent	相同的域名，但证书不一致	使用相同的证书
400	WAF.00014010	port.protocol.mismatch	端口和协议不匹配	选择匹配的协议和端口
400	WAF.00014011	host.blacklist	禁止添加该防护网站，域名被列入黑名单	无
400	WAF.00014012	website.not.register	网站未备案	备案网站
400	WAF.00014013	host.already.access	域名已接入	无
400	WAF.00014014	exclusive.ip.config.error	独享IP配置错误	检查独享IP配置
400	WAF.00014015	resource.is.being.used	资源正在使用中	无
400	WAF.00014016	ip.group.is.being.shared	IP地址组正在共享中	无
400	WAF.00014017	policy.is.being.shared	策略正在共享中	无
400	WAF.00014018	certificate.is.being.shared	证书正在共享中	无
400	WAF.00014019	policy.is.being.disconnecting	策略正在断联中	无
400	WAF.00014026	sdk.already.exists	已存在相同的SDK配置	请检查SDK配置

状态码	错误码	错误信息	描述	处理措施
400	WAF.00014027	certificate.already.exists	已存在相同的证书文件	请检查证书文件
400	WAF.00014028	host.already.exists	已存在相同的防护域名	请检查防护域名
400	WAF.00014029	certificateSharing.already.exists	目标企业项目中已存在当前证书	请检查目标企业项目下证书文件
400	WAF.00014030	policySharing.already.exists	目标企业项目中已存在当前防护策略	请检查目标企业项目下的防护策略
400	WAF.00014031	ipGroupSharing.already.exists	目标企业项目中已存在当前地址组	请检查目标企业项目下的ip地址组
400	WAF.00015001	premium.instance.not.available	独享实例不可用，请检查配置	检查配置
400	WAF.00015005	premium.instance.illegal.flavor	独享实例使用了非法的ECS规格	更换合法的ECS规格
400	WAF.00015006	premium.instance.purchase.config.not.found	独享引擎购买页选项未配置	请检查独享引擎购买页选项配置
400	WAF.00016001	elb.mode.not.available	ELB模式不可用	请检查ELB模式配置
400	WAF.00021001	bad.request	错误请求	无
400	WAF.00021002	url.param.illegal	url格式不正确	建议将请求体参数中url修改为标准的url后重新调试
400	WAF.00021003	request.body.illegal	请求体参数不正确	建议调试前根据文档检验入参
400	WAF.00021004	id.illegal	唯一标识符id格式不正确	建议按照文档中的正确说明获取ID
400	WAF.00021005	name.illegal	name参数格式不正确	检查name格式，只能由字母、数字、-、_和.组成,长度不能超过64个字符

状态码	错误码	错误信息	描述	处理措施
400	WAF.00021006	host.illegal	域名格式不正确	域名只能由字母、数字、-、_和.组成,长度不能超过64个字符
400	WAF.00021007	protocol.illegal	后端协议格式不正确	后端协议只能配置为HTTP或HTTPS,且必须大写
400	WAF.00021008	port.illegal	源站端口格式不正确	检查配置的端口是否为空,目标端口是否在0-65535范围内
400	WAF.00021009	ip.illegal	ip格式不正确	建议检查ip格式是否为标准的ipv4或ipv6格式
400	WAF.00021010	server.address.illegal	服务器配置异常	检查服务器配置是否为空,数量是否在1-80范围内
400	WAF.00021012	path.illegal	规则配置中的url格式不正确	建议将请求体参数中url修改为标准的url后重新调试
400	WAF.00021013	cert.illegal	https证书已经过期	建议重新上传未过期的证书
400	WAF.00021014	action.illegal	防护动作非法	建议根据文档中的枚举值配置防护动作
400	WAF.00021015	rule.status.illegal	规则状态非法	建议根据文档中的规则状态枚举值修改规则状态
400	WAF.00021016	description.illegal	描述异常	建议用标准的中英文语法描述
400	WAF.00021017	incorrect.rule.config	不正确的规则配置	建议根据帮助中心的文档配置防护规则
400	WAF.00021018	incorrect.reference.table.config	不正确的引用表配置	建议根据帮助中心的文档配置引用表
400	WAF.00021019	incorrect.route.config	不正确的线路配置	建议根据帮助中心的文档进行线路配置

状态码	错误码	错误信息	描述	处理措施
400	WAF.00021020	offsetOrLimit.illegal	分页参数错误	建议根据帮助中心的文档填写分页参数
400	WAF.00021021	pageOrPageSize.illegal	参数page或者pageSize非法	建议根据帮助中心的文档查看参数限制
400	WAF.00021022	name.duplicate	名称重复	修改名称
400	WAF.00021023	server.mode.illegal	模式非法	检查服务模式
400	WAF.00021024	proxyConfig.illegal	SDK配置非法	请检查SDK配置
400	WAF.00021025	cookie.secure.conflict	Cookie携带Secure/HttpOnly属性与HTTP的对外协议冲突	请检查配置
400	WAF.00021026	condition.repeat	条件列表里的规则重复	检查条件列表中的规则
400	WAF.00022002	resource.already.exists	资源已存在	建议在控制台查看所创建的资源是否已经存在
400	WAF.00022003	resource.is.being.used	资源正在使用中	先解除资源和使用者的关系，再删除资源
400	WAF.00022004	rule.conflict	规则冲突	检查目标规则是否和已有的规则发生冲突
400	WAF.00022006	host.conflict	其他人已经添加了该域名，请确认该域名是否属于你，如果是，请联系服务人员帮你解决	其他人已经添加了该域名，请确认该域名是否属于你，如果是，请联系服务人员帮你解决
400	WAF.00022007	open.protect.failed	开启防护失败，请您先把流量接入WAF	请先把流量接入WAF
400	WAF.00022012	rule.resource.already.exists	相同条件规则已存在	请检查添加的规则条件

状态码	错误码	错误信息	描述	处理措施
400	WAF.00023001	policy.not.bind.domain	该策略未绑定域名	请先绑定域名
400	WAF.00023002	domain.not.session.tag	域名未配置 session 流量标识	请先配置 session 流量标识
400	WAF.00023003	domain.not.user.tag	域名未配置 user 流量标识	请先配置 user 流量标识
401	WAF.00010005	request.iam.failed	调用 IAM 失败，请检查当前用户是否具有 IAM 读取权限（例如 IAM ReadOnly Access 权限）	请检查当前用户是否具有 IAM 读取权限（例如 IAM ReadOnly Access 权限）
401	WAF.00010006	update.iam.failed	调用 IAM 失败，请检查当前用户是否具有 IAM 写入权限（例如 Security Administrator 权限）	请检查当前用户是否具有 IAM 写入权限（例如 Security Administrator 权限）
401	WAF.00012003	permission.denied	没有权限	给账户分配 WAF 所需权限
401	WAF.00012008	jwt.authentication.invalid.token	JWT token 非法	检查账号是否具有 JWT 权限
401	WAF.00012009	jwt.authentication.failed	JWT 鉴权失败	先给账户授权
401	WAF.00012011	cop.permission.denied	没有 cop 权限	请检查 COP 权限
401	WAF.00015004	premium.instance.agency.not.ready	独享实例所需的 IAM 委托权限未就绪	请检查 IAM 委托权限
401	WAF.00016003	elb.mode.elb.unauthorized	用户没有权限读取与修改 ELB 配置信息	请检查用户权限
403	WAF.00012004	account.frozen	账号冻结	解除冻结
403	WAF.00012005	not.subscribe	未订阅	先订阅 WAF 服务

状态码	错误码	错误信息	描述	处理措施
403	WAF.00012006	pdp.permission.denied	没有权限	检查账户的PDP权限
403	WAF.00012007	jwt.authentication.disabled	JWT认证关闭	开通JWT认证
403	WAF.00012010	eps.all.not.support	所有企业项目下不支持写操作	开通企业项目的写权限
403	WAF.00012012	not.subscribe.cloud	迁移目标企业项目未订阅云模式引擎实例	请订阅云模式引擎实例
403	WAF.00012013	not.subscribe.premium	迁移目标企业项目未订阅独享引擎实例	请订阅独享引擎实例
403	WAF.00013001	insufficient.quota	功能配额不足	购买功能配额升级包
403	WAF.00013002	feature.not.support	功能不支持	无
403	WAF.00013003	port.not.support	端口不支持	通过ELB进行端口转换
403	WAF.00013005	wildcard.domain.not.support	泛域名不支持	使用具体的域名
403	WAF.00013006	ipv6.not.support	不支持IPV6	当前版本不支持IPV6
403	WAF.00013007	insufficient.tenant.quota	租户配额不足	购买配额升级包
403	WAF.00013008	product.sold.out	当前产品已售罄	更换产品规格
403	WAF.00013009	degrade.not.support	当前域名使用量超过降配后规格，不允许降配	减少域名使用量
403	WAF.00013011	insufficient.domain.quota	一级域名配额不足，请购买扩展包或者升级规格	购买扩展包或者升级规格
403	WAF.00013012	insufficient.port.quota	端口配额不足，请购买扩展包或者升级规格	请购买扩展包或者升级规格

状态码	错误码	错误信息	描述	处理措施
403	WAF.00013013	insufficient.host.quota	域名配额不足, 请购买扩展包或者升级规格	请购买扩展包或者升级规格
403	WAF.00013014	insufficient.policy.quota	策略配额不足	购买域名扩展包或者升级规格
403	WAF.00015002	premium.instance.sold.out	所选的独享实例规格已售罄, 请更换规格或等待补货	请更换规格或等待补货
403	WAF.00015003	premium.instance.not.allowed	该账户不被允许创建独享实例	检查当前账户配置
403	WAF.00016004	elb.mode unsupported.elb	该ELB不支持WAF。只有独享型ELB支持WAF	检查ELB模式
403	WAF.00016005	elb.mode unsupported.elb.spec	该ELB不支持WAF。请确认该ELB下存在应用型监听器	请检查ELB配置
403	WAF.00016006	elb.mode unsupported.elb.flavor	该ELB不支持WAF。请确认该ELB下存在指定规格的应用型监听器	请检查ELB配置
403	WAF.00022005	insufficient.quota	资源不足	建议购买对应资源的升级包
404	WAF.00014001	resource.not.found	资源找不到	资源已删除或不存在
404	WAF.00014009	api.not.found	接口不存在	检查接口url
404	WAF.00014020	certificate.not.found	证书未找到	请添加证书资源
404	WAF.00014021	ipGroup.not.found	IP地址组未找到	请添加ip地址组
404	WAF.00014022	host.not.found	防护域名未找到	请添加防护域名
404	WAF.00014023	premium.waf.instance.not.found	独享引擎实例未找到	请添加独享引擎实例

状态码	错误码	错误信息	描述	处理措施
404	WAF.00014024	projectId.not.found	项目Id未找到	请检查项目id
404	WAF.00014025	policy.not.found	防护策略未找到	添加防护策略
404	WAF.00016002	elb.mode.elb.not.found	未找到需要配置WAF的ELB	请检查ELB
404	WAF.00022001	resource.not.found	资源不存在	建议到控制台查看资源状态或求助技术支持
404	WAF.00022008	rule.not.found	策略规则未找到	请检查策略规则
404	WAF.00022009	certificate.not.found	证书未找到	请检查是否已添加证书资源
404	WAF.00022010	ipGroup.not.found	地址组未找到	请先添加ip地址组
404	WAF.00022011	policy.not.found	防护策略未找到	请先添加防护策略
409	WAF.00016007	elb.mode.conflict	一个ELB只允许绑定到一个WAF实例组，不管是独享实例组还是共享实例组	不允许重复绑定
409	WAF.00016008	elb.mode.ep.conflict	一个ELB只允许在一个企业项目下配置WAF防护	不允许重复绑定
500	WAF.00010001	internal.error	服务器内部错误	联系技术支持
500	WAF.00010002	system.busy	系统繁忙，请稍后重试	联系技术支持
500	WAF.00010003	cname.failed	创建或者修改CNAME失败	联系技术支持
500	WAF.00010004	obs.failed	获取OBS文件下载链接失败	联系技术支持

状态码	错误码	错误信息	描述	处理措施
500	WAF.00010007	risk.action.is.blocking	您的账号可能存在风险,已根据《华为云用户协议》限制您购买部分按需计费的云服务资源。如有疑问,请联系客服咨询处理	请联系客服咨询处理
500	WAF.00010008	frozen.deposit.failed	您的账户余额不足,无法开通资源,请立即充值	请充值
500	WAF.00010009	list.eps.failed	获取企业项目失败	无
500	WAF.00020001	internal.error	服务内部异常	建议过五分钟后重试
500	WAF.00020002	system.busy	系统繁忙	建议过五分钟后重试

A.3 获取项目 ID

调用 API 获取项目 ID

项目ID可以通过调用[查询指定条件下的项目信息](#)API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects”，其中{Endpoint}为IAM的终端节点，可以从[地区和终端节点](#)获取。接口的认证鉴权请参见[认证鉴权](#)。

响应示例如下，其中projects下的“id”即为项目ID。

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "xxxxxxx",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
      },
      "id": "a4a5d4098fb4474fa22cd05f897d6b99",
      "enabled": true
    }
  ],
  "links": {
    "next": null,
    "previous": null,
  }
}
```

```

    "self": "https://www.example.com/v3/projects"
  }
}

```

从控制台获取项目 ID

在调用接口的时候，部分URL中需要填入项目编号，所以需要获取到项目编号。项目编号获取步骤如下：

1. 登录管理控制台。
2. 单击用户名，在下拉列表中单击“我的凭证”。
3. 在“API凭证”页面的项目列表中查看项目ID。

图 A-1 查看项目 ID



A.4 地理位置编码

地理位置	地理位置编码	地理位置	地理位置编码	地理位置	地理位置编码
中国	CN	智利	Chile	立陶宛	Lithuania
安徽	AH	圣诞岛	Christmas Island	卢森堡	Luxembourg
北京	BJ	哥伦比亚	Colombia	北马其顿（前南斯拉夫马其顿共和国）	Macedonia (FYROM)
重庆	CQ	科摩罗	Comoros	马达加斯加	Madagascar
福建	FJ	刚果	Congo	马拉维	Malawi
广东	GD	刚果民主共和国	Congo, Democratic Republic	马来西亚	Malaysia
甘肃	GS	库克群岛	Cook Islands	马尔代夫	Maldives
广西	GX	哥斯达黎加	Costa Rica	马里	Mali

地理位置	地理位置编码	地理位置	地理位置编码	地理位置	地理位置编码
贵州	GZ	科特迪瓦	Cote D'Ivoire (Ivory Coast)	马耳他	Malta
河南	HA	克罗地亚	Croatia	马绍尔群岛	Marshall Islands
湖北	HB	津巴布韦	Zimbabwe	马提尼克岛	Martinique
河北	HE	库拉奥	Curaao	毛里塔尼亚	Mauritania
海南	HI	塞浦路斯	Cyprus	毛里求斯	Mauritius
中国-香港	HK	捷克共和国	Czech Republic	马约特	Mayotte
黑龙江	HL	德国	DE	墨西哥	Mexico
湖南	HN	丹麦	Denmark	密克罗尼西亚	Micronesia
吉林	JL	吉布提	Djibouti	摩尔多瓦	Moldova
江苏	JS	多米尼克	Dominica	摩纳哥	Monaco
江西	JX	多米尼加共和国	Dominican Republic	蒙古	Mongolia
辽宁	LN	东帝汶	East Timor	黑山	Montenegro
中国-澳门	MO	厄瓜多尔	Ecuador	蒙特塞拉特	Montserrat
内蒙古	NM	埃及	Egypt	摩洛哥	Morocco
宁夏	NX	萨尔瓦多	El Salvador	莫桑比克	Mozambique
青海	QH	赤道几内亚	Equatorial Guinea	缅甸	Myanmar
四川	SC	厄立特里亚	Eritrea	纳米比亚	Namibia
山东	SD	爱沙尼亚	Estonia	瑙鲁	Nauru
上海	SH	埃塞俄比亚	Ethiopia	尼泊尔	Nepal
陕西	SN	福克兰群岛	Falkland Islands	新喀里多尼亚	New Caledonia
山西	SX	法罗群岛	Faroe Islands	新西兰	New Zealand
天津	TJ	斐济	Fiji	尼加拉瓜	Nicaragua

地理位置	地理位置编码	地理位置	地理位置编码	地理位置	地理位置编码
中国-台湾	TW	芬兰	Finland	尼日尔	Niger
新疆	XJ	法国	FR	尼日利亚	Nigeria
中国-西藏	XZ	法属圭亚那	French Guiana	纽埃	Niue
云南	YN	法属玻利尼西亚	French Polynesia	诺福克岛	Norfolk Island
浙江	ZJ	加蓬	Gabon	阿拉伯联合酋长国	United Arab Emirates
阿富汗	Afghanista	冈比亚	Gambia	北马里亚纳群岛	Northern Mariana Islands
阿赫韦南马	Ahvenanmaa	格鲁吉亚	Georgia	挪威	Norway
阿尔巴尼亚	Albania	加纳	Ghana	阿曼	Oman
阿尔及利亚	Algeria	直布罗陀	Gibraltar	巴基斯坦	Pakistan
美属萨摩亚	American Samoa	希腊	Greece	帕劳	Palau
安道尔	Andorra	格陵兰	Greenland	巴勒斯坦	Palestine
安哥拉	Angola	格林纳达	Grenada	巴拿马	Panama
安圭拉	Anguilla	瓜德罗普岛	Guadeloupe	巴布亚新几内亚	Papua New Guinea
安提瓜和巴布达	Antigua and Barbuda	关岛	Guam	巴拉圭	Paraguay
阿根廷	Argentina	危地马拉	Guatemala	秘鲁	Peru
亚美尼亚	Armenia	根西	Guernsey	菲律宾	Philippines
阿鲁巴岛	Aruba	几内亚	Guinea	波兰	Poland
澳大利亚	AU	几内亚比绍	Guinea-Bissau	葡萄牙	Portugal
奥地利	Austria	圭亚那	Guyana	波多黎各	Puerto Rico
阿塞拜疆	Azerbaijan	海地	Haiti	卡塔尔	Qatar
巴哈马	Bahamas	洪都拉斯	Honduras	留尼旺	Reunion
巴林	Bahrain	匈牙利	Hungary	罗马尼亚	Romania

地理位置	地理位置编码	地理位置	地理位置编码	地理位置	地理位置编码
孟加拉国	Bangladesh	冰岛	Iceland	赞比亚	Zambia
巴巴多斯	Barbados	印度	IN	卢旺达	Rwanda
白俄罗斯	Belarus	印度尼西亚	Indonesia	圣巴莱米	Saint Barthelemy
比利时	Belgium	沃利斯和富图纳	Wallis and Futuna	圣基茨和尼维斯	Saint Kitts and Nevis
伯利兹	Belize	伊拉克	Iraq	圣卢西亚	Saint Lucia
贝宁	Benin	爱尔兰	Ireland	圣马丁	Saint Martin
百慕大	Bermuda	英属马恩岛	Isle of Man, British Dependancy	圣文森特和格林纳丁斯	Saint Vincent and the Grenadines
不丹	Bhutan	也门	Yemen	圣皮埃尔和密克隆	Saint-Pierre and Miquelon
玻利维亚	Bolivia	意大利	Italy	萨摩亚	Samoa
波斯尼亚和黑塞哥维那	Bosnia and Herzegovina	牙买加	Jamaica	圣马力诺	San Marino
博茨瓦纳	Botswana	泽西岛	Jersey	圣多美和普林西比	Sao Tome and Principe
布韦群岛	Bouvet Island	约旦	Jordan	沙特阿拉伯	Saudi Arabia
巴西	BR	日本	JP	塞内加尔	Senegal
英属印度洋领地	British Indian Ocean Territory	哈萨克斯坦	Kazakhstan	塞尔维亚	Serbia
英属维尔京群岛	British Virgin Islands	肯尼亚	Kenya	塞舌尔	Seychelles
文莱达鲁萨兰国	Brunei Darussalam	基里巴斯	Kiribati	塞拉利昂	Sierra Leone
保加利亚	Bulgaria	科威特	Kuwait	新加坡	Singapore

地理位置	地理位置编码	地理位置	地理位置编码	地理位置	地理位置编码
布基纳法索	Burkina Faso	吉尔吉斯斯坦	Kyrgyzstan	圣马丁	Sint Maarten
布隆迪	Burundi	老挝	Laos	斯洛伐克	Slovakia
加拿大	CA	拉脱维亚	Latvia	斯洛文尼亚	Slovenia
柬埔寨	Cambodia	黎巴嫩	Lebanon	所罗门群岛	Solomon Islands
喀麦隆	Cameroon	莱索托	Lesotho	索马里	Somalia
佛得角	Cape Verde	利比里亚	Liberia	南非	South Africa
荷兰加勒比	Caribbean Netherlands	利比亚	Libya	南乔治亚和南桑威奇群岛	South Georgia and The South Sandwich Islands
开曼群岛	Cayman Islands	列支敦士登	Liechtenstein	南苏丹	South Sudan
中非共和国	Central African Republic	苏里南	Suriname	西班牙	Spain
乍得	Chad	泰国	Thailand	斯里兰卡	Sri Lanka
斯威士兰	Eswatini	荷兰	The Netherlands	汤加	Tonga
瑞典	Sweden	韩国	The Republic of Korea	特立尼达和多巴哥	Trinidad and Tobago
瑞士	Switzerland	多哥	Togo	突尼斯	Tunisia
塔吉克斯坦	Tajikistan	托克劳	Tokelau	土库曼斯坦	Turkmenistan
坦桑尼亚	Tanzania	美属维尔京群岛	United States Virgin Islands	特克斯和凯科斯群岛	Turks and Caicos Islands
图瓦卢	Tuvalu	乌拉圭	Uruguay	梵蒂冈	Vatican City
乌干达	Uganda	美国	US	委内瑞拉	Venezuela

地理位置	地理位置编 码	地理位置	地理位置编 码	地理位置	地理位置编 码
英国	UK	乌兹别克斯 坦	Uzbekistan	越南	Viet Nam