

DDoS 防护

API 文档



腾讯云

【版权声明】

©2013–2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【商标声明】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。

您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

API 文档

更新历史

简介

API 概览

调用方式

请求结构

公共参数

签名方法 v3

签名方法

返回结果

参数类型

高防IP资产实例相关接口

绑定高防弹性公网IP

绑定高防弹性公网IP到Clib

解绑高防弹性公网IP

资产列表相关接口

绑定IP到高防包实例

查询与证书ID匹配的七层规则

获取高防IP资产实例列表

获取高防包资产实例列表

获取转发监听器列表

高防IP获取7层规则

高防IP获取7层转发规则

DDoS攻击防护策略相关接口

修改DDoSIP防护等级

修改DDoS清洗阈值

防护配置相关接口

新建CC四层黑白名单

新建CC防护的地域封禁配置

设置DDoS防护的AI防护开关

添加DDoS防护的IP网段黑白名单

配置DDoS连接抑制选项

添加DDoS防护的区域封禁配置

添加DDoS防护的访问限速配置

添加DDoS防护的特征过滤规则

添加DDoS防护的端口acl策略

批量添加DDoS防护的端口acl策略

设置DDoS防护的协议封禁配置

添加DDoS防护的水印防护配置

添加DDoS防护的水印防护密钥

删除CC四层黑白名单

删除CC防护的区域封禁配置

删除DDoS防护的IP网段黑白名单

删除DDoS防护的区域封禁配置

删除DDoS防护的访问限速配置

删除DDoS防护的特征过滤规则

删除DDoS防护的端口acl策略

删除DDoS防护的水印防护配置

删除DDoS防护的水印防护密钥

获取CC四层黑白名单列表

获取CC防护的区域封禁配置列表

获取DDoS防护的IP网段黑白名单
获取DDoS连接抑制配置列表
获取DDoS防护的IP黑白名单列表
获取DDoS防护的AI防护开关列表
获取DDoS防护的区域封禁配置列表
获取DDoS防护的访问限速配置列表
获取DDoS防护的特征过滤规则列表
获取DDoS防护的端口acl策略列表
获取DDoS防护的协议封禁配置列表
获取DDoS防护的水印防护配置列表
修改CC四层黑白名单
修改DDoS黑白名单列表
修改DDoS防护的区域封禁配置
修改DDoS防护的访问限速配置
修改DDoS防护的特征过滤规则
修改DDoS防护的端口acl策略
开启或关闭DDoS防护的水印防护配置

告警通知相关接口

设置单IP默认告警阈值配置
设置单IP告警阈值配置
获取单IP默认告警阈值配置
获取单IP告警阈值配置列表

接入配置相关接口

批量配置L7转发规则的证书
添加7层转发规则
获取L7转发规则健康检查异常结果列表
修改7层转发规则

智能调度相关接口

创建调度的域名
获取智能调度域名列表
修改智能解析域名名称

黑洞解封相关接口

获取IP封堵列表

统计报表相关接口

获取高防包流量折线图
获取业务流量状态码统计列表
获取高防IP业务监控流量曲线
获取高防IP业务流量曲线
获取CC攻击指标曲线
获取DDoS攻击指标曲线
拉取防护概览攻击趋势
获取防护概览趋势图
获取防护概览的ddos攻击事件
获取防护概览DDoS攻击流量带宽和攻击包速率数据
拉取防护概览指标
查询账号维度待处理风险信息

其他接口

新增CC精准防护策略
新增CC频率限制策略
删除CC分级策略
删除CC精准防护策略
删除CC频率限制策略
删除CC清洗阈值策略
获取基础防护攻击状态

- 获取CC防护等级列表
- 获取CC分级策略
- 获取CC精准防护列表
- 获取CC频率限制策略列表
- 获取CC清洗阈值列表
- 修改CC的防护等级
- 修改CC精准防护策略
- 修改CC频率限制策略
- 修改CC清洗阈值

- 数据结构

- 错误码

- 附录

 - RegionId列表

API 文档

更新历史

最近更新时间：2025-03-24 01:08:13

第 70 次发布

发布时间：2025-03-24 01:07:53

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DescribeBlackWhitelplist
- DescribeListProtectThresholdConfig

删除数据结构：

- ListenerCcThresholdConfig
- ProtectThresholdRelation

修改数据结构：

- [RuleInstanceRelation](#)
 - 修改成员：EipList, InstanceId, Cname

第 69 次发布

发布时间：2025-02-21 01:07:28

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstanceSpecification](#)
 - 修改成员：DownGradeProtect
- [NewL7RuleEntry](#)
 - 新增成员：CCThresholdNew

第 68 次发布

发布时间：2024-08-22 01:03:05

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeBGPIPL7Rules](#)
 - 新增入参：Source

第 67 次发布

发布时间：2024-06-04 01:07:52

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterZoneIdList

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：ZoneId

第 66 次发布

发布时间：2024-04-19 01:07:51

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterTransRegionFlag

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：TransRegionFlag

第 65 次发布

发布时间：2024-04-12 01:07:18

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：SuperPackFlag

第 64 次发布

发布时间：2024-01-19 01:06:05

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBGPIPL7Rules](#)

新增数据结构：

- [BGPIPL7RuleEntry](#)

第 63 次发布

发布时间：2024-01-12 01:06:10

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIstances](#)
 - 新增入参: FilterPlanCntFlag

修改数据结构:

- [BGPIstance](#)
 - 新增成员: PlanCntFlag

第 62 次发布

发布时间: 2023-12-13 01:06:45

本次发布包含了以下内容:

改善已有的文档。

新增接口:

- [DescribeIpBlockList](#)

新增数据结构:

- [IpBlockData](#)

第 61 次发布

发布时间: 2023-11-28 01:06:04

本次发布包含了以下内容:

改善已有的文档。

修改接口:

- [DescribeBasicDeviceStatus](#)
 - 新增入参: CnameWafIdList
 - 新增出参: CnameWafData

第 60 次发布

发布时间: 2023-10-10 01:05:53

本次发布包含了以下内容:

改善已有的文档。

修改数据结构:

- [BGPIstance](#)
 - 新增成员: FreeServiceBandwidth
 - 修改成员: GiftServiceBandWidth, ModifyTime, BasicPlusFlag

第 59 次发布

发布时间: 2023-09-19 01:05:58

本次发布包含了以下内容:

改善已有的文档。

预下线接口:

- [DescribeBlackWhitelPList](#)
- [DescribeListProtectThresholdConfig](#)

第 58 次发布

发布时间：2023-07-26 01:01:23

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBizMonitorTrend](#)

第 57 次发布

发布时间：2023-06-27 01:01:43

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIstances](#)
 - 新增入参：FilterBasicPlusFlag

修改数据结构：

- [BGPIstance](#)
 - 新增成员：BasicPlusFlag

第 56 次发布

发布时间：2023-05-19 01:02:07

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateBoundIP](#)
 - 新增入参：FilterRegion
- [DescribeBasicDeviceStatus](#)
 - 新增入参：IdList, FilterRegion
 - 修改入参：IpList
 - 新增出参：CLBData
- [DescribeNewL7Rules](#)
 - 新增入参：Export

修改数据结构：

- [BoundIpInfo](#)
 - 新增成员：Domain

第 55 次发布

发布时间：2023-04-21 01:01:52

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeBizTrend](#)
 - 新增入参：BusinessType

- 新增出参: MaxData
- [DescribeListBGPIstances](#)
 - 新增入参: FilterAssetIpList
- [DescribeListSchedulingDomain](#)
 - 新增入参: Status
- [DescribeOverviewCCTrend](#)
 - 修改入参: Business
- [DescribeOverviewDDoSSTrend](#)
 - 修改入参: Business

修改数据结构:

- [BGPIstance](#)
 - 新增成员: ModifyTime
- [EipProductInfo](#)
 - 新增成员: Domain
- [IPLineInfo](#)
 - 新增成员: Domain

第 54 次发布

发布时间: 2023-03-10 01:01:32

本次发布包含了以下内容:

改善已有的文档。

修改数据结构:

- [BGPIstanceSpecification](#)
 - 新增成员: DownGradeProtect

第 53 次发布

发布时间: 2023-02-22 01:02:37

本次发布包含了以下内容:

改善已有的文档。

修改数据结构:

- [BGPIstance](#)
 - 新增成员: GiftServiceBandWidth

第 52 次发布

发布时间: 2023-02-10 01:02:06

本次发布包含了以下内容:

改善已有的文档。

修改数据结构:

- [Layer7Rule](#)
 - 新增成员: Protocol, Vport
- [SourceServer](#)
 - 新增成员: Port

第 51 次发布

发布时间：2023-01-13 01:02:31

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIstances](#)
 - 新增入参：ExcludeAdvancedInfo

修改数据结构：

- [L7RuleHealth](#)
 - 新增成员：PassiveStatus

第 50 次发布

发布时间：2023-01-06 01:02:00

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIPIstance](#)
 - 新增成员：ElasticBandwidth, EOFlag
- [NewL7RuleEntry](#)
 - 新增成员：Version

第 49 次发布

发布时间：2022-12-20 11:47:08

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBizHttpStatus](#)

新增数据结构：

- [HttpStatusMap](#)

第 48 次发布

发布时间：2022-12-16 01:05:43

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [SwitchWaterPrintConfig](#)
 - 新增入参：CloudSdkProxy

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：Line, ElasticServiceBandwidth
- [WaterPrintConfig](#)
 - 新增成员：CloudSdkProxy

第 47 次发布

发布时间：2022-12-12 01:06:27

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIPIstances](#)
 - 新增入参：FilterConvoy
- [DescribeListBGPIstances](#)
 - 新增入参：FilterTrialFlag, FilterConvoy

修改数据结构：

- [BGPIPIstance](#)
 - 新增成员：ConvoyId

第 46 次发布

发布时间：2022-11-23 06:03:44

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyDDoSThreshold](#)
 - 新增入参：AckFloodThreshold, AckFloodPktThreshold, SynAckFloodThreshold, SynAckFloodPktThreshold, RstFloodThreshold, RstFloodPktThreshold

修改数据结构：

- [ProtectThresholdRelation](#)
 - 新增成员：AckFloodThreshold, AckFloodPktThreshold, SynAckFloodThreshold, SynAckFloodPktThreshold, RstFloodThreshold, RstFloodPktThreshold

第 45 次发布

发布时间：2022-11-09 06:08:09

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyDDoSThreshold](#)
 - 新增入参：OtherThresholdFlag, SynFloodThreshold, SynFloodPktThreshold, UdpFloodThreshold, UdpFloodPktThreshold

修改数据结构：

- [PacketFilterConfig](#)
 - 新增成员：PktLenGT
- [ProtectThresholdRelation](#)
 - 新增成员：SynFloodThreshold, SynFloodPktThreshold, UdpFloodThreshold, UdpFloodPktThreshold
- [ProtocolBlockConfig](#)
 - 新增成员：PingOfDeath, TearDrop

第 44 次发布

发布时间：2022-11-01 06:06:58

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribePendingRiskInfo](#)

第 43 次发布

发布时间：2022-10-26 06:07:49

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIstances](#)
 - 新增入参：FilterLightFlag, FilterChannelFlag

第 42 次发布

发布时间：2022-09-12 06:07:03

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIstance](#)
 - 新增成员：VitalityVersion

第 41 次发布

发布时间：2022-09-06 06:07:26

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- [ModifyL7RulesEdge](#)

第 40 次发布

发布时间：2022-08-29 13:11:21

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeNewL7RulesErrHealth](#)

第 39 次发布

发布时间：2022-08-09 10:41:42

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBgpBizTrend](#)

删除接口：

- DeleteBlackWhitelpList

第 38 次发布

发布时间：2022-07-27 06:03:19

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIPIInstance](#)
 - 新增成员：InstanceVersion
- [L4RuleSource](#)
 - 新增成员：Backup
- [L7RuleHealth](#)
 - 新增成员：ProtocolFlag, PassiveEnable, BlockInter, FailedCountInter, FailedThreshold, PassiveStatusCode

第 37 次发布

发布时间：2022-07-15 06:03:07

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [L7RuleEntry](#)
 - 新增成员：CCAIEnable

第 36 次发布

发布时间：2022-07-11 11:21:01

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateNewL7Rules](#)

第 35 次发布

发布时间：2022-07-11 06:05:26

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeNewL7Rules](#)

新增数据结构：

- [L7RuleHealth](#)

第 34 次发布

发布时间：2022-07-01 06:05:20

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterPackType

新增数据结构：

- [AnycastOutPackRelation](#)

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：AnycastOutPackRelation
- [BGPIInstance](#)
 - 新增成员：IpCountNewFlag

第 33 次发布

发布时间：2022-06-16 06:00:42

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateCCReqLimitPolicy](#)
 - 新增入参：IsGlobal
- [DescribeListBGPIInstances](#)
 - 新增入参：FilterTag
- [DescribeListBGPIInstances](#)
 - 新增入参：FilterTag

新增数据结构：

- [TagFilter](#)
- [TagInfo](#)

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：TagInfoList
- [BGPIInstance](#)
 - 新增成员：TagInfoList
- [BlackWhitelPRelation](#)
 - 新增成员：ModifyTime
- [PacketFilterRelation](#)
 - 新增成员：ModifyTime

第 32 次发布

发布时间：2022-05-17 06:03:35

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterEnterpriseFlag

第 31 次发布

发布时间：2022-05-12 06:05:03

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeOverviewDDoSEventList](#)

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterCname, FilterInstanceIdList
- [DescribeListIPAlarmConfig](#)
 - 新增入参：FilterCname

新增数据结构：

- [OverviewDDoSEvent](#)

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：BGPIPChannelFlag
- [BGPIInstanceSpecification](#)
 - 新增成员：EnterpriseFlag, ElasticLimit
- [IPLineInfo](#)
 - 新增成员：Cname, ResourceFlag

第 30 次发布

发布时间：2022-05-11 06:03:57

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- [RuleInstanceRelation](#)

修改数据结构：

- [Layer4Rule](#)
 - 新增成员：InstanceDetailRule
- [Layer7Rule](#)
 - 新增成员：InstanceDetailRule

第 29 次发布

发布时间：2022-05-06 06:04:11

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeOverviewAttackTrend](#)
- [DescribeOverviewCCTrend](#)
- [DescribeOverviewDDoSSTrend](#)
- [DescribeOverviewIndex](#)

第 28 次发布

发布时间：2022-04-26 06:04:03

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstanceSpecification](#)
 - 新增成员：BattleEditionFlag, ChannelEditionFlag

第 27 次发布

发布时间：2022-04-18 06:05:37

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [NewL7RuleEntry](#)
 - 新增成员：RewriteHttps, ErrCode

第 26 次发布

发布时间：2022-04-07 06:04:07

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateSchedulingDomain](#)
 - 新增入参：Product

第 25 次发布

发布时间：2022-03-24 23:15:57

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterInstanceldList

修改数据结构：

- [BGPIInstanceSpecification](#)
 - 新增成员：ServiceBandWidth

第 24 次发布

发布时间：2022-03-15 08:03:22

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstanceSpecification](#)
 - 新增成员：UnionPackFlag

第 23 次发布

发布时间：2022-03-02 08:04:36

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCcBlackWhitelList](#)
- [DeleteCCLevelPolicy](#)
- [DeleteCCThresholdPolicy](#)
- [DescribeCCLevelList](#)
- [DescribeCCLevelPolicy](#)
- [DescribeCCThresholdList](#)

新增数据结构：

- [CCLevelPolicy](#)
- [CCThresholdPolicy](#)

第 22 次发布

发布时间：2022-02-28 08:05:20

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [AssociateDDoSEipLoadBalancer](#)
 - 新增入参：Vip

第 21 次发布

发布时间：2022-02-14 08:04:59

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCCPrecisionPolicy](#)
- [CreateCCReqLimitPolicy](#)
- [CreateCcGeolPBlockConfig](#)
- [DeleteCCPrecisionPolicy](#)
- [DeleteCCRequestLimitPolicy](#)
- [DeleteCcBlackWhitelList](#)
- [DeleteCcGeolPBlockConfig](#)

- [DeletePortAclConfig](#)
- [DescribeCCPrecisionPlyList](#)
- [DescribeCCReqLimitPolicyList](#)
- [DescribeCcBlackWhitelPList](#)
- [DescribeCcGeolPBlockConfigList](#)
- [ModifyCCLevelPolicy](#)
- [ModifyCCPrecisionPolicy](#)
- [ModifyCCReqLimitPolicy](#)
- [ModifyCCThresholdPolicy](#)
- [ModifyCcBlackWhitelPList](#)
- [ModifyDDoSBlackWhitelPList](#)
- [ModifyDDoSLevel](#)
- [ModifyDDoSThreshold](#)
- [ModifyPortAclConfig](#)

新增数据结构：

- [CCPrecisionPlyRecord](#)
- [CCPrecisionPolicy](#)
- [CCReqLimitPolicy](#)
- [CCReqLimitPolicyRecord](#)
- [CcBlackWhitelPPolicy](#)
- [CcGeolPBlockConfig](#)
- [CcGeolPPolicyNew](#)

第 20 次发布

发布时间：2022-01-19 08:04:47

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeDDoSConnectLimitList](#)
- [DescribeListPortAclList](#)

新增数据结构：

- [AclConfigRelation](#)
- [ConnectLimitRelation](#)

第 19 次发布

发布时间：2022-01-18 08:03:27

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateDDoSConnectLimit](#)

新增数据结构：

- [ConnectLimitConfig](#)

第 18 次发布

发布时间：2022-01-17 08:02:24

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreatePortAclConfig](#)
- [CreatePortAclConfigList](#)

新增数据结构：

- [AclConfig](#)

第 17 次发布

发布时间：2021-12-20 17:09:09

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateDDoSBlackWhitelist](#)
- [DeleteDDoSBlackWhitelist](#)
- [DescribeDDoSBlackWhitelist](#)

新增数据结构：

- [IpSegment](#)

第 16 次发布

发布时间：2021-12-15 08:02:37

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterStatus
- [DescribeListBGPInstances](#)
 - 新增入参：FilterStatus, FilterBoundStatus

第 15 次发布

发布时间：2021-12-02 08:02:10

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：V6Flag

第 14 次发布

发布时间：2021-11-23 08:02:33

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [ModifyNewDomainRules](#)

新增数据结构：

- [NewL7RuleEntry](#)

第 13 次发布

发布时间：2021-11-22 08:01:25

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeListBGPIInstances](#)
 - 新增入参：FilterDamDDoSStatus

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：DamDDoSStatus

第 12 次发布

发布时间：2021-08-31 08:00:10

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [L4RuleSource](#)
 - 新增成员：Port

第 11 次发布

发布时间：2021-08-18 08:00:08

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [L7RuleEntry](#)
 - 新增成员：Id

第 10 次发布

发布时间：2021-08-12 08:00:06

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBizTrend](#)
- [DescribeCCTrend](#)

- [DescribeDDoS Trend](#)

新增数据结构：

- [ProtocolPort](#)

第 9 次发布

发布时间：2021-08-11 08:00:07

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIPIstance](#)
 - 新增成员：Domain

第 8 次发布

发布时间：2021-08-06 08:00:07

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [AssociateDDoS EipLoadBalancer](#)

第 7 次发布

发布时间：2021-08-05 08:00:06

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [ModifyL7RulesEdge](#)

新增数据结构：

- [L4RuleSource](#)
- [L7RuleEntry](#)

第 6 次发布

发布时间：2021-07-28 08:00:07

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeBasicDeviceStatus](#)

新增数据结构：

- [KeyValue](#)

第 5 次发布

发布时间：2021-07-21 08:00:04

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [SwitchWaterPrintConfig](#)

修改数据结构：

- [ForwardListener](#)
 - 新增成员：FrontendPortEnd
- [WaterPrintConfig](#)
 - 新增成员：Verify

第 4 次发布

发布时间：2021-06-24 08:00:04

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateBoundIP](#)

修改接口：

- [CreateL7RuleCerts](#)
 - 新增入参：CertId, L7Rules
- [DescribeL7RulesBySSLCertId](#)
 - 新增入参：Status, CertIds

新增数据结构：

- [BoundIpInfo](#)

第 3 次发布

发布时间：2021-06-23 11:26:33

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateL7RuleCerts](#)
- [DescribeL7RulesBySSLCertId](#)

新增数据结构：

- [CertIdInsL7Rules](#)
- [InsL7Rules](#)
- [SuccessCode](#)

第 2 次发布

发布时间：2021-06-22 08:00:04

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [BGPIInstance](#)
 - 新增成员：DDoSLevel, CCEnable

第 1 次发布

发布时间：2021-06-18 10:36:41

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [AssociateDDoSEipAddress](#)
- [CreateBlackWhitelPList](#)
- [CreateDDoSAI](#)
- [CreateDDoSGeolPBlockConfig](#)
- [CreateDDoSSpeedLimitConfig](#)
- [CreateDefaultAlarmThreshold](#)
- [CreateIPAlarmThresholdConfig](#)
- [CreatePacketFilterConfig](#)
- [CreateProtocolBlockConfig](#)
- [CreateSchedulingDomain](#)
- [CreateWaterPrintConfig](#)
- [CreateWaterPrintKey](#)
- [DeleteBlackWhitelPList](#)
- [DeleteDDoSGeolPBlockConfig](#)
- [DeleteDDoSSpeedLimitConfig](#)
- [DeletePacketFilterConfig](#)
- [DeleteWaterPrintConfig](#)
- [DeleteWaterPrintKey](#)
- [DescribeBlackWhitelPList](#)
- [DescribeDefaultAlarmThreshold](#)
- [DescribeListBGPIInstances](#)
- [DescribeListBGPIInstances](#)
- [DescribeListBlackWhitelPList](#)
- [DescribeListDDoSAI](#)
- [DescribeListDDoSGeolPBlockConfig](#)
- [DescribeListDDoSSpeedLimitConfig](#)
- [DescribeListIPAlarmConfig](#)
- [DescribeListListener](#)
- [DescribeListPacketFilterConfig](#)
- [DescribeListProtectThresholdConfig](#)
- [DescribeListProtocolBlockConfig](#)
- [DescribeListSchedulingDomain](#)
- [DescribeListWaterPrintConfig](#)
- [DisassociateDDoSEipAddress](#)
- [ModifyDDoSGeolPBlockConfig](#)
- [ModifyDDoSSpeedLimitConfig](#)
- [ModifyDomainUsrName](#)
- [ModifyPacketFilterConfig](#)

新增数据结构：

- [BGPIPInstance](#)
- [BGPIPInstanceSpecification](#)
- [BGPIPInstanceUsages](#)
- [BGPIInstance](#)
- [BGPIInstanceSpecification](#)
- [BGPIInstanceUsages](#)
- [BlackWhitelPRelation](#)
- [DDoSAIRelation](#)
- [DDoSGeoIPBlockConfig](#)
- [DDoSGeoIPBlockConfigRelation](#)
- [DDoSSpeedLimitConfig](#)
- [DDoSSpeedLimitConfigRelation](#)
- [DefaultAlarmThreshold](#)
- [EipAddressPackRelation](#)
- [EipAddressRelation](#)
- [EipProductInfo](#)
- [ForwardListener](#)
- [IPAlarmThresholdRelation](#)
- [IPLineInfo](#)
- [InstanceRelation](#)
- [Layer4Rule](#)
- [Layer7Rule](#)
- [ListenerCcThresholdConfig](#)
- [PackInfo](#)
- [PacketFilterConfig](#)
- [PacketFilterRelation](#)
- [PortSegment](#)
- [ProtectThresholdRelation](#)
- [ProtocolBlockConfig](#)
- [ProtocolBlockRelation](#)
- [ProxyTypeInfo](#)
- [RegionInfo](#)
- [SchedulingDomainInfo](#)
- [SourceServer](#)
- [SpeedValue](#)
- [StaticPackRelation](#)
- [WaterPrintConfig](#)
- [WaterPrintKey](#)
- [WaterPrintRelation](#)

简介

最近更新时间：2025-04-25 01:08:05

DDoS 高防包（Anti-DDoS Pro）是为业务部署在腾讯云上的用户提升 DDoS 防护能力的付费服务。对云产品直接生效，无需更换 IP，购买后只需要绑定需要防护的 IP 即可使用，具备接入便捷、0变更等特点。

API 概览

最近更新时间：2025-03-24 01:08:13

高防IP资产实例相关接口

接口名称	接口功能	频率限制（次/秒）
AssociateDDoSIPAddress	绑定高防弹性公网IP	20
AssociateDDoSIPLoadBalancer	绑定高防弹性公网IP到CLB	20
DisassociateDDoSIPAddress	解绑高防弹性公网IP	20

资产列表相关接口

接口名称	接口功能	频率限制（次/秒）
CreateBoundIP	绑定IP到高防包实例	20
DescribeL7RulesBySSLCertId	查询与证书ID匹配的七层规则	20
DescribeListBGPIPInstances	获取高防IP资产实例列表	20
DescribeListBGPInstances	获取高防包资产实例列表	20
DescribeListListener	获取转发监听器列表	20
DescribeBGPIPL7Rules	高防IP获取7层转发规则	20
DescribeNewL7Rules	高防IP获取7层规则	20

DDoS攻击防护策略相关接口

接口名称	接口功能	频率限制（次/秒）
ModifyDDoSLevel	修改DDoSIP防护等级	20
ModifyDDoSThreshold	修改DDoS清洗阈值	20

防护配置相关接口

接口名称	接口功能	频率限制（次/秒）
CreateBlackWhitelist	添加DDoS防护的IP黑白名单	20
CreateCcBlackWhitelist	新建CC四层黑白名单	20
CreateCcGeoIPBlockConfig	新建CC防护的地域封禁配置	20
CreateDDoSAl	设置DDoS防护的AI防护开关	20
CreateDDoSBlackWhitelist	添加DDoS防护的IP网段黑白名单	20
CreateDDoSConnectLimit	配置DDoS连接抑制选项	20
CreateDDoSGeoIPBlockConfig	添加DDoS防护的区域封禁配置	20
CreateDDoSSpeedLimitConfig	添加DDoS防护的访问限速配置	20
CreatePacketFilterConfig	添加DDoS防护的特征过滤规则	20
CreatePortAclConfig	添加DDoS防护的端口acl策略	20

接口名称	接口功能	频率限制（次/秒）
CreatePortAclConfigList	批量添加DDoS防护的端口acl策略	20
CreateProtocolBlockConfig	设置DDoS防护的协议封禁配置	20
CreateWaterPrintConfig	添加DDoS防护的水印防护配置	20
CreateWaterPrintKey	添加DDoS防护的水印防护密钥	20
DeleteCcBlackWhitelList	删除CC四层黑白名单	20
DeleteCcGeoIPBlockConfig	删除CC防护的区域封禁配置	20
DeleteDDoSBlackWhitelList	删除DDoS防护的IP网段黑白名单	20
DeleteDDoSGeoIPBlockConfig	删除DDoS防护的区域封禁配置	20
DeleteDDoSSpeedLimitConfig	删除DDoS防护的访问限速配置	20
DeletePacketFilterConfig	删除DDoS防护的特征过滤规则	20
DeletePortAclConfig	删除DDoS防护的端口acl策略	20
DeleteWaterPrintConfig	删除DDoS防护的水印防护配置	20
DeleteWaterPrintKey	删除DDoS防护的水印防护密钥	20
DescribeCcBlackWhitelList	获取CC四层黑白名单列表	20
DescribeCcGeoIPBlockConfigList	获取CC防护的区域封禁配置列表	20
DescribeDDoSBlackWhitelList	获取DDoS防护的IP网段黑白名单	20
DescribeDDoSConnectLimitList	获取DDoS连接抑制配置列表	20
DescribeListBlackWhitelList	获取DDoS防护的IP黑白名单列表	20
DescribeListDDoSAI	获取DDoS防护的AI防护开关列表	20
DescribeListDDoSGeoIPBlockConfig	获取DDoS防护的区域封禁配置列表	20
DescribeListDDoSSpeedLimitConfig	获取DDoS防护的访问限速配置列表	20
DescribeListPacketFilterConfig	获取DDoS防护的特征过滤规则列表	20
DescribeListPortAclList	获取DDoS防护的端口acl策略列表	20
DescribeListProtocolBlockConfig	获取DDoS防护的协议封禁配置列表	20
DescribeListWaterPrintConfig	获取DDoS防护的水印防护配置列表	20
ModifyCcBlackWhitelList	修改CC四层黑白名单	20
ModifyDDoSBlackWhitelList	修改DDoS黑白名单列表	20
ModifyDDoSGeoIPBlockConfig	修改DDoS防护的区域封禁配置	20
ModifyDDoSSpeedLimitConfig	修改DDoS防护的访问限速配置	20
ModifyPacketFilterConfig	修改DDoS防护的特征过滤规则	20
ModifyPortAclConfig	修改DDoS防护的端口acl策略	20
SwitchWaterPrintConfig	开启或关闭DDoS防护的水印防护配置	20

其他接口

接口名称	接口功能	频率限制（次/秒）
CreateCCPrecisionPolicy	新增CC精准防护策略	20
CreateCCReqLimitPolicy	新增CC频率限制策略	20
DeleteCCLevelPolicy	删除CC分级策略	20
DeleteCCPrecisionPolicy	删除CC精准防护策略	20
DeleteCCRequestLimitPolicy	删除CC频率限制策略	20
DeleteCCThresholdPolicy	删除CC清洗阈值策略	20
DescribeBasicDeviceStatus	获取基础防护攻击状态	20
DescribeCCLevelList	获取CC防护等级列表	20
DescribeCCLevelPolicy	获取CC分级策略	20
DescribeCCPrecisionPlyList	获取CC精准防护列表	20
DescribeCCReqLimitPolicyList	获取CC频率限制策略列表	20
DescribeCCThresholdList	获取CC清洗阈值列表	20
ModifyCCLevelPolicy	修改CC的防护等级	20
ModifyCCPrecisionPolicy	修改CC精准防护策略	20
ModifyCCReqLimitPolicy	修改CC频率限制策略	20
ModifyCCThresholdPolicy	修改CC清洗阈值	20

告警通知相关接口

接口名称	接口功能	频率限制（次/秒）
CreateDefaultAlarmThreshold	设置单IP默认告警阈值配置	20
CreateIPAlarmThresholdConfig	设置单IP告警阈值配置	20
DescribeDefaultAlarmThreshold	获取单IP默认告警阈值配置	20
DescribeListIPAlarmConfig	获取单IP告警阈值配置列表	20

接入配置相关接口

接口名称	接口功能	频率限制（次/秒）
CreateL7RuleCerts	批量配置L7转发规则的证书	20
CreateNewL7Rules	添加7层转发规则	20
DescribeNewL7RulesErrHealth	获取L7转发规则健康检查异常结果列表	20
ModifyNewDomainRules	修改7层转发规则	20

智能调度相关接口

接口名称	接口功能	频率限制（次/秒）
CreateSchedulingDomain	创建调度的域名	20
DescribeListSchedulingDomain	获取智能调度域名列表	20

接口名称	接口功能	频率限制（次/秒）
ModifyDomainUsrName	修改智能解析域名名称	20

黑洞解封相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeIpBlockList	获取IP封堵列表	20

统计报表相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeBgpBizTrend	获取高防包流量折线图	20
DescribeBizHttpStatus	获取业务流量状态码统计列表	20
DescribeBizMonitorTrend	获取高防IP业务监控流量曲线	20
DescribeBizTrend	获取高防IP业务流量曲线	20
DescribeCCTrend	获取CC攻击指标曲线	20
DescribeDDoSAttackTrend	获取DDoS攻击指标曲线	20
DescribeOverviewAttackTrend	拉取防护概览攻击趋势	20
DescribeOverviewCCTrend	获取防护概览趋势图	20
DescribeOverviewDDoSAttackEventList	获取防护概览的ddos攻击事件	20
DescribeOverviewDDoSAttackTrend	获取防护概览DDoS攻击流量带宽和攻击包速率数据	20
DescribeOverviewIndex	拉取防护概览指标	20
DescribePendingRiskInfo	查询账号维度待处理风险信息	20

注意：

以上给出的接口频率限制维度为 API + 接入地域 + 子账号，有关限频更多说明参考：[API 频率限制说明](#)

调用方式

请求结构

最近更新时间：2025-05-21 01:07:55

1. 服务地址

API 支持就近地域接入，本产品就近地域接入域名为 `antiddos.tencentcloudapi.com`，也支持指定地域域名访问，例如广州地域的域名为 `antiddos.ap-guangzhou.tencentcloudapi.com`。

推荐使用就近地域接入域名。根据调用接口时客户端所在位置，会自动解析到最近的某个具体地域的服务器。例如在广州发起请求，会自动解析到广州的服务器，效果和指定 `antiddos.ap-guangzhou.tencentcloudapi.com` 是一致的。

注意：对时延敏感的业务，建议指定带地域的域名。

注意：域名是 API 的接入点，并不代表产品或者接口实际提供服务的地域。产品支持的地域列表请在调用方式/公共参数文档中查阅，接口支持的地域请在接口文档输入参数中查阅。

目前支持的域名列表为：

接入地域	域名
就近地域接入（推荐，只支持非金融区）	antiddos.tencentcloudapi.com
华南地区（广州）	antiddos.ap-guangzhou.tencentcloudapi.com
华东地区（上海）	antiddos.ap-shanghai.tencentcloudapi.com
华东地区（南京）	antiddos.ap-nanjing.tencentcloudapi.com
华北地区（北京）	antiddos.ap-beijing.tencentcloudapi.com
西南地区（成都）	antiddos.ap-chengdu.tencentcloudapi.com
西南地区（重庆）	antiddos.ap-chongqing.tencentcloudapi.com
港澳台地区（中国香港）	antiddos.ap-hongkong.tencentcloudapi.com
亚太东南（新加坡）	antiddos.ap-singapore.tencentcloudapi.com
亚太东南（雅加达）	antiddos.ap-jakarta.tencentcloudapi.com
亚太东南（曼谷）	antiddos.ap-bangkok.tencentcloudapi.com
亚太东北（首尔）	antiddos.ap-seoul.tencentcloudapi.com
亚太东北（东京）	antiddos.ap-tokyo.tencentcloudapi.com
美国东部（弗吉尼亚）	antiddos.na-ashburn.tencentcloudapi.com
美国西部（硅谷）	antiddos.na-siliconvalley.tencentcloudapi.com
南美地区（圣保罗）	antiddos.sa-saopaulo.tencentcloudapi.com
欧洲地区（法兰克福）	antiddos.eu-frankfurt.tencentcloudapi.com

注意：由于金融区和非金融区是隔离不互通的，因此当访问金融区服务时（公共参数 `Region` 为金融区地域），需要同时指定带金融区地域的域名，最好和 `Region` 的地域保持一致。

金融区接入地域	金融区域名
华东地区（上海金融）	antiddos.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	antiddos.ap-shenzhen-fsi.tencentcloudapi.com

2. 通信协议

腾讯云 API 的所有接口均通过 HTTPS 进行通信，提供高安全性的通信通道。

3. 请求方法

支持的 HTTP 请求方法:

- POST (推荐)
- GET

POST 请求支持的 Content-Type 类型:

- application/json (推荐)，必须使用签名方法 v3 (TC3-HMAC-SHA256)。
- application/x-www-form-urlencoded，必须使用签名方法 v1 (HmacSHA1 或 HmacSHA256)。
- multipart/form-data (仅部分接口支持)，必须使用签名方法 v3 (TC3-HMAC-SHA256)。

GET 请求的请求包大小不得超过32KB。POST 请求使用签名方法 v1 (HmacSHA1、HmacSHA256) 时不得超过1MB。POST 请求使用签名方法 v3 (TC3-HMAC-SHA256) 时支持10MB。

4. 字符编码

均使用 UTF-8 编码。

公共参数

最近更新时间：2024-11-15 01:08:10

公共参数是用于标识用户和接口签名的参数，如非必要，在每个接口单独的文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

公共参数的具体内容会因您使用的签名方法版本不同而有所差异。

使用签名方法 v3 的公共参数

签名方法 v3（有时也称作 TC3-HMAC-SHA256）相比签名方法 v1（有些文档可能会简称签名方法），更安全，支持更大的请求包，支持 POST JSON 格式，性能有一定提升，推荐使用该签名方法计算签名。完整介绍详见 [签名方法 v3](#)。

注意：出于简化的目的，部分接口文档中的示例使用的是签名方法 v1 GET 请求，而不是更安全的签名方法 v3。

使用签名方法 v3 时，公共参数需要统一放到 HTTP Header 请求头部中，如下表所示：

参数名称	类型	必选	描述
Action	String	是	HTTP 请求头：X-TC-Action。操作的接口名称。取值参考接口文档输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	-	HTTP 请求头：X-TC-Region。地域参数，用来标识希望操作哪个地域的数据。取值参考接口文档中输入参数章节关于公共参数 Region 的说明。 注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。
Timestamp	Integer	是	HTTP 请求头：X-TC-Timestamp。当前 UNIX 时间戳，可记录发起 API 请求的时间。例如 1529223702。 注意：如果与服务器时间相差超过5分钟，会引起签名过期错误。
Version	String	是	HTTP 请求头：X-TC-Version。操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
Authorization	String	是	HTTP 标准身份认证头部字段，例如： TC3-HMAC-SHA256 Credential=AKID***/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 其中， - TC3-HMAC-SHA256：签名方法，目前固定取该值； - Credential：签名凭证，AKID*** 是 SecretId；Date 是 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为具体产品名，通常为域名前缀。例如，域名 cvm.tencentcloudapi.com 意味着产品名是 cvm。本产品取值为 antiddos；tc3_request 为固定字符串； - SignedHeaders：参与签名计算的头部信息，content-type 和 host 为必选头部； - Signature：签名摘要，计算过程详见 文档 。
Token	String	否	HTTP 请求头：X-TC-Token。即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	HTTP 请求头：X-TC-Language。指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表中的前十个，接口参数设置为偏移量 Offset=0，返回数量 Limit=10，则其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
```

```
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: ap-guangzhou
```

HTTP POST (application/json) 请求结构示例:

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-05-30/cvm/tc3_request, SignedHe
aders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: application/json
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

{"Offset":0,"Limit":10}
```

HTTP POST (multipart/form-data) 请求结构示例 (仅特定的接口支持):

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-05-30/cvm/tc3_request, SignedHe
aders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: multipart/form-data; boundary=58731222010402
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

--58731222010402
Content-Disposition: form-data; name="Offset"

0
--58731222010402
Content-Disposition: form-data; name="Limit"

10
--58731222010402--
```

使用签名方法 v1 的公共参数

使用签名方法 v1 (有时会称作 HmacSHA256 和 HmacSHA1), 公共参数需要统一放到请求串中, 完整介绍详见[文档](#)

参数名称	类型	必选	描述
Action	String	是	操作的接口名称。取值参考接口文档中输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口, 取值为 DescribeInstances。
Region	String	-	地域参数, 用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。 注意: 某些接口不需要传递该参数, 接口文档中会对此特别说明, 此时即使传递该参数也不会生效。

参数名称	类型	必选	描述
Timestamp	Integer	是	当前 UNIX 时间戳，可记录发起 API 请求的时间。例如1529223702，如果与当前时间相差过大，会引起签名过期错误。
Nonce	Integer	是	随机正整数，与 Timestamp 联合起来，用于防止重放攻击。
SecretId	String	是	在 云API密钥 上申请的标识身份的 SecretId，一个 SecretId 对应唯一的 SecretKey，而 SecretKey 会用来生成请求签名 Signature。
Signature	String	是	请求签名，用来验证此次请求的合法性，需要用户根据实际的输入参数计算得出。具体计算方法参见 文档 。
Version	String	是	操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
SignatureMethod	String	否	签名方式，目前支持 HmacSHA256 和 HmacSHA1。只有指定此参数为 HmacSHA256 时，才使用 HmacSHA256 算法验证签名，其他情况均使用 HmacSHA1 验证签名。
Token	String	否	即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表，其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbe224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****

Host: cvm.tencentcloudapi.com
```

HTTP POST 请求结构示例：

```
https://cvm.tencentcloudapi.com/

Host: cvm.tencentcloudapi.com
Content-Type: application/x-www-form-urlencoded

Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbe224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****
```

地域列表

本产品所有接口 Region 字段的可选值如下表所示。如果接口不支持该表中的所有地域，则会在接口文档中单独说明。

地域	取值
华南地区（广州）	ap-guangzhou

签名方法 v3

最近更新时间：2024-12-25 01:09:16

以下文档说明了签名方法 v3 的签名过程，但仅在您编写自己的代码来调用腾讯云 API 时才有用。我们推荐您使用 [腾讯云 API Explorer](#)，[腾讯云 SDK](#) 和 [腾讯云命令行工具（TCCLI）](#) 等开发者工具，从而无需学习如何对 API 请求进行签名。

推荐使用 API Explorer

</> 点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个请求进行身份验证，用户需要使用安全凭证，经过特定的步骤对请求进行签名（Signature），每个请求都需要在公共参数中指定该签名结果并以指定的方式和格式发送请求。

为什么要进行签名

签名通过以下方式帮助保护请求：

1. 验证请求者的身份

签名确保请求是由持有有效访问密钥的人发送的。请参阅控制台 [云 API 密钥](#) 页面获取密钥相关信息。

2. 保护传输中的数据

为了防止请求在传输过程中被篡改，腾讯云 API 会使用请求参数来计算请求的哈希值，并将生成的哈希值加密后作为请求的一部分，发送到腾讯云 API 服务器。服务器会使用收到的请求参数以同样的过程计算哈希值，并验证请求中的哈希值。如果请求被篡改，将导致哈希值不一致，腾讯云 API 将拒绝本次请求。

签名方法 v3（TC3-HMAC-SHA256）功能上覆盖了以前的签名方法 v1，而且更安全，支持更大的请求，支持 JSON 格式，POST 请求支持传空数组和空字符串，性能有一定提升，推荐使用该签名方法计算签名。

首次接触，建议使用 [API Explorer](#) 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v3”，可以对生成签名过程进行验证，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 [Python](#)、[Java](#)、[PHP](#)、[Go](#)、[NodeJS](#)、[.NET](#)、[C++](#)、[Ruby](#)。

申请安全凭证

本文使用的安全凭证为密钥，密钥包括 SecretId 和 SecretKey。每个用户最多可以拥有两对密钥。

- SecretId：用于标识 API 调用者身份，可以简单类比为用户名。
- SecretKey：用于验证 API 调用者的身份，可以简单类比为密码。
- 用户必须严格保管安全凭证，避免泄露，否则将危及财产安全。如已泄露，请立刻禁用该安全凭证。

申请安全凭证的具体步骤如下：

- 登录 [腾讯云管理中心控制台](#)。
- 前往 [云API密钥](#) 的控制台页面。
- 在 [云API密钥](#) 页面，单击【新建密钥】创建一对密钥。

签名版本 v3 签名过程

云 API 支持 GET 和 POST 请求。对于 GET 方法，只支持 Content-Type: application/x-www-form-urlencoded 协议格式。对于 POST 方法，目前支持 Content-Type: application/json 以及 Content-Type: multipart/form-data 两种协议格式，json 格式绝大多数接口均支持，multipart 格式只有特定接口支持，此时该接口不能使用 json 格式调用，参考具体业务接口文档说明。推荐使用 POST 请求，因为两者的结果并无差异，但 GET 请求只支持 32 KB 以内的请求包。

下面以云服务器查询广州实例列表作为例子，分步骤介绍签名的计算过程。我们选择该接口是因为：

- 云服务器默认已开通，该接口很常用；
- 该接口是只读的，不会改变现有资源的状态；
- 接口覆盖的参数种类较全，可以演示包含数据结构的数组如何使用。

在示例中，不论公共参数或者接口的参数，我们尽量选择容易犯错的情况。在实际调用接口时，请根据实际情况来，每个接口的参数并不相同，不要照抄这个例子的参数和值。此外，这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数（在 HTTP 头部设置，添加 X-TC- 前缀）。

假设用户的 SecretId 和 SecretKey 分别是：AKID***** 和 *****。用户想查看广州云服务器名为“未命名”的主机状态，只返回一条数据。则请求可能为：

```
curl -X POST https://cvm.tencentcloudapi.com \
-H "Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f" \
-H "Content-Type: application/json; charset=utf-8" \
-H "Host: cvm.tencentcloudapi.com" \
-H "X-TC-Action: DescribeInstances" \
-H "X-TC-Timestamp: 1551113065" \
-H "X-TC-Version: 2017-03-12" \
-H "X-TC-Region: ap-guangzhou" \
-d '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
```

下面详细解释签名计算过程。

1. 拼接规范请求串

按如下伪代码格式拼接规范请求串（CanonicalRequest）：

```
CanonicalRequest =
HTTPRequestMethod + '\n' +
CanonicalURI + '\n' +
CanonicalQueryString + '\n' +
CanonicalHeaders + '\n' +
SignedHeaders + '\n' +
HashedRequestPayload
```

字段名称	解释
HTTPRequestMethod	HTTP 请求方法（GET、POST）。此示例取值为 POST。
CanonicalURI	URI 参数，API 3.0 固定为正斜杠（/）。
CanonicalQueryString	发起 HTTP 请求 URL 中的查询字符串，对于 POST 请求，固定为空字符串""，对于 GET 请求，则为 URL 中间号（？）后面的字符串内容，例如：Limit=10&Offset=0。 注意：CanonicalQueryString 需要参考 RFC3986 进行 URLEncode 编码（特殊字符编码后需大写字母），字符集 UTF-8。推荐使用编程语言标准库进行编码。
CanonicalHeaders	参与签名的头部信息，至少包含 host 和 content-type 两个头部，也可加入其他头部参与签名以提高自身请求的唯一性和安全性，此示例额外增加了接口名头部。 拼接规则： 1. 头部 key 和 value 统一转成小写，并去掉首尾空格，按照 key:value\n 格式拼接； 2. 多个头部，按照头部 key（小写）的 ASCII 升序进行拼接。 此示例计算结果是 content-type:application/json; charset=utf-8\nhost:cvm.tencentcloudapi.com\nx-tc-action:describeinstances\n。 注意：content-type 必须和实际发送的相符合，有些编程语言网络库即使未指定也会自动添加 charset 值，如果签名时和发送时不一致，服务器会返回签名校验失败。
SignedHeaders	参与签名的头部信息，说明此次请求有哪些头部参与了签名，和 CanonicalHeaders 包含的头部内容是一一对应的。 content-type 和 host 为必选头部。 拼接规则：

字段名称	解释
	1. 头部 key 统一转成小写； 2. 多个头部 key (小写) 按照 ASCII 升序进行拼接，并且以分号 (;) 分隔。 此示例为 content-type;host;x-tc-action
HashedRequestPayload	请求正文 (payload, 即 body, 此示例为 {"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}) 的哈希值, 计算伪代码为 Lowercase(HexEncode(Hash.SHA256(RequestPayload)))，即对 HTTP 请求正文做 SHA256 哈希，然后十六进制编码，最后编码串转换成小写字母。对于 GET 请求，RequestPayload 固定为空字符串。此示例计算结果是 35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064。

根据以上规则，示例中得到的规范请求串如下：

```
POST
/

content-type:application/json; charset=utf-8
host:cvm.tencentcloudapi.com
x-tc-action:describeinstances

content-type;host;x-tc-action
35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064
```

2. 拼接待签名字符串

按如下格式拼接待签名字符串：

```
StringToSign =
Algorithm + "\n" +
RequestTimestamp + "\n" +
CredentialScope + "\n" +
HashedCanonicalRequest
```

字段名称	解释
Algorithm	签名算法，目前固定为 TC3-HMAC-SHA256。
RequestTimestamp	请求时间戳，即请求头部的公共参数 X-TC-Timestamp 取值，取当前时间 UNIX 时间戳，精确到秒。此示例取值为 1551113065。
CredentialScope	凭证范围，格式为 Date/service/tc3_request，包含日期、所请求的服务和终止字符串 (tc3_request)。Date 为 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为产品名称，必须与调用的产品域名一致。此示例计算结果是 2019-02-25/cvm/tc3_request。
HashedCanonicalRequest	前述步骤拼接所得规范请求串的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(CanonicalRequest)))。此示例计算结果是 7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84。

注意：

1. Date 必须从时间戳 X-TC-Timestamp 计算得到，且时区为 UTC+0。如果加入系统本地时区信息，例如东八区，将导致白天和晚上调用成功，但是凌晨时调用必定失败。假设时间戳为 1551113065，在东八区的时间是 2019-02-26 00:44:25，但是计算得到的 Date 取 UTC+0 的日期应为 2019-02-25，而不是 2019-02-26。

2. Timestamp 必须是当前系统时间，且需确保系统时间和标准时间是同步的，如果相差超过五分钟则必定失败。如果长时间不和标准时间同步，可能运行一段时间后，请求失败，返回签名过期错误。

根据以上规则，示例中得到的待签名字符串如下：

```
TC3-HMAC-SHA256
1551113065
2019-02-25/cvm/tc3_request
7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84
```

3. 计算签名

1) 计算派生签名密钥，伪代码如下：

```
SecretKey = "*****"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

派生出的密钥 SecretDate、SecretService 和 SecretSigning 是二进制的数，可能包含不可打印字符，将其转为十六进制字符串打印的输出分别为：
da98fb70dcf6b112dc21038d1eeeb3a95c74b4dcb12c1131f864f6066bd02be0，
8d70cbefb03939f929db64d32dc2ba89b1095620119fe3e050e2b18c5bd2752f，
b596b923aad85185e2d1f6659d2a062e0a86731226e021e61bfe06f7ed05f5af。

请注意，不同的编程语言，HMAC 库函数中参数顺序可能不一样，请以实际情况为准。此处的伪代码密钥参数 key 在前，消息参数 data 在后。通常标准库函数会提供二进制格式的返回值，也可能会提供打印友好的十六进制格式的返回值，此处使用的是二进制格式。

字段名称	解释
SecretKey	原始的 SecretKey，即 *****。
Date	即 Credential 中的 Date 字段信息。此示例取值为 2019-02-25。
Service	即 Credential 中的 Service 字段信息。此示例取值为 cvm。

2) 计算签名，伪代码如下：

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

此示例计算结果是 10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f。

4. 拼接 Authorization

按如下格式拼接 Authorization：

```
Authorization =
Algorithm + ' ' +
'Credential=' + SecretId + '/' + CredentialScope + ', ' +
'SignedHeaders=' + SignedHeaders + ', ' +
'Signature=' + Signature
```

字段名称	解释
Algorithm	签名方法，固定为 TC3-HMAC-SHA256。

字段名称	解释
SecretId	密钥对中的 SecretId，即 AKID*****。
CredentialScope	见上文，凭证范围。此示例计算结果是 2019-02-25/cvm/tc3_request。
SignedHeaders	见上文，参与签名的头部信息。此示例取值为 content-type;host;x-tc-action。
Signature	签名值。此示例计算结果是 10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f。

根据以上规则，示例中得到的值为：

```
TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
```

最终完整的调用信息如下：

```
POST https://cvm.tencentcloudapi.com/
Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
Content-Type: application/json; charset=utf-8
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1551113065
X-TC-Region: ap-guangzhou

{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}
```

⚠ 注意：

请求发送时的 HTTP 头部（Header）和请求体（Payload）必须和签名计算过程中的内容完全一致，否则会返回签名不一致错误。可以通过打印实际请求内容，网络抓包等方式对比排查。

签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 [SDK 中心](#)。当前支持的编程语言有：

- [Python](#)
- [Java](#)
- [PHP](#)
- [Go](#)
- [NodeJS](#)
- [.NET](#)
- [C++](#)
- [Ruby](#)

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- [Signature Demo](#)

为了更清楚地解释签名过程，下面以实际编程语言为例，将上述的签名过程完整实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

Java

```
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPITC3Demo {
    private final static Charset UTF8 = StandardCharsets.UTF_8;
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    private final static String SECRET_ID = System.getenv("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    private final static String SECRET_KEY = System.getenv("TENCENTCLOUD_SECRET_KEY");
    private final static String CT_JSON = "application/json; charset=utf-8";

    public static byte[] hmac256(byte[] key, String msg) throws Exception {
        Mac mac = Mac.getInstance("HmacSHA256");
        SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
        mac.init(secretKeySpec);
        return mac.doFinal(msg.getBytes(UTF8));
    }

    public static String sha256Hex(String s) throws Exception {
        MessageDigest md = MessageDigest.getInstance("SHA-256");
        byte[] d = md.digest(s.getBytes(UTF8));
        return DatatypeConverter.printHexBinary(d).toLowerCase();
    }

    public static void main(String[] args) throws Exception {
        String service = "cvm";
        String host = "cvm.tencentcloudapi.com";
        String region = "ap-guangzhou";
        String action = "DescribeInstances";
        String version = "2017-03-12";
        String algorithm = "TC3-HMAC-SHA256";
        String timestamp = "1551113065";
        //String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
        // 注意时区, 否则容易出错
        sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
        String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

        // ***** 步骤 1: 拼接规范请求串 *****
        String httpRequestMethod = "POST";
        String canonicalUri = "/";
        String canonicalQueryString = "";
        String canonicalHeaders = "content-type:application/json; charset=utf-8\n"
            + "host:" + host + "\n" + "x-tc-action:" + action.toLowerCase() + "\n";
```

```
String signedHeaders = "content-type;host;x-tc-action";

String payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}";
String hashedRequestPayload = sha256Hex(payload);
String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryString + "\n"
+ canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
System.out.println(canonicalRequest);

// ***** 步骤 2: 拼接待签名字符串 *****
String credentialScope = date + "/" + service + "/" + "tc3_request";
String hashedCanonicalRequest = sha256Hex(canonicalRequest);
String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
System.out.println(stringToSign);

// ***** 步骤 3: 计算签名 *****
byte[] secretDate = hmac256(("TC3" + SECRET_KEY).getBytes(UTF8), date);
byte[] secretService = hmac256(secretDate, service);
byte[] secretSigning = hmac256(secretService, "tc3_request");
String signature = DatatypeConverter.printHexBinary(hmac256(secretSigning, stringToSign)).toLowerCase();
System.out.println(signature);

// ***** 步骤 4: 拼接 Authorization *****
String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
System.out.println(authorization);

TreeMap<String, String> headers = new TreeMap<String, String>();
headers.put("Authorization", authorization);
headers.put("Content-Type", CT_JSON);
headers.put("Host", host);
headers.put("X-TC-Action", action);
headers.put("X-TC-Timestamp", timestamp);
headers.put("X-TC-Version", version);
headers.put("X-TC-Region", region);

StringBuilder sb = new StringBuilder();
sb.append("curl -X POST https://").append(host)
.append(" -H \"Authorization: ").append(authorization).append("\"")
.append(" -H \"Content-Type: application/json; charset=utf-8\"")
.append(" -H \"Host: ").append(host).append("\"")
.append(" -H \"X-TC-Action: ").append(action).append("\"")
.append(" -H \"X-TC-Timestamp: ").append(timestamp).append("\"")
.append(" -H \"X-TC-Version: ").append(version).append("\"")
.append(" -H \"X-TC-Region: ").append(region).append("\"")
.append(" -d '").append(payload).append("'");
System.out.println(sb.toString());
}
}
```

Python

```
# -*- coding: utf-8 -*-
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

service = "cvm"
host = "cvm.tencentcloudapi.com"
endpoint = "https://" + host
region = "ap-guangzhou"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
#timestamp = int(time.time())
timestamp = 1551113065
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 1, "Filters": [{"Values": [u"未命名"], "Name": "instance-name"}]}

# ***** 步骤 1: 拼接规范请求串 *****
http_request_method = "POST"
canonical_uri = "/"
canonical_querystring = ""
ct = "application/json; charset=utf-8"
payload = json.dumps(params)
canonical_headers = "content-type:%s\nhost:%s\nx-tc-action:%s\n" % (ct, host, action.lower())
signed_headers = "content-type;host;x-tc-action"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
canonical_uri + "\n" +
canonical_querystring + "\n" +
canonical_headers + "\n" +
signed_headers + "\n" +
hashed_request_payload)
print(canonical_request)

# ***** 步骤 2: 拼接待签名字符串 *****
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
str(timestamp) + "\n" +
credential_scope + "\n" +
hashed_canonical_request)
print(string_to_sign)

# ***** 步骤 3: 计算签名 *****
# 计算签名摘要函数
def sign(key, msg):
return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
```

```
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ***** 步骤 4: 拼接 Authorization *****
authorization = (algorithm + " " +
"Credential=" + secret_id + "/" + credential_scope + ", " +
"SignedHeaders=" + signed_headers + ", " +
"Signature=" + signature)
print(authorization)

print('curl -X POST ' + endpoint
+ ' -H "Authorization: ' + authorization + '"
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + host + '"
+ ' -H "X-TC-Action: ' + action + '"
+ ' -H "X-TC-Timestamp: ' + str(timestamp) + '"
+ ' -H "X-TC-Version: ' + version + '"
+ ' -H "X-TC-Region: ' + region + '"
+ " -d '" + payload + '"')
```

Golang

```
package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/hex"
    "fmt"
    "os"
    "strings"
    "time"
)

func sha256hex(s string) string {
    b := sha256.Sum256([]byte(s))
    return hex.EncodeToString(b[:])
}

func hmacsha256(s, key string) string {
    hashed := hmac.New(sha256.New, []byte(key))
    hashed.Write([]byte(s))
    return string(hashed.Sum(nil))
}

func main() {
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
```

```
host := "cvm.tencentcloudapi.com"
algorithm := "TC3-HMAC-SHA256"
service := "cvm"
version := "2017-03-12"
action := "DescribeInstances"
region := "ap-guangzhou"
//var timestamp int64 = time.Now().Unix()
var timestamp int64 = 1551113065

// step 1: build canonical request string
httpRequestMethod := "POST"
canonicalURI := "/"
canonicalQueryString := ""
canonicalHeaders := fmt.Sprintf("content-type:%s\nhost:%s\nx-tc-action:%s\n",
    "application/json; charset=utf-8", host, strings.ToLower(action))
signedHeaders := "content-type;host;x-tc-action"
payload := `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`
hashedRequestPayload := sha256hex(payload)
canonicalRequest := fmt.Sprintf("%s\n%s\n%s\n%s\n%s\n%s",
    httpRequestMethod,
    canonicalURI,
    canonicalQueryString,
    canonicalHeaders,
    signedHeaders,
    hashedRequestPayload)
fmt.Println(canonicalRequest)

// step 2: build string to sign
date := time.Unix(timestamp, 0).UTC().Format("2006-01-02")
credentialScope := fmt.Sprintf("%s/%s/tc3_request", date, service)
hashedCanonicalRequest := sha256hex(canonicalRequest)
string2sign := fmt.Sprintf("%s\n%d\n%s\n%s",
    algorithm,
    timestamp,
    credentialScope,
    hashedCanonicalRequest)
fmt.Println(string2sign)

// step 3: sign string
secretDate := hmacsha256(date, "TC3"+secretKey)
secretService := hmacsha256(service, secretDate)
secretSigning := hmacsha256("tc3_request", secretService)
signature := hex.EncodeToString([]byte(hmacsha256(string2sign, secretSigning)))
fmt.Println(signature)

// step 4: build authorization
authorization := fmt.Sprintf("%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
    algorithm,
    secretId,
    credentialScope,
    signedHeaders,
    signature)
fmt.Println(authorization)
```

```
curl := fmt.Sprintf(`curl -X POST https://%s\
-H "Authorization: %s"\
-H "Content-Type: application/json; charset=utf-8"\
-H "Host: %s" -H "X-TC-Action: %s"\
-H "X-TC-Timestamp: %d"\
-H "X-TC-Version: %s"\
-H "X-TC-Region: %s"\
-d '%s'`, host, authorization, host, action, timestamp, version, region, payload)

fmt.Println(curl)
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$host = "cvm.tencentcloudapi.com";
$service = "cvm";
$version = "2017-03-12";
$action = "DescribeInstances";
$region = "ap-guangzhou";
// $timestamp = time();
$timestamp = 1551113065;
$algorithm = "TC3-HMAC-SHA256";

// step 1: build canonical request string
$httpRequestMethod = "POST";
$canonicalUri = "/";
$canonicalQueryString = "";
$canonicalHeaders = implode("\n", [
    "content-type:application/json; charset=utf-8",
    "host:".$host,
    "x-tc-action:".strtolower($action),
    ""
]);
$signedHeaders = implode(";", [
    "content-type",
    "host",
    "x-tc-action",
]);
$payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}';
$hashedRequestPayload = hash("SHA256", $payload);
$canonicalRequest = $httpRequestMethod."\n"
.$canonicalUri."\n"
.$canonicalQueryString."\n"
.$canonicalHeaders."\n"
.$signedHeaders."\n"
.$hashedRequestPayload;
echo $canonicalRequest.PHP_EOL;

// step 2: build string to sign
```

```
$date = gmdate("Y-m-d", $timestamp);
$credentialScope = $date."/".$service."/tc3_request";
$shashedCanonicalRequest = hash("SHA256", $canonicalRequest);
$stringToSign = $algorithm."\n"
.$timestamp."\n"
.$credentialScope."\n"
.$shashedCanonicalRequest;
echo $stringToSign.PHP_EOL;

// step 3: sign string
$secretDate = hash_hmac("SHA256", $date, "TC3".$secretKey, true);
$secretService = hash_hmac("SHA256", $service, $secretDate, true);
$secretSigning = hash_hmac("SHA256", "tc3_request", $secretService, true);
$signature = hash_hmac("SHA256", $stringToSign, $secretSigning);
echo $signature.PHP_EOL;

// step 4: build authorization
$authorization = $algorithm
." Credential=".$secretId."/".$credentialScope
.", SignedHeaders=".$signedHeaders.", Signature=".$signature;
echo $authorization.PHP_EOL;

$curl = "curl -X POST https://".$host
.' -H "Authorization: '.$authorization.'"
.' -H "Content-Type: application/json; charset=utf-8"
.' -H "Host: '.$host.'"
.' -H "X-TC-Action: '.$action.'"
.' -H "X-TC-Timestamp: '.$timestamp.'"
.' -H "X-TC-Version: '.$version.'"
.' -H "X-TC-Region: '.$region.'"
." -d '$payload'";
echo $curl.PHP_EOL;
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'digest'
require 'json'
require 'time'
require 'openssl'

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

service = 'cvm'
host = 'cvm.tencentcloudapi.com'
endpoint = 'https://' + host
region = 'ap-guangzhou'
action = 'DescribeInstances'
```

```
version = '2017-03-12'
algorithm = 'TC3-HMAC-SHA256'
# timestamp = Time.now.to_i
timestamp = 1551113065
date = Time.at(timestamp).utc.strftime('%Y-%m-%d')

# ***** 步骤 1: 拼接规范请求串 *****
http_request_method = 'POST'
canonical_uri = '/'
canonical_querystring = ''
canonical_headers = "content-type:application/json; charset=utf-8\nhost:#{host}\nx-tc-action:#{action.downcase}\n"
signed_headers = 'content-type;host;x-tc-action'
# params = { 'Limit' => 1, 'Filters' => [{ 'Name' => 'instance-name', 'Values' => ['未命名'] }] }
# payload = JSON.generate(params, { 'ascii_only' => true, 'space' => ' ' })
# json will generate in random order, to get specified result in example, we hard-code it here.
payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
hashed_request_payload = Digest::SHA256.hexdigest(payload)
canonical_request = [
  http_request_method,
  canonical_uri,
  canonical_querystring,
  canonical_headers,
  signed_headers,
  hashed_request_payload,
].join("\n")

puts canonical_request

# ***** 步骤 2: 拼接待签名字符串 *****
credential_scope = date + '/' + service + '/' + 'tc3_request'
hashed_request_payload = Digest::SHA256.hexdigest(canonical_request)
string_to_sign = [
  algorithm,
  timestamp.to_s,
  credential_scope,
  hashed_request_payload,
].join("\n")
puts string_to_sign

# ***** 步骤 3: 计算签名 *****
digest = OpenSSL::Digest.new('sha256')
secret_date = OpenSSL::HMAC.digest(digest, 'TC3' + secret_key, date)
secret_service = OpenSSL::HMAC.digest(digest, secret_date, service)
secret_signing = OpenSSL::HMAC.digest(digest, secret_service, 'tc3_request')
signature = OpenSSL::HMAC.hexdigest(digest, secret_signing, string_to_sign)
puts signature

# ***** 步骤 4: 拼接 Authorization *****
authorization = "#{algorithm} Credential=#{secret_id}/#{credential_scope}, SignedHeaders=#{signed_headers}, Signatu
re=#{signature}"
puts authorization

puts 'curl -X POST ' + endpoint \
+ ' -H "Authorization: ' + authorization + '" \
```

```
+ ' -H "Content-Type: application/json; charset=utf-8" ' \
+ ' -H "Host: ' + host + '" ' \
+ ' -H "X-TC-Action: ' + action + '" ' \
+ ' -H "X-TC-Timestamp: ' + timestamp.to_s + '" ' \
+ ' -H "X-TC-Version: ' + version + '" ' \
+ ' -H "X-TC-Region: ' + region + '" ' \
+ " -d '" + payload + '"
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Security.Cryptography;
using System.Text;

public class Application
{
    public static string SHA256Hex(string s)
    {
        using (SHA256 algo = SHA256.Create())
        {
            byte[] hashbytes = algo.ComputeHash(Encoding.UTF8.GetBytes(s));
            StringBuilder builder = new StringBuilder();
            for (int i = 0; i < hashbytes.Length; ++i)
            {
                builder.Append(hashbytes[i].ToString("x2"));
            }
            return builder.ToString();
        }
    }

    public static byte[] HmacSHA256(byte[] key, byte[] msg)
    {
        using (HMACSHA256 mac = new HMACSHA256(key))
        {
            return mac.ComputeHash(msg);
        }
    }

    public static Dictionary<String, String> BuildHeaders(string secretid,
        string secretkey, string service, string endpoint, string region,
        string action, string version, DateTime date, string requestPayload)
    {
        string datestr = date.ToString("yyyy-MM-dd");
        DateTime startTime = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc);
        long requestTimestamp = (long)Math.Round((date - startTime).TotalMilliseconds, MidpointRounding.AwayFromZero) / 1000;
        // ***** 步骤 1: 拼接规范请求串 *****
        string algorithm = "TC3-HMAC-SHA256";
        string httpRequestMethod = "POST";
        string canonicalUri = "/";
        string canonicalQueryString = "";
        string contentType = "application/json";
```

```
string canonicalHeaders = "content-type:" + contentType + "; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.ToLower() + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string hashedRequestPayload = SHA256Hex(requestPayload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
Console.WriteLine(canonicalRequest);

// ***** 步骤 2: 拼接待签名字符串 *****
string credentialScope = datestr + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = SHA256Hex(canonicalRequest);
string stringToSign = algorithm + "\n"
+ requestTimestamp.ToString() + "\n"
+ credentialScope + "\n"
+ hashedCanonicalRequest;
Console.WriteLine(stringToSign);

// ***** 步骤 3: 计算签名 *****
byte[] tc3SecretKey = Encoding.UTF8.GetBytes("TC3" + secretkey);
byte[] secretDate = HmacSHA256(tc3SecretKey, Encoding.UTF8.GetBytes(datestr));
byte[] secretService = HmacSHA256(secretDate, Encoding.UTF8.GetBytes(service));
byte[] secretSigning = HmacSHA256(secretService, Encoding.UTF8.GetBytes("tc3_request"));
byte[] signatureBytes = HmacSHA256(secretSigning, Encoding.UTF8.GetBytes(stringToSign));
string signature = BitConverter.ToString(signatureBytes).Replace("-", "").ToLower();
Console.WriteLine(signature);

// ***** 步骤 4: 拼接 Authorization *****
string authorization = algorithm + " "
+ "Credential=" + secretid + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", "
+ "Signature=" + signature;
Console.WriteLine(authorization);

Dictionary<string, string> headers = new Dictionary<string, string>();
headers.Add("Authorization", authorization);
headers.Add("Host", endpoint);
headers.Add("Content-Type", contentType + "; charset=utf-8");
headers.Add("X-TC-Timestamp", requestTimestamp.ToString());
headers.Add("X-TC-Version", version);
headers.Add("X-TC-Action", action);
headers.Add("X-TC-Region", region);
return headers;
}

public static void Main(string[] args)
{
    // 密钥参数
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
```

```
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";

// 此处由于示例规范的原因，采用时间戳2019-02-26 00:44:25，此参数作为示例，如果在项目中，您应当使用：
// DateTime date = DateTime.UtcNow;
// 注意时区，建议此时间统一采用UTC时间戳，否则容易出错
DateTime date = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc).AddSeconds(1551113065);
string requestPayload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]\"}";

Dictionary<string, string> headers = BuildHeaders(SECRET_ID, SECRET_KEY, service
, endpoint, region, action, version, date, requestPayload);

Console.WriteLine("POST https://cvm.tencentcloudapi.com");
foreach (KeyValuePair<string, string> kv in headers)
{
    Console.WriteLine(kv.Key + ": " + kv.Value);
}
Console.WriteLine();
Console.WriteLine(requestPayload);
}
```

NodeJS

```
const crypto = require('crypto');

function sha256(message, secret = '', encoding) {
    const hmac = crypto.createHmac('sha256', secret)
    return hmac.update(message).digest(encoding)
}

function getHash(message, encoding = 'hex') {
    const hash = crypto.createHash('sha256')
    return hash.update(message).digest(encoding)
}

function getDate(timestamp) {
    const date = new Date(timestamp * 1000)
    const year = date.getUTCFullYear()
    const month = ('0' + (date.getUTCMonth() + 1)).slice(-2)
    const day = ('0' + date.getUTCDate()).slice(-2)
    return `${year}-${month}-${day}`
}

function main(){
    // 密钥参数
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID*****
```

```
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const service = "cvm"
const region = "ap-guangzhou"
const action = "DescribeInstances"
const version = "2017-03-12"
//const timestamp = getTime()
const timestamp = 1551113065
//时间处理, 获取世界时间日期
const date = getDate(timestamp)

// ***** 步骤 1: 拼接规范请求串 *****
const payload = `{"Limit": 1, "Filters": [{"Values": [{"u672a\\u547d\\u540d"}], "Name": "instance-name"}]}`

const hashedRequestPayload = getHash(payload);
const httpRequestMethod = "POST"
const canonicalUri = "/"
const canonicalQueryString = ""
const canonicalHeaders = "content-type:application/json; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.toLowerCase() + "\n"
const signedHeaders = "content-type;host;x-tc-action"

const canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload
console.log(canonicalRequest)

// ***** 步骤 2: 拼接待签名字符串 *****
const algorithm = "TC3-HMAC-SHA256"
const hashedCanonicalRequest = getHash(canonicalRequest);
const credentialScope = date + "/" + service + "/" + "tc3_request"
const stringToSign = algorithm + "\n" +
timestamp + "\n" +
credentialScope + "\n" +
hashedCanonicalRequest
console.log(stringToSign)

// ***** 步骤 3: 计算签名 *****
const kDate = sha256(date, 'TC3' + SECRET_KEY)
const kService = sha256(service, kDate)
const kSigning = sha256('tc3_request', kService)
const signature = sha256(stringToSign, kSigning, 'hex')
console.log(signature)

// ***** 步骤 4: 拼接 Authorization *****
const authorization = algorithm + " " +
```

```
"Credential=" + SECRET_ID + "/" + credentialScope + ", " +
"SignedHeaders=" + signedHeaders + ", " +
"Signature=" + signature
console.log(authorization)

const curlcmd = 'curl -X POST ' + "https://" + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + endpoint + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + timestamp.toString() + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + '"'
console.log(curlcmd)
}
main()
```

C++

```
#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <iomanip>
#include <sstream>
#include <string>
#include <stdio.h>
#include <time.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

using namespace std;

string get_data(int64_t &timestamp)
{
    string utcDate;
    char buff[20] = {0};
    // time_t timenow;
    struct tm sttime;
    sttime = *gmtime(&timestamp);
    strftime(buff, sizeof(buff), "%Y-%m-%d", &sttime);
    utcDate = string(buff);
    return utcDate;
}

string int2str(int64_t n)
{
    std::stringstream ss;
    ss << n;
    return ss.str();
}

string sha256Hex(const string &str)
```

```
{
    char buf[3];
    unsigned char hash[SHA256_DIGEST_LENGTH];
    SHA256_CTX sha256;
    SHA256_Init(&sha256);
    SHA256_Update(&sha256, str.c_str(), str.size());
    SHA256_Final(hash, &sha256);
    std::string NewString = "";
    for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
    {
        sprintf(buf, sizeof(buf), "%02x", hash[i]);
        NewString = NewString + buf;
    }
    return NewString;
}

string HmacSha256(const string &key, const string &input)
{
    unsigned char hash[32];

    HMAC_CTX *h;
    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX hmac;
    HMAC_CTX_init(&hmac);
    h = &hmac;
    #else
    h = HMAC_CTX_new();
    #endif

    HMAC_Init_ex(h, &key[0], key.length(), EVP_sha256(), NULL);
    HMAC_Update(h, (unsigned char*)&input[0], input.length());
    unsigned int len = 32;
    HMAC_Final(h, hash, &len);

    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX_cleanup(h);
    #else
    HMAC_CTX_free(h);
    #endif

    std::stringstream ss;
    ss << std::setfill('0');
    for (int i = 0; i < len; i++)
    {
        ss << hash[i];
    }

    return (ss.str());
}

string HexEncode(const string &input)
{
    static const char* const lut = "0123456789abcdef";
    size_t len = input.length();
```

```
string output;
output.reserve(2 * len);
for (size_t i = 0; i < len; ++i)
{
    const unsigned char c = input[i];
    output.push_back(lut[c >> 4]);
    output.push_back(lut[c & 15]);
}
return output;
}

int main()
{
    // 密钥参数
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    string SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    string SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");

    string service = "cvm";
    string host = "cvm.tencentcloudapi.com";
    string region = "ap-guangzhou";
    string action = "DescribeInstances";
    string version = "2017-03-12";
    int64_t timestamp = 1551113065;
    string date = get_data(timestamp);

    // ***** 步骤 1: 拼接规范请求串 *****
    string httpRequestMethod = "POST";
    string canonicalUri = "/";
    string canonicalQueryString = "";
    string lower = action;
    std::transform(action.begin(), action.end(), lower.begin(), ::tolower);
    string canonicalHeaders = string("content-type:application/json; charset=utf-8\n")
        + "host:" + host + "\n"
        + "x-tc-action:" + lower + "\n";
    string signedHeaders = "content-type;host;x-tc-action";
    string payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\\\"}]}\"";
    string hashedRequestPayload = sha256Hex(payload);
    string canonicalRequest = httpRequestMethod + "\n"
        + canonicalUri + "\n"
        + canonicalQueryString + "\n"
        + canonicalHeaders + "\n"
        + signedHeaders + "\n"
        + hashedRequestPayload;
    cout << canonicalRequest << endl;

    // ***** 步骤 2: 拼接待签名字符串 *****
    string algorithm = "TC3-HMAC-SHA256";
    string RequestTimestamp = int2str(timestamp);
    string credentialScope = date + "/" + service + "/" + "tc3_request";
    string hashedCanonicalRequest = sha256Hex(canonicalRequest);
```

```
string stringToSign = algorithm + "\n" + RequestTimestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
cout << stringToSign << endl;

// ***** 步骤 3: 计算签名 *****
string kKey = "TC3" + SECRET_KEY;
string kDate = HmacSha256(kKey, date);
string kService = HmacSha256(kDate, service);
string kSigning = HmacSha256(kService, "tc3_request");
string signature = HexEncode(HmacSha256(kSigning, stringToSign));
cout << signature << endl;

// ***** 步骤 4: 拼接 Authorization *****
string authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
cout << authorization << endl;

string curlcmd = "curl -X POST https://" + host + "\n"
+ " -H \"Authorization: \" + authorization + "\"\n"
+ " -H \"Content-Type: application/json; charset=utf-8\" + "\n"
+ " -H \"Host: \" + host + "\"\n"
+ " -H \"X-TC-Action: \" + action + "\"\n"
+ " -H \"X-TC-Timestamp: \" + RequestTimestamp + "\"\n"
+ " -H \"X-TC-Version: \" + version + "\"\n"
+ " -H \"X-TC-Region: \" + region + "\"\n"
+ " -d '\" + payload + \"'";
cout << curlcmd << endl;
return 0;
};
```

C

```
#include <ctype.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <stdint.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

void get_utc_date(int64_t timestamp, char* utc, int len)
{
    // time_t timenow;
    struct tm sttime;
    sttime = *gmtime(&timestamp);
    strftime(utc, len, "%Y-%m-%d", &sttime);
}

void sha256_hex(const char* str, char* result)
{
    char buf[3];
    unsigned char hash[SHA256_DIGEST_LENGTH];
    SHA256_CTX sha256;
```

```
SHA256_Init(&sha256);
SHA256_Update(&sha256, str, strlen(str));
SHA256_Final(hash, &sha256);
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
    snprintf(buf, sizeof(buf), "%02x", hash[i]);
    strcat(result, buf);
}
}

void hmac_sha256(const char* key, int key_len,
const char* input, int input_len,
unsigned char* output, unsigned int* output_len)
{
    HMAC_CTX *h;
    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX hmac;
    HMAC_CTX_init(&hmac);
    h = &hmac;
    #else
    h = HMAC_CTX_new();
    #endif

    HMAC_Init_ex(h, key, key_len, EVP_sha256(), NULL);
    HMAC_Update(h, ( unsigned char* )input, input_len);
    HMAC_Final(h, output, output_len);

    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX_cleanup(h);
    #else
    HMAC_CTX_free(h);
    #endif

}

void hex_encode(const char* input, int input_len, char* output)
{
    static const char* const lut = "0123456789abcdef";

    char add_out[128] = {0};
    char temp[2] = {0};
    for (size_t i = 0; i < input_len; ++i)
    {
        const unsigned char c = input[i];
        temp[0] = lut[c >> 4];
        strcat(add_out, temp);
        temp[0] = lut[c & 15];
        strcat(add_out, temp);
    }
    strncpy(output, add_out, 128);
}

void lowercase(const char * src, char * dst)
```

```
{
for (int i = 0; src[i]; i++)
{
dst[i] = tolower(src[i]);
}
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
const char* SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
const char* SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");
const char* service = "cvm";
const char* host = "cvm.tencentcloudapi.com";
const char* region = "ap-guangzhou";
const char* action = "DescribeInstances";
const char* version = "2017-03-12";
int64_t timestamp = 1551113065;
char date[20] = {0};
get_utc_date(timestamp, date, sizeof(date));

// ***** 步骤 1: 拼接规范请求串 *****
const char* http_request_method = "POST";
const char* canonical_uri = "/";
const char* canonical_query_string = "";
char canonical_headers[100] = {"content-type:application/json; charset=utf-8\nhost:"};
strcat(canonical_headers, host);
strcat(canonical_headers, "\n\nx-tc-action:");
char value[100] = {0};
lowercase(action, value);
strcat(canonical_headers, value);
strcat(canonical_headers, "\n");
const char* signed_headers = "content-type;host;x-tc-action";
const char* payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}] }";
char hashed_request_payload[100] = {0};
sha256_hex(payload, hashed_request_payload);

char canonical_request[256] = {0};
sprintf(canonical_request, "%s\n%s\n%s\n%s\n%s\n%s", http_request_method,
canonical_uri, canonical_query_string, canonical_headers,
signed_headers, hashed_request_payload);
printf("%s\n", canonical_request);

// ***** 步骤 2: 拼接待签名字符串 *****
const char* algorithm = "TC3-HMAC-SHA256";
char request_timestamp[16] = {0};
sprintf(request_timestamp, "%d", timestamp);
char credential_scope[64] = {0};
strcat(credential_scope, date);
sprintf(credential_scope, "%s/%s/tc3_request", date, service);
```

```
char hashed_canonical_request[100] = {0};
sha256_hex(canonical_request, hashed_canonical_request);
char string_to_sign[256] = {0};
sprintf(string_to_sign, "%s\n%s\n%s\n%s", algorithm, request_timestamp,
credential_scope, hashed_canonical_request);
printf("%s\n", string_to_sign);

// ***** 步骤 3: 计算签名 *****
char k_key[64] = {0};
sprintf(k_key, "%s%s", "TC3", SECRET_KEY);
unsigned char k_date[64] = {0};
unsigned int output_len = 0;
hmac_sha256(k_key, strlen(k_key), date, strlen(date), k_date, &output_len);
unsigned char k_service[64] = {0};
hmac_sha256(k_date, output_len, service, strlen(service), k_service, &output_len);
unsigned char k_signing[64] = {0};
hmac_sha256(k_service, output_len, "tc3_request", strlen("tc3_request"), k_signing, &output_len);
unsigned char k_hmac_sha_sign[64] = {0};
hmac_sha256(k_signing, output_len, string_to_sign, strlen(string_to_sign), k_hmac_sha_sign, &output_len);

char signature[128] = {0};
hex_encode(k_hmac_sha_sign, output_len, signature);
printf("%s\n", signature);

// ***** 步骤 4: 拼接 Authorization *****
char authorization[512] = {0};
sprintf(authorization, "%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm, SECRET_ID, credential_scope, signed_headers, signature);
printf("%s\n", authorization);

char curlcmd[10240] = {0};
sprintf(curlcmd, "curl -X POST https://%s\n \
-H \"Authorization: %s\"\n \
-H \"Content-Type: application/json; charset=utf-8\"\n \
-H \"Host: %s\"\n \
-H \"X-TC-Action: %s\"\n \
-H \"X-TC-Timestamp: %s\"\n \
-H \"X-TC-Version: %s\"\n \
-H \"X-TC-Region: %s\"\n \
-d '%s'\",
host, authorization, host, action, request_timestamp, version, region, payload);
printf("%s\n", curlcmd);
return 0;
}
```

其他语言

- Lua: [GitHub](#)
- Swift: [GitHub](#)
- Dart: [GitHub](#)
- Shell(Bash): [GitHub](#)

签名失败

存在以下签名失败的错误码，请根据实际情况处理。

错误码	错误描述
AuthFailure.SignatureExpire	签名过期。Timestamp 与服务器接收到请求的时间相差不得超过五分钟。
AuthFailure.SecretIdNotFound	密钥不存在。请到控制台查看密钥是否被禁用，是否少复制了字符或者多了字符。
AuthFailure.SignatureFailure	签名错误。可能是签名计算错误，或者签名与实际发送的内容不相符合，也有可能是密钥 SecretKey 错误导致的。
AuthFailure.TokenFailure	临时证书 Token 错误。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。

签名方法

最近更新时间：2024-12-25 01:09:16

签名方法 v1 简单易用，但是功能和安全性都不如签名方法 v3，推荐使用签名方法 v3。

首次接触，建议使用 [API Explorer](#) 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v1”，可以生成签名过程进行验证，并提供了部分编程语言的签名示例，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 [Python](#)、[Java](#)、[PHP](#)、[Go](#)、[NodeJS](#)、[.NET](#)、[C++](#)、[Ruby](#)。

推荐使用 API Explorer

</> 点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（Signature）以验证请求者身份。签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往 [云API密钥页面](#) 申请，否则无法调用云 API 接口。

1. 申请安全凭证

在第一次使用云 API 之前，请前往 [云 API 密钥页面](#) 申请安全凭证。
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- 用户必须严格保管安全凭证，避免泄露。

申请安全凭证的具体步骤如下：

1. 登录 [腾讯云管理中心控制台](#)。
2. 前往 [云 API 密钥](#) 的控制台页面
3. 在 [云 API 密钥](#) 页面，单击【新建密钥】即可以创建一对 SecretId/SecretKey。

注意：每个账号最多可以拥有两对 SecretId/SecretKey。

2. 生成签名串

有了安全凭证 SecretId 和 SecretKey 后，就可以生成签名串了。以下是使用签名方法 v1 生成签名串的详细过程：

假设用户的 SecretId 和 SecretKey 分别是：

- SecretId: AKID*****
- SecretKey: *****

注意：这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作！

以云服务器查看实例列表（DescribeInstances）请求为例，当用户调用这一接口时，其请求参数可能如下：

参数名称	中文	参数值
Action	方法名	DescribeInstances
SecretId	密钥 ID	AKID*****
Timestamp	当前时间戳	1465185768
Nonce	随机正整数	11886
Region	实例所在区域	ap-guangzhou
InstanceIds.0	待查询的实例 ID	ins-09dx96dg
Offset	偏移量	0

参数名称	中文	参数值
Limit	最大允许输出	20
Version	接口版本号	2017-03-12

这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数。

2.1. 对参数排序

首先对所有请求参数按参数名的字典序（ASCII 码）升序排序。注意：1）只按参数名进行排序，参数值保持对应即可，不参与比大小；2）按 ASCII 码比大小，如 InstanceIds.2 要排在 InstanceIds.12 后面，不是按字母表，也不是按数值。用户可以借助编程语言中的相关排序函数来实现这一功能，如 PHP 中的 ksort 函数。上述示例参数的排序结果如下：

```
{
  'Action' : 'DescribeInstances',
  'InstanceIds.0' : 'ins-09dx96dg',
  'Limit' : 20,
  'Nonce' : 11886,
  'Offset' : 0,
  'Region' : 'ap-guangzhou',
  'SecretId' : 'AKID*****',
  'Timestamp' : 1465185768,
  'Version': '2017-03-12',
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

2.2. 拼接请求字符串

此步骤生成请求字符串。

将把上一步排序好的请求参数格式化成“参数名称=参数值”的形式，如对 Action 参数，其参数名称为 "Action"，参数值为 "DescribeInstances"，因此格式化后即为 Action=DescribeInstances。

注意：“参数值”为原始值而非 url 编码后的值。

然后将格式化后的各个参数用"&"拼接在一起，最终生成的请求字符串为：

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*****&Timestamp=1465185768&Version=2017-03-12
```

2.3. 拼接签名原字符串

此步骤生成签名原字符串。

签名原字符串由以下几个参数构成：

1. 请求方法: 支持 POST 和 GET 方式，这里使用 GET 请求，注意方法为全大写。
2. 请求主机: 查看实例列表(DescribeInstances)的请求域名为：cvm.tencentcloudapi.com。实际的请求域名根据接口所属模块的不同而不同，详见各接口说明。
3. 请求路径: 当前版本云API的请求路径固定为 /。
4. 请求字符串: 即上一步生成的请求字符串。

签名原串的拼接规则为：请求方法 + 请求主机 + 请求路径 + ? + 请求字符串。

示例的拼接结果为：

```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*****&Timestamp=1465185768&Version=2017-03-12
```

2.4. 生成签名串

此步骤生成签名串。

首先使用 HMAC-SHA1 算法对上一步中获得的签名原字符串进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。

具体代码如下，以 PHP 语言为例：

```
$secretKey = '*****';
$srcStr = 'GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*****&Timestamp=1465185768&Version=2017-03-12';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

最终得到的签名串为：

```
7RAM2xfNMO9EiVTNmPg06MRnCvQ=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

3. 签名串编码

生成的签名串并不能直接作为请求参数，需要对其进行 URL 编码。

如上一步生成的签名串为 7RAM2xfNMO9EiVTNmPg06MRnCvQ=，最终得到的签名串请求参数（Signature）为：7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D，它将用于生成最终的请求 URL。

注意：如果用户的请求方法是 GET，或者请求方法为 POST 同时 Content-Type 为 application/x-www-form-urlencoded，则发送请求时所有请求参数的值均需要做 URL 编码，参数键和=符号不需要编码。非 ASCII 字符在 URL 编码前需要以 UTF-8 进行编码。

注意：有些编程语言的库会自动为所有参数进行 urlencode，在这种情况下，就不需要对签名串进行 URL 编码了，否则两次 URL 编码会导致签名失败。

注意：其他参数值也需要进行编码，编码采用 RFC 3986。使用 %XY 对特殊字符例如汉字进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F），使用小写将引发错误。

4. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理。

错误代码	错误描述
AuthFailure.SignatureExpire	签名过期
AuthFailure.SecretIdNotFound	密钥不存在
AuthFailure.SignatureFailure	签名错误
AuthFailure.TokenFailure	token 错误
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）

5. 签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 [SDK 中心](#)。当前支持的编程语言有：

- [Python](#)
- [Java](#)
- [PHP](#)
- [Go](#)
- [NodeJS](#)

- [.NET](#)
- [C++](#)
- [Ruby](#)

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- [Signature Demo](#)

为了更清楚的解释签名过程，下面以实际编程语言为例，将上述的签名过程具体实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

最终输出的 url 可能为：`https://cvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*****&Signature=7RAM2xfNM09EiVTNmPg06MRnCvQ%3D&Timestamp=1465185768&Version=2017-03-12`。

注意：由于示例中的密钥是虚构的，时间戳也不是系统当前时间，因此如果将此 url 在浏览器中打开或者用 curl 等命令调用时会返回鉴权错误：签名过期。为了得到一个可以正常返回的 url，需要修改示例中的 SecretId 和 SecretKey 为真实的密钥，并使用系统当前时间戳作为 Timestamp。

注意：在下面的示例中，不同编程语言，甚至同一语言每次执行得到的 url 可能都有所不同，表现为参数的顺序不同，但这并不影响正确性。只要所有参数都在，且签名计算正确即可。

注意：以下代码仅适用于 API 3.0，不能直接用于其他的签名流程，请以对应的实际文档为准。

Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPIDemo {
    private final static String CHARSET = "UTF-8";

    public static String sign(String s, String key, String method) throws Exception {
        Mac mac = Mac.getInstance(method);
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(s.getBytes(CHARSET));
        return DatatypeConverter.printBase64Binary(hash);
    }

    public static String getStringToSign(TreeMap<String, Object> params) {
        StringBuilder s2s = new StringBuilder("GETcvm.tencentcloudapi.com/?");
        // 签名时要求对参数进行字典排序，此处用TreeMap保证顺序
        for (String k : params.keySet()) {
            s2s.append(k).append("=").append(params.get(k).toString()).append("&");
        }
        return s2s.toString().substring(0, s2s.length() - 1);
    }

    public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException {
        StringBuilder url = new StringBuilder("https://cvm.tencentcloudapi.com/?");
        // 实际请求的url中对参数顺序没有要求
    }
```

```
for (String k : params.keySet()) {
// 需要对请求串进行urlencode, 由于key都是英文字母, 故此处仅对其value进行urlencode
url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).append("&");
}
return url.toString().substring(0, url.length() - 1);
}

public static void main(String[] args) throws Exception {
TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap可以自动排序
// 实际调用时应当使用随机数, 例如: params.put("Nonce", new Random().nextInt(java.lang.Integer.MAX_VALUE));
params.put("Nonce", 11886); // 公共参数
// 实际调用时应当使用系统当前时间, 例如: params.put("Timestamp", System.currentTimeMillis() / 1000);
params.put("Timestamp", 1465185768); // 公共参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
params.put("SecretId", System.getenv("TENCENTCLOUD_SECRET_ID")); // 公共参数
params.put("Action", "DescribeInstances"); // 公共参数
params.put("Version", "2017-03-12"); // 公共参数
params.put("Region", "ap-guangzhou"); // 公共参数
params.put("Limit", 20); // 业务参数
params.put("Offset", 0); // 业务参数
params.put("InstanceIds.0", "ins-09dx96dg"); // 业务参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
params.put("Signature", sign(getStringToSign(params), System.getenv("TENCENTCLOUD_SECRET_KEY"), "HmacSHA1")); // 公共参数
System.out.println(getUrl(params));
}
}
```

Python

注意: 如果是在 Python 2 环境中运行, 需要先安装 requests 依赖包: `pip install requests`。

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import os
import time

import requests

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

def get_string_to_sign(method, endpoint, params):
s = method + endpoint + "/"
query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
return s + query_str

def sign_str(key, s, method):
hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
return base64.b64encode(hmac_str)
```

```
if __name__ == '__main__':
    endpoint = "cvm.tencentcloudapi.com"
    data = {
        'Action': 'DescribeInstances',
        'InstanceIds.0': 'ins-09dx96dg',
        'Limit': 20,
        'Nonce': 11886,
        'Offset': 0,
        'Region': 'ap-guangzhou',
        'SecretId': secret_id,
        'Timestamp': 1465185768, # int(time.time())
        'Version': '2017-03-12'
    }
    s = get_string_to_sign("GET", endpoint, data)
    data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
    print(data["Signature"])
    # 此处会实际调用，成功后可能产生计费
    # resp = requests.get("https://" + endpoint, params=data)
    # print(resp.url)
```

Golang

```
package main

import (
    "bytes"
    "crypto/hmac"
    "crypto/sha1"
    "encoding/base64"
    "fmt"
    "os"
    "sort"
    "strconv"
)

func main() {
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
    params := map[string]string{
        "Nonce": "11886",
        "Timestamp": strconv.Itoa(1465185768),
        "Region": "ap-guangzhou",
        "SecretId": secretId,
        "Version": "2017-03-12",
        "Action": "DescribeInstances",
        "InstanceIds.0": "ins-09dx96dg",
        "Limit": strconv.Itoa(20),
        "Offset": strconv.Itoa(0),
    }
}
```

```
var buf bytes.Buffer
buf.WriteString("GET")
buf.WriteString("cvm.tencentcloudapi.com")
buf.WriteString("/")
buf.WriteString("?")

// sort keys by ascii asc order
keys := make([]string, 0, len(params))
for k, _ := range params {
    keys = append(keys, k)
}
sort.Strings(keys)

for i := range keys {
    k := keys[i]
    buf.WriteString(k)
    buf.WriteString("=")
    buf.WriteString(params[k])
    buf.WriteString("&")
}
buf.Truncate(buf.Len() - 1)

hashed := hmac.New(sha1.New, []byte(secretKey))
hashed.Write(buf.Bytes())

fmt.Println(base64.StdEncoding.EncodeToString(hashed.Sum(nil)))
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$param["Nonce"] = 11886;//rand();
$param["Timestamp"] = 1465185768;//time();
$param["Region"] = "ap-guangzhou";
$param["SecretId"] = $secretId;
$param["Version"] = "2017-03-12";
$param["Action"] = "DescribeInstances";
$param["InstanceIds.0"] = "ins-09dx96dg";
$param["Limit"] = 20;
$param["Offset"] = 0;

ksort($param);

$signStr = "GETcvm.tencentcloudapi.com/?";
foreach ( $param as $key => $value ) {
    $signStr = $signStr . $key . "=" . $value . "&";
}
$signStr = substr($signStr, 0, -1);
```

```
$signature = base64_encode(hash_hmac("sha1", $signStr, $secretKey, true));
echo $signature.PHP_EOL;
// need to install and enable curl extension in php.ini
// $param["Signature"] = $signature;
// $url = "https://cvm.tencentcloudapi.com/?".http_build_query($param);
// echo $url.PHP_EOL;
// $ch = curl_init();
// curl_setopt($ch, CURLOPT_URL, $url);
// $output = curl_exec($ch);
// curl_close($ch);
// echo json_decode($output);
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'time'
require 'openssl'
require 'base64'

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

method = 'GET'
endpoint = 'cvm.tencentcloudapi.com'
data = {
  'Action' => 'DescribeInstances',
  'InstanceIds.0' => 'ins-09dx96dg',
  'Limit' => 20,
  'Nonce' => 11886,
  'Offset' => 0,
  'Region' => 'ap-guangzhou',
  'SecretId' => secret_id,
  'Timestamp' => 1465185768, # Time.now.to_i
  'Version' => '2017-03-12',
}
sign = method + endpoint + '/?'
params = []
data.sort.each do |item|
  params << "#{item[0]}=#{item[1]}"
end
sign += params.join('&')
digest = OpenSSL::Digest.new('sha1')
data['Signature'] = Base64.encode64(OpenSSL::HMAC.digest(digest, secret_key, sign))
puts data['Signature']

# require 'net/http'
# uri = URI('https://' + endpoint)
# uri.query = URI.encode_www_form(data)
# p uri
```

```
# res = Net::HTTP.get_response(uri)
# puts res.body
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Security.Cryptography;
using System.Text;

public class Application {
    public static string Sign(string signKey, string secret)
    {
        string signRet = string.Empty;
        using (HMACSHA1 mac = new HMACSHA1(Encoding.UTF8.GetBytes(signKey)))
        {
            byte[] hash = mac.ComputeHash(Encoding.UTF8.GetBytes(secret));
            signRet = Convert.ToBase64String(hash);
        }
        return signRet;
    }

    public static string MakeSignPlainText(SortedDictionary<string, string> requestParams, string requestMethod, string requestHost, string requestPath)
    {
        string retStr = "";
        retStr += requestMethod;
        retStr += requestHost;
        retStr += requestPath;
        retStr += "?";
        string v = "";
        foreach (string key in requestParams.Keys)
        {
            v += string.Format("{0}={1}&", key, requestParams[key]);
        }
        retStr += v.TrimEnd('&');
        return retStr;
    }

    public static void Main(string[] args)
    {
        // 密钥参数
        // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
        string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
        // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
        string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

        string endpoint = "cvm.tencentcloudapi.com";
        string region = "ap-guangzhou";
        string action = "DescribeInstances";
        string version = "2017-03-12";
        double RequestTimestamp = 1465185768; // 时间戳 2019-02-26 00:44:25, 此参数作为示例, 以实际为准
        // long timestamp = ToTimestamp() / 1000;
```

```
// string requestTimestamp = timestamp.ToString();
Dictionary<string, string> param = new Dictionary<string, string>();
param.Add("Limit", "20");
param.Add("Offset", "0");
param.Add("InstanceIds.0", "ins-09dx96dg");
param.Add("Action", action);
param.Add("Nonce", "11886");
// param.Add("Nonce", Math.Abs(new Random().Next()).ToString());

param.Add("Timestamp", RequestTimestamp.ToString());
param.Add("Version", version);

param.Add("SecretId", SECRET_ID);
param.Add("Region", region);
SortedDictionary<string, string> headers = new SortedDictionary<string, string>(param, StringComparer.Ordinal);
string sigInParameter = MakeSignPlainText(headers, "GET", endpoint, "/");
string sigOutParam = Sign(SECRET_KEY, sigInParameter);
Console.WriteLine(sigOutParam);
}
}
```

NodeJS

```
const crypto = require('crypto');

function get_req_url(params, endpoint){
    params['Signature'] = encodeURIComponent(params['Signature']);
    const url_strParam = sort_params(params)
    return "https://" + endpoint + "/" + "?" + url_strParam.slice(1);
}

function formatSignString(reqMethod, endpoint, path, strParam){
    let strSign = reqMethod + endpoint + path + "?" + strParam.slice(1);
    return strSign;
}

function sha1(secretKey, strsign){
    let signMethodMap = {'HmacSHA1': "sha1"};
    let hmac = crypto.createHmac(signMethodMap['HmacSHA1'], secretKey || "");
    return hmac.update(Buffer.from(strsign, 'utf8')).digest('base64')
}

function sort_params(params){
    let strParam = "";
    let keys = Object.keys(params);
    keys.sort();
    for (let k in keys) {
        //k = k.replace(/_/g, '.');
        strParam += ("&" + keys[k] + "=" + params[keys[k]]);
    }
    return strParam
}

function main(){
```

```
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const Region = "ap-guangzhou"
const Version = "2017-03-12"
const Action = "DescribeInstances"
const Timestamp = 1465185768 // 时间戳 2016-06-06 12:02:48, 此参数作为示例, 以实际为准
// const Timestamp = Math.round(Date.now() / 1000)
const Nonce = 11886 // 随机正整数
//const nonce = Math.round(Math.random() * 65535)

let params = {};
params['Action'] = Action;
params['InstanceIds.0'] = 'ins-09dx96dg';
params['Limit'] = 20;
params['Offset'] = 0;
params['Nonce'] = Nonce;
params['Region'] = Region;
params['SecretId'] = SECRET_ID;
params['Timestamp'] = Timestamp;
params['Version'] = Version;

// 1. 对参数排序, 并拼接请求字符串
strParam = sort_params(params)

// 2. 拼接签名原字符串
const reqMethod = "GET";
const path = "/";
strSign = formatSignString(reqMethod, endpoint, path, strParam)
// console.log(strSign)

// 3. 生成签名串
params['Signature'] = sha1(SECRET_KEY, strSign)
console.log(params['Signature'])

// 4. 进行url编码并拼接请求url
// const req_url = get_req_url(params, endpoint)
// console.log(params['Signature'])
// console.log(req_url)
}

main()
```

返回结果

最近更新时间：2024-03-12 01:08:09

云 API 3.0 接口默认返回 JSON 数据，返回非 JSON 格式的接口会在文档中做出说明。返回 JSON 数据时最大限制为 50 MB，如果返回的数据超过最大限制，请求会失败并返回内部错误。请根据接口文档中给出的过滤功能（例如时间范围）或者分页功能，控制返回数据不要过大。

注意：目前只要请求被服务端正常处理了，响应的 HTTP 状态码均为 200。例如返回的消息体里的错误码是签名失败，但 HTTP 状态码是 200，而不是 401。

正确返回结果

以云服务器的接口查看实例状态列表 (DescribeInstancesStatus) 2017-03-12 版本为例，若调用成功，其可能的返回如下为：

```
{
  "Response": {
    "TotalCount": 0,
    "InstanceStatusSet": [],
    "RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
  }
}
```

- Response 及其内部的 RequestId 是固定的字段，无论请求成功与否，只要 API 处理了，则必定会返回。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 [腾讯云客服](#) 或 [提交工单](#)，并提供该 ID 来解决问题。
- 除了固定的字段外，其余均为具体接口定义的字段，不同的接口所返回的字段参见接口文档中的定义。此例中的 TotalCount 和 InstanceStatusSet 均为 DescribeInstancesStatus 接口定义的字段，由于调用请求的用户暂时还没有云服务器实例，因此 TotalCount 在此情况下的返回值为 0，InstanceStatusSet 列表为空。

错误返回结果

若调用失败，其返回值示例如下为：

```
{
  "Response": {
    "Error": {
      "Code": "AuthFailure.SignatureFailure",
      "Message": "The provided credentials could not be validated. Please check your signature is correct."
    },
    "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
  }
}
```

- Error 的出现代表着该请求调用失败。Error 字段连同其内部的 Code 和 Message 字段在调用失败时是必定返回的。
- Code 表示具体出错的错误码，当请求出错时可以先根据该错误码在公共错误码和当前接口对应的错误码列表里面查找对应原因和解决方案。
- Message 显示出了这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新，用户不应依赖这个返回值。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 [腾讯云客服](#) 或 [提交工单](#)，并提供该 ID 来解决问题。

公共错误码

返回结果中如果存在 Error 字段，则表示调用 API 接口失败。Error 中的 Code 字段表示错误码，所有业务都可能出现的错误码为公共错误码。完整的错误码列表请参考本产品“API 文档”目录下的“错误码”页面。

参数类型

最近更新时间：2023-07-24 01:08:51

目前腾讯云 API 3.0 输入参数和输出参数支持如下几种数据格式：

- String: 字符串。
- Integer: 整型，上限为无符号64位整数。SDK 3.0 不同编程语言支持的类型有所差异，建议以所使用编程语言的最大整型定义，例如 Golang 的 `uint64`。
- Boolean: 布尔型。
- Float: 浮点型。
- Double: 双精度浮点型。
- Date: 字符串，日期格式。例如：2022-01-01。
- Timestamp: 字符串，时间格式。例如：2022-01-01 00:00:00。
- Timestamp ISO8601: ISO 8601 是由国际标准化组织（International Organization for Standardization，ISO）发布的关于日期和时间格式的国际标准，对应国标《[GB/T 7408-2005数据元和交换格式信息交换日期和时间表示法](#)》。建议以所使用编程语言的标准库进行格式解析。例如：2022-01-01T00:00:00+08:00。
- Binary: 二进制内容，需要以特定协议请求和解析。

高防IP资产实例相关接口

绑定高防弹性公网IP

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

本接口 (AssociateDDoSEipAddress) 用于将高防弹性公网IP绑定到实例或弹性网卡的指定内网 IP 上。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：AssociateDDoSEipAddress。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID，实例ID形如：bgpip-0000011x。只能填写高防IP实例。 示例值：bgpip-0000011x
Eip	是	String	资源实例ID对应的高防弹性公网IP。 示例值：1.1.1.1
CvmInstanceId	是	String	要绑定的实例 ID。实例 ID 形如：ins-11112222。可通过登录控制台查询，也可通过 DescribeInstances 接口返回值中的 InstanceId 获取。 示例值：ins-11112222
CvmRegion	是	String	cvm实例所在地域，例如：ap-hongkong。 示例值：ap-hongkong

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 绑定高防弹性公网IP到实例上

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: AssociateDDoSEipAddress
```

<公共请求参数>

```
{
  "InstanceId": "bgpip-0000011x",
  "Eip": "1.1.1.1",
  "CvmInstanceId": "ins-11112222",
  "CvmRegion": "ap-hongkong"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。

错误码	描述
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

绑定高防弹性公网IP到Clb

最近更新时间：2025-04-25 01:07:52

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

本接口 (AssociateDDoSEipLoadBalancer) 用于将高防弹性公网IP绑定到负载均衡指定内网 IP 上。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：AssociateDDoSEipLoadBalancer。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID，实例ID形如：bgpip-0000011x。只能填写高防IP实例。 示例值：bgpip-0000011x
Eip	是	String	资源实例ID对应的高防弹性公网IP。 示例值：1.1.1.1
LoadBalancerID	是	String	要绑定的负载均衡ID。负载均衡 ID 形如：lb-0000002i。可通过登录控制台查询，也可通过 DescribeLoadBalancers 接口返回值中的LoadBalancerId获取。 示例值：lb-0000002i
LoadBalancerRegion	是	String	CLB所在地域，例如：ap-hongkong。 示例值：ap-hongkong
Vip	否	String	CLB内网IP 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 绑定高防弹性公网IP到负载均衡上

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
```

```
X-TC-Action: AssociateDDoS_EipLoadBalancer
```

<公共请求参数>

```
{
  "InstanceId": "bgpip-0000011x",
  "Eip": "1.1.1.1",
  "Vip": "1.1.1.1",
  "LoadBalancerRegion": "ap-hongkong",
  "LoadBalancerID": "lb-0000002i"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。

错误码	描述
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

解绑高防弹性公网IP

最近更新时间：2025-04-25 01:07:52

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

本接口 (DisassociateDDoSEipAddress) 用于解绑高防弹性公网IP。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DisassociateDDoSEipAddress。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID，实例ID形如：bgpip-0000011x。只能填写高防IP实例。 示例值：bgpip-0000011x
Eip	是	String	资源实例ID对应的高防弹性公网IP。 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 解绑高防弹性公网IP

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DisassociateDDoSEipAddress
<公共请求参数>

{
  "InstanceId": "bgpip-0000011x",
  "Eip": "1.1.1.1"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

资产列表相关接口

绑定IP到高防包实例

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

绑定IP到高防包实例，支持独享包、共享包（新版）；需要注意的是此接口绑定或解绑IP是异步接口，当处于绑定或解绑中时，则不允许再进行绑定或解绑，需要等待当前绑定或解绑完成。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateBoundIP。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp表示独享包；bgp-multip表示共享包） 示例值：bgp
Id	是	String	资源实例ID 示例值：bgp-000000x1
BoundDevList.N	否	Array of BoundIpInfo	绑定到资源实例的IP数组，当资源实例为高防包(独享包)时，数组只允许填1个IP；当没有要绑定的IP时可以为空数组；但是BoundDevList和UnBoundDevList至少有一个不为空；
UnBoundDevList.N	否	Array of BoundIpInfo	与资源实例解绑的IP数组，当资源实例为高防包(独享包)时，数组只允许填1个IP；当没有要解绑的IP时可以为空数组；但是BoundDevList和UnBoundDevList至少有一个不为空；
CopyPolicy	否	String	已弃用，不填 示例值：yes
FilterRegion	否	String	如果该资源实例为域名化资产以及跨地域绑定则，该参数必填 示例值：ap-guangzhou

3. 输出参数

参数名称	类型	描述
Success	SuccessCode	成功码
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 绑定资产到高防包

绑定资产到高防包

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateBoundIP
<公共请求参数>

{
  "BoundDevList": [
    {
      "BizType": "cvm",
      "Ip": "1.1.1.1",
      "InstanceId": "id-xxx",
      "DeviceType": "cvm",
      "IspCode": 1,
      "Domain": "www.abc.com"
    }
  ],
  "Business": "bgp",
  "Id": "id-xxx",
  "UnBoundDevList": [
    {
      "BizType": "cvm",
      "Ip": "1.1.1.1",
      "InstanceId": "id-xxx",
      "DeviceType": "cvm",
      "IspCode": 1,
      "Domain": "www.abc.com"
    }
  ],
  "CopyPolicy": "policy",
  "FilterRegion": "guangzhou"
}
```

输出示例

```
{
  "Response": {
    "Success": {
      "Message": "msg",
      "Code": "200"
    },
    "RequestId": "08afbb87-5a2c-4562-acbd-eeF06a47c7db"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceInUse	资源被占用。
UnsupportedOperation	操作不支持。

查询与证书ID匹配的七层规则

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

查询与证书ID对于域名匹配的七层规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeL7RulesBySSLCertId。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Status	是	String	域名状态，可取bindable, binded, opened, closed, all，all表示全部状态 示例值：bindable
CertIds.N	是	Array of String	证书ID列表 示例值：["cMgC0TpV", "kmLKvJiD", "kmCL9Kv7", "kneSCNrZ"]

3. 输出参数

参数名称	类型	描述
CertSet	Array of CertIdInsL7Rules	证书规则集合
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询可绑定证书的域名规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeL7RulesBySSLCertId
<公共请求参数>

{
  "Status": "bindable",
  "CertIds": [
```

```
"cMgC0TpV"  
]  
}
```

输出示例

```
{  
  "Response": {  
    "CertSet": [  
      {  
        "CertId": "kneSCNrZ",  
        "L7Rules": [  
          {  
            "InsId": "bgpip-000001eo",  
            "AppId": "251008915",  
            "Protocol": "https",  
            "Domain": "www.xxx.com",  
            "VirtualPort": "443",  
            "SSLId": "",  
            "Status": 8  
          }  
        ]  
      }  
    ],  
    "RequestId": "d87b0215-5dc6-4b99-ad82-704f0ef1bfb7"  
  }  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnsupportedOperation	操作不支持。

获取高防IP资产实例列表

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取高防IP资产实例列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListBGPIPIstances。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为20;最大取值为100 示例值：25
FilterIp	否	String	IP搜索 示例值：1.1.1.1
FilterInstanceId	否	String	资产实例ID搜索，例如，bgpip-00000001 示例值：bgpip-00000001
FilterLine	否	Integer	高防IP线路搜索，取值为[1: BGP线路 2: 电信 3: 联通 4: 移动 99: 第三方合作线路] 示例值：1
FilterRegion	否	String	地域搜索，例如，ap-guangzhou 示例值：ap-guangzhou
FilterName	否	String	名称搜索 示例值：name
FilterEipType	否	Integer	是否只获取高防弹性公网IP实例。填写时，只能填写1或者0。当填写1时，表示返回高防弹性公网IP实例。当填写0时，表示返回非高防弹性公网IP实例。 示例值：1
FilterEipEipAddressStatus.N	否	Array of String	高防弹性公网IP实例的绑定状态搜索条件，取值范围 [BINDING、BIND、UNBINDING、UNBIND]。该搜索条件只在FilterEipType=1时才有效。 示例值：1

参数名称	必选	类型	描述
FilterDamDDoSStatus	否	Integer	是否只获取安全加速实例。填写时，只能填写1或者0。当填写1时，表示返回安全加速实例。当填写0时，表示返回非安全加速实例。 示例值：1
FilterStatus	否	String	获取特定状态的资源，运行中填idle，攻击中填attacking，封堵中填blocking，试用资源填trial 示例值：attacking
FilterCname	否	String	获取特定的实例Cname 示例值：abcdefg
FilterInstanceIdList.N	否	Array of String	批量查询实例ID对应的高防IP实例资源 示例值：id-xxxx
FilterTag	否	TagFilter	标签搜索 示例值：tag
FilterPackType.N	否	Array of String	按照套餐类型进行过滤 示例值：type
FilterConvoy	否	Integer	重保护航搜索 示例值：1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：2
InstanceList	Array of BGPIInstance	高防IP资产实例列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取高防IP资产实例列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeListBGPIInstances
<公共请求参数>

{
  "Limit": 1,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
```

```
"RequestId": "0381c271-292d-4e2a-adcc-cd7bcd6977a7",
"InstanceList": [
{
  "InstanceDetail": {
    "EipList": [
      "2402:4e00:31:804::17"
    ],
    "InstanceId": "bgpip-00000001"
  },
  "SpecificationLimit": {
    "ProtectBandwidth": 30000,
    "ProtectCCQPS": 40000,
    "ElasticBandwidth": 50000,
    "NormalBandwidth": 1000,
    "ForwardRulesLimit": 60,
    "AutoRenewFlag": 0,
    "Line": 1
  },
  "Usage": {
    "PortRulesUsage": 0,
    "DomainRulesUsage": 0,
    "Last7DayAttackCount": 0
  },
  "Region": {
    "Region": "ap-shanghai"
  },
  "PackInfo": null,
  "StaticPackRelation": null,
  "EipAddressPackRelation": null,
  "AnycastOutPackRelation": null,
  "Tgw": 2,
  "ZoneId": 0,
  "Name": "",
  "EOFlag": 0,
  "ElasticBandwidth": 2000,
  "V6Flag": 1,
  "Domain": "",
  "Status": "idle",
  "ExpiredTime": "2023-01-08 21:22:24",
  "CreatedTime": "2022-12-08 21:22:24",
  "EipAddressStatus": "",
  "EipFlag": 0,
  "EipAddressInfo": null,
  "ConvoyId": "",
  "DamDDoSStatus": 0,
  "BGPIPChannelFlag": 0,
  "TagInfoList": [],
  "InstanceVersion": 3
}
],
"Total": 12
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。

获取高防包资产实例列表

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取高防包资产实例列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListBGPIstances。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为20;最大取值为100 示例值：25
FilterIp	否	String	IP搜索 示例值：1.1.1.1
FilterInstanceId	否	String	资产实例ID搜索，例如，bgp-00000001 示例值：bgp-00000001
FilterRegion	否	String	地域搜索，例如，ap-guangzhou 示例值：ap-guangzhou
FilterName	否	String	名称搜索 示例值：name
FilterLine	否	Integer	按照线路搜索，1: BGP; 2: 三网 示例值：1
FilterStatus	否	String	状态搜索，idle：运行中；attacking：攻击中；blocking：封堵中 示例值：attacking
FilterBoundStatus	否	String	高防包绑定状态搜索，bounding：绑定中；failed：绑定失败 示例值：bounding
FilterInstanceIdList.N	否	Array of String	实例id数组 示例值：[bgp-00000001]
FilterEnterpriseFlag	否	Integer	企业版搜索，1: 包含重保护航套餐下的企业版列表，2: 不包含重保护航套餐的企业版列表 示例值：1
FilterLightFlag	否	Integer	轻量版搜索 示例值：1

参数名称	必选	类型	描述
FilterChannelFlag	否	Integer	定制版搜索 示例值：1
FilterTag	否	TagFilter	标签搜索
FilterTrialFlag	否	Integer	试用资源搜索，1: 应急防护资源；2: PLG试用资源 示例值：1
FilterConvoy	否	Integer	重保护航搜索 示例值：1
ExcludeAdvancedInfo	否	Boolean	默认false；接口传true，返回数据中不包含高级信息，高级信息包含：InstanceList[0].Usage。 示例值：false
FilterAssetIpList.N	否	Array of String	资产IP数组 示例值：[1.1.1.1]
FilterBasicPlusFlag	否	Integer	是否包含基础防护增强版 0: 不包含 1: 包含 示例值：1
FilterPlanCntFlag	否	Integer	是否标准版2.0 0: 包含标准版2.0 0 1: 只查询标准版2.0 0 2: 不查标准版2.0 示例值：1
FilterTransRegionFlag	否	Integer	是否跨区域产品 0: 不包含跨区域产品 1: 中国大陆跨区域产品 2: 非中国大陆跨区域产品 3: 包含全部 示例值：0
FilterZoneldList.N	否	Array of Integer	zoenid列表 示例值：1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
InstanceList	Array of BGPInstance	高防包资产实例列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取高防包资产实例列表

获取高防包资产实例列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeListBGPInstances
<公共请求参数>

{
  "Limit": 1,
  "Offset": 0,
```

```
"FilterInstanceId": "bgp-00000001",
"FilterIp": "1.1.1.1"
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "InstanceList": [
      {
        "InstanceDetail": {
          "EipList": [
            "1.1.1.1"
          ],
          "InstanceId": "id-xxx"
        },
        "SpecificationLimit": {
          "ProtectBandwidth": 1,
          "ProtectCountLimit": 1,
          "ProtectIPNumberLimit": 1,
          "AutoRenewFlag": 1,
          "UnionPackFlag": 1,
          "ServiceBandWidth": 1,
          "BattleEditionFlag": 1,
          "ChannelEditionFlag": 1,
          "EnterpriseFlag": 1,
          "ElasticLimit": 1,
          "DownGradeProtect": 1
        },
        "Usage": {
          "ProtectCountUsage": 1,
          "ProtectIPNumberUsage": 1,
          "Last7DayAttackCount": 1
        },
        "Region": {
          "Region": "guangzhou"
        },
        "Status": "1",
        "CreatedTime": "2020-09-22 00:00:00",
        "ExpiredTime": "2020-09-22 00:00:00",
        "Name": "name",
        "PackInfo": {
          "PackType": "type",
          "PackId": "id"
        },
        "EipProductInfos": [
          {
            "Ip": "1.1.1.1",
            "BizType": "eip",
            "DeviceType": "eip",
            "InstanceId": "ins-xx",
            "Domain": "www.abc.com"
          }
        ]
      }
    ]
  }
}
```

```
}
],
"BoundStatus": "creat",
"DDoSLevel": "1",
"CCEnable": 0,
"TagInfoList": [
{
"TagKey": "key",
"TagValue": "value"
}
],
"IpCountNewFlag": 1,
"VitalityVersion": 1,
"Line": 1,
"FreeServiceBandwidth": 1,
"ElasticServiceBandwidth": 1,
"GiftServiceBandWidth": 0,
"ModifyTime": "2024.12.1",
"BasicPlusFlag": 1
}
],
"RequestId": "08afbb87-5a2c-4562-acbd-eeef06a47c7db"
}
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。

获取转发监听器列表

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取转发监听器列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListListener。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。

3. 输出参数

参数名称	类型	描述
Layer4Listeners	Array of Layer4Rule	4层转发监听器列表
Layer7Listeners	Array of Layer7Rule	7层转发监听器列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取转发监听器列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListListener
&<<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "83ef9ee6-a46a-4aa7-b8aa-42c65637edad",
    "Layer4Listeners": [
      {
        "BackendPort": 12,
        "FrontendPort": 12,
```

```
"Protocol": "TCP",
"RealServers": [
{
"RealServer": "1.1.1.1",
"RsType": 2,
"Weight": 50
}
],
"InstanceDetails": [
{
"EipList": [
"1.1.1.1"
],
"InstanceId": "bgpip-0000011e"
}
]
},
{
"Layer7Listeners": [
{
"Domain": "121.test.com",
"ProxyTypeList": [
{
"ProxyPorts": [
443
],
"ProxyType": "https"
}
],
"RealServers": [
{
"RealServer": "1.1.1.1",
"RsType": 2,
"Weight": 100
}
],
"InstanceDetails": [
{
"EipList": [
"1.1.1.1"
],
"InstanceId": "bgpip-000000y2"
}
]
}
]
}
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

高防IP获取7层规则

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

高防IP获取7层规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeNewL7Rules。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
StatusList.N	否	Array of Integer	状态搜索，选填，取值[0(规则配置成功)，1(规则配置生效中)，2(规则配置失败)，3(规则删除生效中)，5(规则删除失败)，6(规则等待配置)，7(规则等待删除)，8(规则待配置证书)] 示例值：[1,2]
Domain	否	String	域名搜索，选填，当需要搜索域名请填写 示例值：www.abc.com
Ip	否	String	IP搜索，选填，当需要搜索IP请填写 示例值：1.1.1.1
Limit	否	Integer	一页条数，默认值100，最大值100，超过100最大返回100条 示例值：10
Offset	否	Integer	规则偏移量，取值为(页码-1)*一页条数 示例值：1
ProtocolList.N	否	Array of String	转发协议搜索，选填，取值[http, https, http/https] 示例值：http
Cname	否	String	高防IP实例的Cname 示例值：abcdefg
Export	否	Boolean	默认为false，当为true时，将不对各个规则做策略检查，直接导出所有规则 示例值：false

3. 输出参数

参数名称	类型	描述
Rules	Array of NewL7RuleEntry	转发规则列表

参数名称	类型	描述
Healths	Array of L7RuleHealth	健康检查配置列表
Total	Integer	总规则数 示例值：1
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取高防ip所有规则

获取高防ip所有规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeNewL7Rules
<公共请求参数>

{
  "StatusList": [
    1
  ],
  "Domain": "www.abc.com",
  "Business": "bgp",
  "Ip": "1.1.1.1",
  "Limit": 1,
  "Offset": 1,
  "ProtocolList": [
    "1.1.1.1"
  ],
  "Cname": "www.abc.com"
}
```

输出示例

```
{
  "Response": {
    "Rules": [
      {
        "Region": 1,
        "Id": "bgp-121",
        "Ip": "1.1.1.1",
        "RuleId": "idxxx",
        "RuleName": "namexxx",
        "Protocol": "http",
        "Domain": "www.abc.com",
        "LbType": 1,
        "KeepEnable": 1,
        "KeepTime": 1,

```

```
"SourceType": 1,
"CertType": 1,
"SSLId": "idxxx",
"Cert": "cert",
"PrivateKey": "key",
"SourceList": [
{
"Source": "1.1.1.1",
"Weight": 1,
"Port": 1,
"Backup": 1
}
],
"Status": 1,
"CCStatus": 1,
"CCEnable": 1,
"CCThreshold": 1,
"CCLevel": "2",
"ModifyTime": "2020-09-22 00:00:00",
"HttpsToHttpEnable": 1,
"VirtualPort": 1,
"RewriteHttps": 1,
"ErrCode": 1,
"Version": 1
}
],
"Healths": [
{
"Status": 1,
"Enable": 1,
"RuleId": "ruleid",
"Url": "url",
"Interval": 1,
"AliveNum": 1,
"KickNum": 1,
"Method": "get",
"StatusCode": 1,
"ProtocolFlag": 1,
"PassiveEnable": 1,
"BlockInter": 1,
"FailedCountInter": 1,
"FailedThreshold": 1,
"PassiveStatusCode": 1,
"PassiveStatus": 1
}
],
"Total": 1,
"RequestId": "08afbb87-5a2c-4562-acbd-eef06a47c7db"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。

高防IP获取7层转发规则

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

高防IP获取7层规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBGPIPL7Rules。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，此参数为可选参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
StatusList.N	否	Array of Integer	状态搜索，选填，取值[0(规则配置成功)，1(规则配置生效中)，2(规则配置失败)，3(规则删除生效中)，5(规则删除失败)，6(规则等待配置)，7(规则等待删除)，8(规则待配置证书)] 示例值：creat
Domain	否	String	域名搜索，选填，当需要搜索域名请填写 示例值：www.xxx.com
Ip	否	String	IP搜索，选填，当需要搜索IP请填写 示例值：1.1.1.1
Limit	否	Integer	一页条数，默认值100，最大值100，超过100最大返回100条 示例值：10
Offset	否	Integer	规则偏移量，取值为(页码-1)*一页条数 示例值：1
ProtocolList.N	否	Array of String	转发协议搜索，选填，取值[http, https, http/https] 示例值：http
Cname	否	String	高防IP实例的Cname 示例值：abcdefg
Export	否	Boolean	默认为false，当为true时，将不对各个规则做策略检查，直接导出所有规则 示例值：false
Source	否	String	源站，模糊查询 示例值：1.1.1.1:80

3. 输出参数

参数名称	类型	描述
Rules	Array of BGPIPL7RuleEntry	转发规则列表
Healths	Array of L7RuleHealth	健康检查配置列表
Total	Integer	总规则数 示例值：1
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 请求示例

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBGPIPL7Rules
<公共请求参数>

{
  "StatusList": [
    1
  ],
  "Domain": "www.abc.com",
  "Business": "bgp",
  "Ip": "1.1.1.1",
  "Limit": 1,
  "Offset": 1,
  "ProtocolList": [
    "http"
  ],
  "Cname": "www.ab.com",
  "Export": true
}
```

输出示例

```
{
  "Response": {
    "Rules": [
      {
        "Region": 1,
        "Id": "id",
        "Ip": "1.1.1.1",
        "RuleId": "id",
        "RuleName": "name",
        "Protocol": "http",
        "Domain": "www.abc.com",
        "LbType": 1,

```

```
"KeepEnable": 1,
"KeepTime": 1,
"SourceType": 1,
"CertType": 1,
"SSLId": "id",
"Cert": "cert",
"PrivateKey": "key",
"SourceList": [
{
"Source": "1.1.1.1",
"Weight": 1,
"Port": 1,
"Backup": 1
}
],
"Status": 1,
"CCStatus": 1,
"CCEnable": 1,
"CCThreshold": 1,
"CCLevel": "1",
"ModifyTime": "2020-09-22 00:00:00",
"HttpsToHttpEnable": 1,
"VirtualPort": 1,
"RewriteHttps": 1,
"ErrCode": 1,
"Version": 1
}
],
"Healths": [
{
"Status": 1,
"Enable": 1,
"RuleId": "id",
"Url": "url",
"Interval": 1,
"AliveNum": 1,
"KickNum": 1,
"Method": "post",
"StatusCode": 1,
"ProtocolFlag": 1,
"PassiveEnable": 1,
"BlockInter": 1,
"FailedCountInter": 1,
"FailedThreshold": 1,
"PassiveStatusCode": 1,
"PassiveStatus": 1
}
],
"Total": 1,
"RequestId": "08afbb87-5a2c-4562-acbd-eeF06a47c7db"
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。

DDoS攻击防护策略相关接口

修改DDoSIP防护等级

最近更新时间：2025-04-25 01:08:05

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

读取或修改DDoS的防护等级

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDDoSLevel。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Id	是	String	资源ID 示例值：bgpip-000000xe
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版） 示例值：bgpip
Method	是	String	=get表示读取防护等级；=set表示修改防护等级 示例值：set
DDoSLevel	否	String	防护等级，取值[low,middle,high]；当Method=set时必须填 示例值：low

3. 输出参数

参数名称	类型	描述
DDoSLevel	String	防护等级，取值[low,middle,high] 示例值：middle
Id	String	资源ID 示例值：bgpip-000000xe
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS的防护等级

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyDDoSLevel
<公共请求参数>
```

```
{
  "Id": "bgpip-000000xe",
  "Business": "bgpip",
  "Method": "get"
}
```

输出示例

```
{
  "Response": {
    "DDoSLevel": "middle",
    "Id": "bgpip-000000xe",
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
InvalidParameterValue	参数取值错误。

修改DDoS清洗阈值

最近更新时间：2025-04-25 01:08:05

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

修改DDoS清洗阈值

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDDoSThreshold。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Threshold	是	Integer	DDoS清洗阈值，取值[0, 60, 80, 100, 150, 200, 250, 300, 400, 500, 700, 1000]； 当设置值为0时，表示采用默认值； 示例值：1000
Id	是	String	资源ID 示例值：bgpip-000000xe
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版） 示例值：bgpip
OtherThresholdFlag	否	Integer	配置其他阈值标志位，1表示配置其他阈值 示例值：1
SynFloodThreshold	否	Integer	SYN FLOOD流量阈值 示例值：1000
SynFloodPktThreshold	否	Integer	SYN FLOOD包量阈值 示例值：1000
UdpFloodThreshold	否	Integer	UDP FLOOD流量阈值 示例值：1000
UdpFloodPktThreshold	否	Integer	UDP FLOOD包量阈值 示例值：1000
AckFloodThreshold	否	Integer	ACK FLOOD流量阈值 示例值：1000
AckFloodPktThreshold	否	Integer	ACK FLOOD包量阈值 示例值：1000
SynAckFloodThreshold	否	Integer	SYNACK FLOOD流量阈值 示例值：1000
SynAckFloodPktThreshold	否	Integer	SYNACK FLOOD包量阈值

参数名称	必选	类型	描述
			示例值：1000
RstFloodThreshold	否	Integer	RST FLOOD流量阈值 示例值：1000
RstFloodPktThreshold	否	Integer	RST FLOOD包量阈值 示例值：1000

3. 输出参数

参数名称	类型	描述
Success	SuccessCode	成功码
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS清洗阈值

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyDDoSThreshold
<公共请求参数>

{
  "Threshold": 1000,
  "Id": "bgpip-000000xe",
  "Business": "bgpip"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Success": {
      "Code": "Success",
      "Message": "Success"
    }
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

防护配置相关接口

新建CC四层黑白名单

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

新建CC四层黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateCcBlackWhitelist。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
IpList.N	是	Array of IpSegment	IP列表
Type	是	String	IP类型，取值[black(黑名单IP), white(白名单IP)] 示例值：black
Ip	是	String	Ip地址 示例值：1.1.1.1
Domain	是	String	域名 示例值：www.xx.com
Protocol	是	String	协议 示例值：http

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 新建CC四层黑白名单

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCcBlackWhiteIpList
<公共请求参数>
```

```
{
  "Domain": "www.test.com",
  "Protocol": "http",
  "InstanceId": "bgpip-00001001",
  "Ip": "1.1.1.1",
  "IpList": [
    {
      "Ip": "1.1.1.0",
      "Mask": 24
    }
  ],
  "Type": "black"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

新建CC防护的地域封禁配置

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

新建CC防护的地域封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateCcGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例id 示例值：bgpip-0000011x
IP	是	String	ip地址 示例值：1.1.1.1
Domain	是	String	域名 示例值：www.qq.com
Protocol	是	String	协议类型 示例值：http
CcGeoIPBlockConfig	是	CcGeoIPBlockConfig	CC区域封禁配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加CC防护的区域封禁配置

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCcGeoIPBlockConfig
<公共请求参数>
```

```
{
  "InstanceId": "bgpip-0000011x",
  "IP": "1.1.1.1",
  "Domain": "www.test.com",
  "Protocol": "http",
  "CcGeoIPBlockConfig": {
    "Action": "drop",
    "Id": "ccGeoip-00000111",
    "RegionType": "oversea",
    "AreaList": [
      100025
    ]
  }
}
```

输出示例

```
{
  "Response": {
    "RequestId": "54a41bf4-dca0-438b-a1cd-eb1f99e9a6b3"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
UnsupportedOperation	操作不支持。

设置DDoS防护的AI防护开关

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

设置DDoS防护的AI防护开关

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDDoSAI。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceIdList.N	是	Array of String	资源实例ID列表 示例值：["bgpip-0000011x\r\n"]
DDoSAI	是	String	AI防护开关，取值[on(开启) off(关闭)] 示例值：on

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 设置DDoS防护的AI防护开关

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDDoSAI
&DDoSAI=on
&InstanceIdList.0=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "01729f06-522c-4e97-9443-fb5492ab5dbf"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。

添加DDoS防护的IP网段黑白名单

最近更新时间：2025-04-25 01:07:58

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的IP网段黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDDoSBlackWhiteIpList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
IpList.N	是	Array of IpSegment	IP列表 示例值：[{"Ip":"1.1.1.1\r\n", "Mask":0}]
Type	是	String	IP类型，取值[black(黑名单IP), white(白名单IP)] 示例值：black

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的IP黑白名单

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDDoSBlackWhiteIpList
&InstanceId=bgpip-00001001
&IpList.0.Ip=1.1.1.1
&IpList.0.Mask=0
&Type=black
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

配置DDoS连接抑制选项

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

配置DDoS连接抑制选项

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDDoSConnectLimit。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例Id 示例值：bgpip-00000001
ConnectLimitConfig	是	ConnectLimitConfig	连接抑制配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置DDoS连接抑制选项

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDDoSConnectLimit
&InstanceId=bgpip-00000001
&ConnectLimitConfig.SdNewLimit=0
&ConnectLimitConfig.DstNewLimit=0
&ConnectLimitConfig.SdConnLimit=0
&ConnectLimitConfig.DstConnLimit=0
&ConnectLimitConfig.BadConnThreshold=0
&ConnectLimitConfig.NullConnEnable=0
&ConnectLimitConfig.ConnTimeout=0
&ConnectLimitConfig.SynRate=0
```

```
&ConnectLimitConfig.SynLimit=0
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。

错误码	描述
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

添加DDoS防护的区域封禁配置

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的区域封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDDoSGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSGeoIPBlockConfig	是	DDoSGeoIPBlockConfig	DDoS区域封禁配置，填写参数时配置ID请为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的区域封禁配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDDoSGeoIPBlockConfig
&InstanceId=bgpip-0000011x
&DDoSGeoIPBlockConfig.RegionType=customized
&DDoSGeoIPBlockConfig.AreaList.0=100001
&DDoSGeoIPBlockConfig.AreaList.1=100002
&DDoSGeoIPBlockConfig.Action=drop
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

添加DDoS防护的访问限速配置

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的访问限速配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDDoSSpeedLimitConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSSpeedLimitConfig	是	DDoSSpeedLimitConfig	访问限速配置，填写参数时配置ID请为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的访问限速配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDDoSSpeedLimitConfig
&InstanceId=bgpip-0000011x
&DDoSSpeedLimitConfig.Mode=1
&DDoSSpeedLimitConfig.ProtocolList=TCP;UDP
&DDoSSpeedLimitConfig.DstPortList=80;443;1000-2000
&DDoSSpeedLimitConfig.SpeedValues.0.Type=1
&DDoSSpeedLimitConfig.SpeedValues.0.Value=1000
&DDoSSpeedLimitConfig.SpeedValues.1.Type=2
&DDoSSpeedLimitConfig.SpeedValues.1.Value=30000000
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

添加DDoS防护的特征过滤规则

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的特征过滤规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreatePacketFilterConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
PacketFilterConfig	是	PacketFilterConfig	特征过滤规则

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的特征过滤规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreatePacketFilterConfig
<公共请求参数>

{
  "InstanceId": "bgpip-000002q2",
  "PacketFilterConfig": {
    "Action": "drop_rst",
    "Depth": 4,
    "DportEnd": 18373,
```

```
"DportStart": 18373,
"IsNot": 0,
"MatchBegin": "begin_l4",
"MatchType": "sunday",
"Offset": 14,
"PktlenMax": 100,
"PktlenMin": 1,
"Protocol": "tcp",
"SportEnd": 65535,
"SportStart": 1,
"Str": "\\x05ac"
}
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceInUse	资源被占用。

错误码	描述
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

添加DDoS防护的端口acl策略

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的端口acl策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreatePortAclConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
AclConfig	是	AclConfig	端口acl策略

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的端口acl策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreatePortAclConfig
&InstanceId=bgpip-00001001
&AclConfig.ForwardProtocol=tcp
&AclConfig.DPortStart=0
&AclConfig.DPortEnd=65535
&AclConfig.SPortStart=0
&AclConfig.SPortEnd=65535
&AclConfig.Action=drop
&AclConfig.Priority=10
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

批量添加DDoS防护的端口acl策略

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

批量添加DDoS防护的端口acl策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreatePortAclConfigList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceIdList.N	是	Array of String	资源实例ID列表 示例值：["bgpip-00001001"]
AclConfig	是	AclConfig	端口acl策略

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 批量添加DDoS防护的端口acl策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreatePortAclConfigList
&InstanceIdList.0=bgpip-00001001
&AclConfig.ForwardProtocol=tcp
&AclConfig.DPortStart=0
&AclConfig.DPortEnd=65535
&AclConfig.SPortStart=0
&AclConfig.SPortEnd=65535
&AclConfig.Action=drop
&AclConfig.Priority=10
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

设置DDoS防护的协议封禁配置

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

设置DDoS防护的协议封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateProtocolBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
ProtocolBlockConfig	是	ProtocolBlockConfig	协议封禁配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 设置DDoS防护的协议封禁配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateProtocolBlockConfig
&InstanceId=bgpip-0000011x
&ProtocolBlockConfig.DropTcp=1
&ProtocolBlockConfig.DropUdp=1
&ProtocolBlockConfig.DropIcmp=1
&ProtocolBlockConfig.DropOther=1
&ProtocolBlockConfig.CheckExceptNullConnect=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

添加DDoS防护的水印防护配置

最近更新时间：2025-04-25 01:07:57

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的水印防护配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateWaterPrintConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
WaterPrintConfig	是	WaterPrintConfig	水印防护配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的水印防护配置

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateWaterPrintConfig
<公共请求参数>

{
  "InstanceId": "bgp-111111aj",
  "WaterPrintConfig": {
    "Keys": [
      {
        "KeyVersion": "fbf1",
```

```
"KeyContent": "82e352b956e8d512-fbf1-82e352b956e8d512eeb95315d5772e86fac99256",
"KeyId": "000005hj",
"KeyOpenStatus": 0,
"CreateTime": "2020-09-22 00:00:00"
},
"Verify": "checkall",
"Listeners": [
{
"ForwardProtocol": "TCP",
"FrontendPort": 88,
"FrontendPortEnd": 88
}
],
"OpenStatus": 0,
"Offset": 5
}
}
```

输出示例

```
{
  "Response": {
    "RequestId": "512b8c34-3df9-448a-ae90-c28643d351bf"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

添加DDoS防护的水印防护密钥

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加DDoS防护的水印防护密钥

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateWaterPrintKey。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加DDoS防护的水印防护密钥

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateWaterPrintKey
&InstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "512b8c34-3df9-448a-ae90-c28643d351bf"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

删除CC四层黑白名单

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC四层黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCcBlackWhitelpList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
PolicyId	是	String	策略Id 示例值：ccBlackWhite-00000312

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC四层黑白名单

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DeleteCcBlackWhiteIpList
<公共请求参数>

{
  "InstanceId": "bgpip-111111qq",
  "PolicyId": "ccBlackWhite-000000s0"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnsupportedOperation	操作不支持。

删除CC防护的区域封禁配置

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC防护的区域封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCcGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
CcGeoIPBlockConfig	是	CcGeoIPBlockConfig	CC区域封禁配置，填写参数时配置ID不能为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC防护的区域封禁配置

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DeleteCcGeoIPBlockConfig
<公共请求参数>

{
  "InstanceId": "bgpip-000000q0",
  "CcGeoIPBlockConfig": {
    "Action": "drop",
    "Id": "ccGeoip-000000mm",
    "RegionType": "china",
```

```
"AreaList": [
  0
]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "efc47bf5-b16a-410d-a993-0d4584446b6w"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

删除DDoS防护的IP网段黑白名单

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的IP网段黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteDDoSBlackWhiteIpList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
IpList.N	是	Array of IpSegment	IP列表 示例值：[{"ip":"1.1.1.1","Mask":0}]
Type	是	String	IP类型，取值[black(黑名单IP), white(白名单IP)] 示例值：black

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的IP黑白名单

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteDDoSBlackWhiteIpList
&InstanceId=bgpip-00001001
&IpList.0.Ip=1.1.1.1
&IpList.0.Mask=0
&Type=black
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。

删除DDoS防护的区域封禁配置

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的区域封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteDDoSGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSGeoIPBlockConfig	是	DDoSGeoIPBlockConfig	DDoS区域封禁配置，填写参数时配置ID不能为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的区域封禁配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteDDoSGeoIPBlockConfig
&InstanceId=bgpip-0000011x
&DDoSGeoIPBlockConfig.Id=00000001
&DDoSGeoIPBlockConfig.RegionType=customized
&DDoSGeoIPBlockConfig.AreaList.0=100001
&DDoSGeoIPBlockConfig.AreaList.1=100002
&DDoSGeoIPBlockConfig.Action=drop
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceNotFound	资源不存在。

删除DDoS防护的访问限速配置

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的访问限速配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteDDoSSpeedLimitConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSSpeedLimitConfig	是	DDoSSpeedLimitConfig	访问限速配置，填写参数时配置ID不能为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的访问限速配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteDDoSSpeedLimitConfig
&InstanceId=bgpip-0000011x
&DDoSSpeedLimitConfig.Id=00000001
&DDoSSpeedLimitConfig.Mode=1
&DDoSSpeedLimitConfig.ProtocolList=TCP;UDP
&DDoSSpeedLimitConfig.DstPortList=80;443;1000-2000
&DDoSSpeedLimitConfig.SpeedValues.0.Type=1
&DDoSSpeedLimitConfig.SpeedValues.0.Value=1000
&DDoSSpeedLimitConfig.SpeedValues.1.Type=2
&DDoSSpeedLimitConfig.SpeedValues.1.Value=30000000
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
InvalidParameterValue	参数取值错误。
ResourceNotFound	资源不存在。

删除DDoS防护的特征过滤规则

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的特征过滤规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeletePacketFilterConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
PacketFilterConfig	是	PacketFilterConfig	特征过滤配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的特征过滤规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DeletePacketFilterConfig
<公共请求参数>

{
  "InstanceId": "bgpip-111111q1",
  "PacketFilterConfig": {
    "MatchType2": "",
    "MatchBegin2": "no_match",
    "Str2": "",
```

```
"SportEnd": 0,
"IsNot": 0,
"PktlenMax": 0,
"MatchLogic": "none",
"MatchBegin": "begin_l4",
"Offset": 0,
"SportStart": 0,
"DportStart": 0,
"PktlenMin": 0,
"IsNot2": 0,
"Depth": 0,
"Str": "\\x3908",
"Action": "drop",
"Protocol": "all",
"MatchType": "sunday",
"DportEnd": 0,
"Offset2": 0,
"Depth2": 0,
"Id": "01j9d4hi"
}
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。

删除DDoS防护的端口acl策略

最近更新时间：2025-04-25 01:07:56

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的端口acl策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeletePortAclConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
AclConfig	是	AclConfig	端口acl策略

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的端口acl策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeletePortAclConfig
&InstanceId=bgpip-00001001
&AclConfig.ForwardProtocol=tcp
&AclConfig.DPortStart=0
&AclConfig.DPortEnd=65535
&AclConfig.SPortStart=0
&AclConfig.SPortEnd=65535
&AclConfig.Action=drop
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除DDoS防护的水印防护配置

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的水印防护配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteWaterPrintConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的水印防护配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteWaterPrintConfig
&InstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "512b8c34-3df9-448a-ae90-c28643d351bf"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。

删除DDoS防护的水印防护密钥

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除DDoS防护的水印防护密钥

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteWaterPrintKey。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
KeyId	是	String	水印密钥ID 示例值：000000f8

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除DDoS防护的水印防护密钥

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteWaterPrintKey
&InstanceId=bgpip-0000011x
&KeyId=000000f8
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "512b8c34-3df9-448a-ae90-c28643d351bf"
```

```
}  
  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

获取CC四层黑白名单列表

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC四层黑白名单列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCcBlackWhitelistList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp-multip：表示高防包；bgpip：表示高防IP） 示例值：bgp-multip
InstanceId	是	String	指定特定实例Id 示例值：bgpip-0000011x
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：1
Limit	是	Integer	一页条数 示例值：1
Ip	否	String	IP地址，普通高防IP要传该字段 示例值：1.1.1.1
Domain	否	String	域名，普通高防IP要传该字段 示例值：www.xx.com
Protocol	否	String	协议，普通高防IP要传该字段 示例值：http
FilterIp	否	String	筛选IP，需要筛选黑白名单IP时传该字段 示例值：1.1.1.1
FilterType	否	String	黑白名单筛选字段，需要筛选黑白名单列表时传该字段 示例值：black

3. 输出参数

参数名称	类型	描述
Total	Integer	CC四层黑白名单策略列表总数 示例值：1

参数名称	类型	描述
CcBlackWhiteIpList	Array of CcBlackWhiteIpPolicy	CC四层黑白名单策略列表详情
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC四层黑白名单列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCcBlackWhiteIpList
<公共请求参数>

{
  "Domain": "www.test1.com",
  "Protocol": "https",
  "Business": "bgpip",
  "InstanceId": "bgpip-000000z9",
  "Ip": "1.1.1.1",
  "Limit": 1,
  "Offset": 1
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "CcBlackWhiteIpList": [
      {
        "Domain": "www.test1.com",
        "BlackWhiteIp": "2.2.2.2",
        "Protocol": "https",
        "InstanceId": "bgpip-000004z9",
        "Ip": "1.1.1.1",
        "Mask": 1,
        "PolicyId": "ccBlackWhite-000002s7",
        "ModifyTime": "2022-04-30 08:09:20",
        "Type": "white",
        "CreateTime": "2022-04-30 07:09:20"
      }
    ],
    "RequestId": "1be9f39c-7060-11ed-aae2-525400f921f0"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceUnavailable	资源不可用。
UnsupportedOperation	操作不支持。

获取CC防护的区域封禁配置列表

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC防护的区域封禁配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCcGeoIPBlockConfigList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip-multip：表示高防包；bgpip：表示高防IP） 示例值：bgpip-multip
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：1
Limit	是	Integer	一页条数 示例值：1
InstanceId	否	String	指定特定实例ID 示例值：bgpip-0000011x
Ip	否	String	IP地址，普通高防IP要传该字段 示例值：1.1.1.1
Domain	否	String	域名，普通高防IP要传该字段 示例值：www.xx.com
Protocol	否	String	协议，普通高防IP要传该字段 示例值：http

3. 输出参数

参数名称	类型	描述
Total	Integer	CC地域封禁策略列表总数 示例值：1
CcGeoIpPolicyList	Array of CcGeoIpPolicyNew	CC地域封禁策略列表详情
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC防护的区域封禁配置列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCcGeoIPBlockConfigList
<公共请求参数>
```

```
{
  "Domain": "test.dayu.com",
  "Protocol": "http",
  "Business": "bgpip",
  "InstanceId": "bgpip-111111q1",
  "Ip": "1.1.1.1",
  "Limit": 1,
  "Offset": 1
}
```

输出示例

```
{
  "Response": {
    "CcGeoIpPolicyList": [
      {
        "Domain": "test.dayu.com",
        "Protocol": "http",
        "InstanceId": "bgpip-111111q1",
        "Ip": "1.1.1.1",
        "RegionType": "china",
        "PolicyId": "rule-0000011x",
        "ModifyTime": "2020-09-22 00:00:00",
        "Action": "1",
        "CreateTime": "2020-09-22 00:00:00",
        "AreaList": [
          1
        ]
      }
    ],
    "Total": 1,
    "RequestId": "c9520936-381c-4c04-b535-c8f4df1f8059"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

获取DDoS防护的IP网段黑白名单

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的IP网段黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeDDoSBlackWhiteIpList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001

3. 输出参数

参数名称	类型	描述
BlackIpList	Array of IpSegment	黑名单IP列表 示例值：[{"Ip": "123.123.123.1", "Mask": 0}]
WhiteIpList	Array of IpSegment	白名单IP列表 示例值：[{"Ip": "123.123.123.2", "Mask": 0}]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的IP黑白名单

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeDDoSBlackWhiteIpList
&InstanceId=bgpip-00001001
&<公共请求参数>
```

输出示例

```
{
  "Response": {
```

```
"RequestId": "0f996729-bb5b-49da-89c3-8fc2a7400bd5",
"BlackIpList": [
{
  "Ip": "123.123.123.1",
  "Mask": 0
},
],
"WhiteIpList": [
{
  "Ip": "123.123.123.2",
  "Mask": 0
},
]
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。

错误码	描述
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DDoS连接抑制配置列表

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

获取DDoS连接抑制配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeDDoSConnectLimitList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数 示例值：10
FilterIp	否	String	可选参数，按照IP进行过滤 示例值："1.1.1.1"
FilterInstanceId	否	String	可选参数，按照实例id进行过滤 示例值："id-xxxx"

3. 输出参数

参数名称	类型	描述
Total	Integer	连接抑制配置总数 示例值：1
ConfigList	Array of ConnectLimitRelation	连接抑制配置详情信息
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS连接抑制配置列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeDDoSConnectLimitList
<公共请求参数>

{
  "Limit": 10,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "ConfigList": [
      {
        "InstanceDetailList": [
          {
            "InstanceId": "bgpip-111111q1",
            "EipList": [
              "1.1.1.1"
            ]
          }
        ],
        "ConnectLimitConfig": {
          "SynLimit": 1,
          "BadConnThreshold": 1,
          "SdConnLimit": 1,
          "SdNewLimit": 1,
          "SynRate": 1,
          "ConnTimeout": 1,
          "DstConnLimit": 1,
          "DstNewLimit": 1,
          "NullConnEnable": 1
        }
      }
    ],
    "RequestId": "c9520936-381c-4c04-b535-c8f4df1f8059"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DDoS防护的IP黑白名单列表

最近更新时间：2025-04-25 01:07:55

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的IP黑白名单列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListBlackWhitelplist。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstancelid	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
Filterip	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：2
IpList	Array of BlackWhitelplistRelation	黑白IP列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的IP黑白名单列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeListBlackWhiteIpList
<公共请求参数>

{
  "FilterInstanceId": "bgpip-0000011x",
  "FilterIp": "1.1.1.1",
  "Limit": 25,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "IpList": [
      {
        "InstanceDetailList": [
          {
            "InstanceId": "bgpip-0000011x",
            "EipList": [
              "1.2.2.19"
            ]
          }
        ],
        "Ip": "1.1.4.1",
        "Type": "white",
        "Mask": 1,
        "ModifyTime": "2022-11-29 18:34:13"
      }
    ],
    "Total": 1,
    "RequestId": "j5RV4a5576e8cf46f24e3bae026ebb391a"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的AI防护开关列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

获取DDoS防护的AI防护开关列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListDDoSAI。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceId	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of DDoSAIRelation	AI防护开关列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的AI防护开关列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListDDoSAI
&Offset=0
```

```
&Limit=25
&FilterIp=1.1.1.1
&FilterInstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "c8c3f585-dd94-4d99-b877-f2c612445630",
    "ConfigList": [
      {
        "DDoSAI": "off",
        "InstanceDetailList": [
          {
            "EipList": [
              "1.1.1.1"
            ],
            "InstanceId": "bgpip-0000011y"
          }
        ]
      }
    ],
    "Total": 1
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的区域封禁配置列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的区域封禁配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListDDoSGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceId	是	String	资源实例ID搜索，支持资源实例前缀通配搜索，例如bgp-表示获取高防包类型的资源实例 示例值：bgpip-
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of DDoSGeoIPBlockConfigRelation	DDoS区域封禁配置列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的区域封禁配置列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListDDoSGeoIPBlockConfig
&Offset=0
&Limit=25
&FilterIp=1.1.1.1
&FilterInstanceId=bgpip-*
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "bcfa39ae-5b8b-4b5b-9e53-016176e54b59",
    "ConfigList": [
      {
        "GeoIPBlockConfig": {
          "RegionType": "customized",
          "AreaList": [
            100001,
            100002
          ],
          "Action": "drop",
          "Id": "00000001"
        },
        "InstanceDetailList": [
          {
            "EipList": [
              "1.1.1.1"
            ],
            "InstanceId": "bgpip-0000011x"
          }
        ]
      }
    ],
    "Total": 1
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的访问限速配置列表

最近更新时间：2025-04-25 01:07:54

- - - - -

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的访问限速配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListDDoSSpeedLimitConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceld	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-表示获取高防包类型的资源实例 示例值：bgpip-
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of DDoSSpeedLimitConfigRelation	访问限速配置列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的访问限速配置列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListDDoSSpeedLimitConfig
&Offset=0
&Limit=25
&FilterIp=1.1.1.1
```

```
&FilterInstanceId=bgpip-*  
&<公共请求参数>
```

输出示例

```
{  
  "Response": {  
    "RequestId": "bcfa39ae-5b8b-4b5b-9e53-016176e54b59",  
    "ConfigList": [  
      {  
        "SpeedLimitConfig": {  
          "Mode": 1,  
          "DstPortList": "0-65535",  
          "ProtocolList": "ALL",  
          "SpeedValues": [  
            {  
              "Type": 1,  
              "Value": 1200  
            }  
          ],  
          "Id": "00000001"  
        },  
        "InstanceDetailList": [  
          {  
            "EipList": [  
              "1.1.1.1"  
            ],  
            "InstanceId": "bgpip-0000011x"  
          }  
        ],  
        "Total": 1  
      }  
    ]  
  }  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的特征过滤规则列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的特征过滤规则列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListPacketFilterConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceld	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of PacketFilterRelation	特征过滤配置
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的特征过滤规则列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeListPacketFilterConfig
<公共请求参数>
```

```
{
  "FilterInstanceId": "bgpip-0000011x",
  "FilterIp": "1.1.1.1",
  "Limit": 25,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "ConfigList": [
      {
        "InstanceDetailList": [
          {
            "InstanceId": "bgpip-0000011x",
            "EipList": [
              "1.1.1.1"
            ]
          }
        ],
        "PacketFilterConfig": {
          "MatchType2": "sunday",
          "MatchBegin2": "begin-15",
          "Str2": "",
          "SportEnd": 0,
          "IsNot": 0,
          "PktlenMax": 1500,
          "MatchLogic": "none",
          "MatchBegin": "begin-15",
          "Offset": 0,
          "SportStart": 0,
          "DportStart": 0,
          "PktlenMin": 0,
          "IsNot2": 0,
          "Depth": 20,
          "Str": "",
          "Action": "drop",
          "Protocol": "tcp",
          "MatchType": "sunday",
          "PktLenGT": 50,
          "DportEnd": 0,
          "Offset2": 5,
          "Depth2": 20,
          "Id": ""
        },
        "ModifyTime": "2022-01-29 17:47:48"
      }
    ]
  }
}
```

```
},
  "RequestId": "3eb91f01-96dd-4ad3-9e16-edde46b8cf40"
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的端口acl策略列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的端口acl策略列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListPortAclList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：10
FilterInstanceld	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterIp	否	String	ip搜索 示例值：1.1.1.3

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：2
AclList	Array of AclConfigRelation	端口acl策略
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的端口acl策略列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListPortAclList
&Offset=0
```

```
&Limit=10
&FilterInstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "83978459-3a66-46e7-bb97-c946656d16eb",
    "AclList": [
      {
        "AclConfig": {
          "ForwardProtocol": "tcp",
          "DPortStart": 0,
          "DPortEnd": 65535,
          "SPortStart": 0,
          "SPortEnd": 65535,
          "Action": "drop",
          "Priority": 10
        },
        "InstanceDetailList": [
          {
            "EipList": [
              "1.2.2.19"
            ],
            "InstanceId": "bgpip-0000011x"
          }
        ]
      }
    ],
    "Total": 2
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DDoS防护的协议封禁配置列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的协议封禁配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListProtocolBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceld	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of ProtocolBlockRelation	协议封禁配置
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的协议封禁配置列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListProtocolBlockConfig
&Offset=0
```

```
&Limit=25
&FilterIp=1.1.1.1
&FilterInstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "70a4b0a9-4dc9-4a29-ae47-ce1c498b176a",
    "ConfigList": [
      {
        "ProtocolBlockConfig": {
          "DropTcp": 0,
          "DropUdp": 0,
          "DropIcmp": 0,
          "DropOther": 0,
          "CheckExceptNullConnect": 0
        },
        "InstanceDetailList": [
          {
            "EipList": [
              "1.1.1.1"
            ],
            "InstanceId": "bgpip-0000011x"
          }
        ]
      }
    ],
    "Total": 1
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS防护的水印防护配置列表

最近更新时间：2025-04-25 01:07:54

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取DDoS防护的水印防护配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListWaterPrintConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceld	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterIp	否	String	IP搜索 示例值：1.1.1.1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
ConfigList	Array of WaterPrintRelation	水印配置列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS防护的水印防护配置列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeListWaterPrintConfig
<公共请求参数>
```

```
{
  "FilterInstanceId": "bgpip-0000011x",
  "FilterIp": "1.1.1.1",
  "Limit": 25,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "ConfigList": [
      {
        "WaterPrintConfig": {
          "Keys": [
            {
              "KeyVersion": "fbf1",
              "KeyContent": "82e352b956e8d512-fbf1-82e352b956e8d512eeb95315d5772e86fac99256",
              "KeyId": "000005hj",
              "KeyOpenStatus": 0,
              "CreateTime": "2020-09-22 00:00:00"
            }
          ],
          "Verify": "checkall",
          "Listeners": [
            {
              "ForwardProtocol": "TCP",
              "FrontendPort": 80,
              "FrontendPortEnd": 0
            }
          ],
          "CloudSdkProxy": 1,
          "OpenStatus": 0,
          "Offset": 30
        },
        "InstanceDetailList": [
          {
            "InstanceId": "bgpip-0000011x",
            "EipList": [
              "1.1.1.1"
            ]
          }
        ]
      }
    ],
    "RequestId": "c5ff0d5d-d2bd-4b34-b081-4acd4cf7be1f"
  }
}
```

```
}  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

修改CC四层黑白名单

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改CC四层黑白名单

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyCcBlackWhiteIpList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
IpList.N	是	Array of IpSegment	IP列表
Type	是	String	IP类型，取值[black(黑名单IP), white(白名单IP)] 示例值：black
PolicyId	是	String	策略Id 示例值：ccBlackWhite-00000312

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改CC四层黑白名单

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyCcBlackWhiteIpList
<公共请求参数>

{
```

```
"InstanceId": "bgpip-111111q1",
"IpList": [
{
  "Ip": "1.1.1.1",
  "Mask": 1
}
],
"Type": "black",
"PolicyId": "ccBlackWhite-00000312"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
LimitExceeded	超过配额限制。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

修改DDoS黑白名单列表

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

修改DDoS黑白名单列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDDoSBlackWhitelplist。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源Id 示例值：bgpip-00023423
OldIpType	是	String	当前配置的黑白名单类型，取值black时表示黑名单；取值white时表示白名单 示例值：black
OldIp	是	IpSegment	当前配置的Ip段，包含ip与掩码
NewIpType	是	String	修改后黑白名单类型，取值black时黑名单，取值white时白名单 示例值：black
NewIp	是	IpSegment	当前配置的Ip段，包含ip与掩码

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS黑白名单列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyDDoSBlackWhiteIpList
<公共请求参数>
```

```
{
  "InstanceId": "bgpip-00023423",
  "OldIp": {
    "Ip": "1.1.1.1",
    "Mask": "0"
  },
  "NewIp": {
    "Ip": "1.1.1.2",
    "Mask": "0"
  },
  "NewIpType": "black",
  "OldIpType": "black"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "63591603-c943-46eb-9b78-c5498072291a"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

修改DDoS防护的区域封禁配置

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改DDoS防护的区域封禁配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDDoSGeoIPBlockConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSGeoIPBlockConfig	是	DDoSGeoIPBlockConfig	DDoS区域封禁配置，填写参数时配置ID不能为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS防护的区域封禁配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyDDoSGeoIPBlockConfig
&InstanceId=bgpip-0000011x
&DDoSGeoIPBlockConfig.Id=00000001
&DDoSGeoIPBlockConfig.RegionType=customized
&DDoSGeoIPBlockConfig.AreaList.0=100001
&DDoSGeoIPBlockConfig.AreaList.1=100002
&DDoSGeoIPBlockConfig.Action=drop
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceNotFound	资源不存在。

修改DDoS防护的访问限速配置

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改DDoS防护的访问限速配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDDoSSpeedLimitConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
DDoSSpeedLimitConfig	是	DDoSSpeedLimitConfig	访问限速配置，填写参数时配置ID不能为空

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS防护的访问限速配置

ModifyDDoSSpeedLimitConfig

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyDDoSSpeedLimitConfig
&InstanceId=bgpip-0000011x
&DDoSSpeedLimitConfig.Id=00000001
&DDoSSpeedLimitConfig.Mode=1
&DDoSSpeedLimitConfig.ProtocolList=TCP;UDP
&DDoSSpeedLimitConfig.DstPortList=80;443;1000-2000
&DDoSSpeedLimitConfig.SpeedValues.0.Type=1
&DDoSSpeedLimitConfig.SpeedValues.0.Value=1000
&DDoSSpeedLimitConfig.SpeedValues.1.Type=2
```

```
&DDoSSpeedLimitConfig.SpeedValues.1.Value=30000000
```

```
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceNotFound	资源不存在。

修改DDoS防护的特征过滤规则

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改DDoS防护的特征过滤规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyPacketFilterConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
PacketFilterConfig	是	PacketFilterConfig	特征过滤配置

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS防护的特征过滤规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyPacketFilterConfig
<公共请求参数>

{
  "InstanceId": "bgpip-00000001",
  "PacketFilterConfig": {
    "Action": "drop_rst",
    "Depth": 4,
    "Depth2": 4,
```

```
"DportEnd": 8101,
"DportStart": 8101,
"Id": "0176fm61",
"IsNot": 0,
"IsNot2": 0,
"MatchBegin": "begin_l4",
"MatchBegin2": "begin_l4",
"MatchLogic": "and",
"MatchType": "sunday",
"MatchType2": "pcre",
"Offset": 14,
"Offset2": 14,
"PktlenMax": 100,
"PktlenMin": 1,
"Protocol": "tcp",
"SportEnd": 65535,
"SportStart": 1,
"Str": "\\x3908",
"Str2": "\\xfaf0"
}
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。

修改DDoS防护的端口acl策略

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改DDoS防护的端口acl策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyPortAclConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-00001001
OldAclConfig	是	AclConfig	旧端口acl策略
NewAclConfig	是	AclConfig	新端口acl策略

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改DDoS防护的端口acl策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyPortAclConfig
&InstanceId=bgpip-00001001
&OldAclConfig.ForwardProtocol=tcp
&OldAclConfig.DPortStart=0
&OldAclConfig.DPortEnd=65535
&OldAclConfig.SPortStart=0
&OldAclConfig.SPortEnd=65535
&OldAclConfig.Action=drop
&OldAclConfig.Priority=5
&NewAclConfig.ForwardProtocol=udp
```

```
&NewAclConfig.DPortStart=0
&NewAclConfig.DPortEnd=65535
&NewAclConfig.SPortStart=0
&NewAclConfig.SPortEnd=65535
&NewAclConfig.Action=drop
&NewAclConfig.Priority=5
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。

错误码	描述
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

开启或关闭DDoS防护的水印防护配置

最近更新时间：2025-04-25 01:07:53

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

开启或关闭DDoS防护的水印防护配置，此功能为付费增值服务，有需求请联系售后

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：SwitchWaterPrintConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	资源实例ID 示例值：bgpip-0000011x
OpenStatus	是	Integer	水印开启/关闭状态，1表示开启；0表示关闭 示例值：1
CloudSdkProxy	否	Integer	是否开启代理，1开启则忽略IP+端口校验；0关闭则需要IP+端口校验 示例值：1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 开启或关闭DDoS防护的水印防护配置

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: SwitchWaterPrintConfig
<公共请求参数>

{
  "InstanceId": "bgpip-0000011x",
```

```
"OpenStatus": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "512b8c34-3df9-448a-ae90-c28643d351bf"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
InternalServerError	内部错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

告警通知相关接口

设置单IP默认告警阈值配置

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

设置单IP默认告警阈值配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateDefaultAlarmThreshold。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
DefaultAlarmConfig	是	DefaultAlarmThreshold	默认告警阈值配置
InstanceType	是	String	产品类型，取值[bgp(表示高防包产品) bgpip(表示高防IP产品)] 示例值：bgp

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 设置单IP默认告警阈值配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateDefaultAlarmThreshold
&InstanceType=bgp
&DefaultAlarmConfig.AlarmType=1
&DefaultAlarmConfig.AlarmThreshold=2000
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "b7739a1e-837d-4248-bf9f-16a9bf77db22"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。

设置单IP告警阈值配置

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

设置单IP告警阈值配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateIPAlarmThresholdConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
IpAlarmThresholdConfigList.N	是	Array of IPAlarmThresholdRelation	IP告警阈值配置列表

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 设置单IP告警阈值配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateIPAlarmThresholdConfig
&IpAlarmThresholdConfigList.0.AlarmType=1
&IpAlarmThresholdConfigList.0.AlarmThreshold=1000
&IpAlarmThresholdConfigList.0.InstanceDetailList.0.EipList.0=1.1.1.1
&IpAlarmThresholdConfigList.0.InstanceDetailList.0.InstanceId=bgpip-0000011x
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
```

```
}  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。

获取单IP默认告警阈值配置

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取单IP默认告警阈值配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeDefaultAlarmThreshold。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceType	是	String	产品类型，取值[bgp(表示高防包产品) bgpip(表示高防IP产品)] 示例值：bgp
FilterAlarmType	是	Integer	告警阈值类型搜索，取值[1(入流量告警阈值) 2(攻击清洗流量告警阈值)] 示例值：1

3. 输出参数

参数名称	类型	描述
DefaultAlarmConfigList	Array of DefaultAlarmThreshold	默认告警阈值配置
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取单IP默认告警阈值配置

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeDefaultAlarmThreshold
&InstanceType=bgp
```

```
&FilterAlarmType=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "a7df6626-fb01-4c9e-b43a-7af1db3fe998",
    "DefaultAlarmConfigList": [
      {
        "AlarmType": 1,
        "AlarmThreshold": 1000
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取单IP告警阈值配置列表

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取单IP告警阈值配置列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListIPAlarmConfig。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为100;最大取值为100 示例值：25
FilterInstanceId	是	String	资源实例ID搜索, 支持资源实例前缀通配搜索，例如bgp-*表示获取高防包类型的资源实例 示例值：bgpip-0000011x
FilterAlarmType	是	Integer	告警阈值类型搜索，取值[1(入流量告警阈值) 2(攻击清洗流量告警阈值)] 示例值：1
FilterIp	否	String	IP搜索 示例值：1.1.1.1
FilterCname	否	String	高防IP实例资源的cname 示例值：abcdefg

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：2
ConfigList	Array of IPAlarmThresholdRelation	IP告警阈值配置列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取单IP告警阈值配置列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeListIPAlarmConfig
&Offset=0
&Limit=25
&FilterIp=1.1.1.1
&FilterInstanceId=bgpip-0000011x
&FilterAlarmType=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "a3056c16-1cb6-4bdb-9d3b-a2361fe5b63c",
    "ConfigList": [
      {
        "AlarmType": 2,
        "AlarmThreshold": 2000,
        "InstanceDetailList": [
          {
            "EipList": [
              "1.1.1.1"
            ],
            "InstanceId": "bgpip-0000011x"
          }
        ],
      },
      {
        "AlarmType": 1,
        "AlarmThreshold": 5000,
        "InstanceDetailList": [
          {
            "EipList": [
              "1.1.1.1"
            ],
            "InstanceId": "bgpip-0000011x"
          }
        ],
      },
      {
        "Total": 2
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

接入配置相关接口

批量配置L7转发规则的证书

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

批量配置L7转发规则的证书供SSL测调用

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateL7RuleCerts。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
CertId	是	String	SSL证书ID 示例值：cMgC0TpV
L7Rules.N	是	Array of InsL7Rules	L7域名转发规则列表

3. 输出参数

参数名称	类型	描述
Success	SuccessCode	成功码
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 证书绑定域名列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateL7RuleCerts
<公共请求参数>

{
  "CertId": "cMgC0TpV",
```

```
"L7Rules": [
{
  "InsId": "bgpip-000001eo",
  "AppId": "251008915",
  "Protocol": "https",
  "Domain": "www.test.com",
  "VirtualPort": "443",
  "SSLId": "",
  "Status": 8
}
]
```

输出示例

```
{
  "Response": {
    "RequestId": "a7d078cc-2547-47f8-b1d6-65bc67d28ae7",
    "Success": {
      "Code": "Success",
      "Message": "Success"
    }
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnknownParameter	未知参数错误。

添加7层转发规则

最近更新时间：2025-04-25 01:08:02

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

添加7层转发规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateNewL7Rules。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Rules.N	是	Array of L7RuleEntry	规则列表
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
IdList.N	是	Array of String	资源ID列表 示例值：["bgpip-000001\r\n"]
VipList.N	是	Array of String	资源IP列表 示例值：["192.168.1.1\r\n"]

3. 输出参数

参数名称	类型	描述
Success	SuccessCode	成功码
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 添加7层转发规则

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateNewL7Rules
<公共请求参数>
```

```
{
  "Rules": [
    {
      "KeepTime": "300",
      "Domain": "www.abc.com",
      "Protocol": "http",
      "SourceType": "2",
      "CertType": "0",
      "LbType": "1",
      "SourceList": [
        {
          "Source": "1.1.1.20",
          "Weight": "100"
        },
        {
          "Source": "1.1.1.10",
          "Weight": "100"
        }
      ],
      "KeepEnable": "0",
      "RuleName": "name"
    }
  ],
  "VipList": [
    "192.168.1.1"
  ],
  "IdList": [
    "bgpip-000001"
  ],
  "Business": "bgpip"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Success": {
      "Code": "Success",
      "Message": "Success"
    }
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

获取L7转发规则健康检查异常结果列表

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取L7转发规则健康检查异常结果列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeNewL7RulesErrHealth。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号(bgpip表示高防IP) 示例值：bgpip
RuleIdList.N	否	Array of String	规则Id列表 示例值：id-xxx

3. 输出参数

参数名称	类型	描述
ErrHealths	Array of KeyValue	异常规则列表，返回值说明：Key值为规则ID，Value值为异常IP及错误信息，多个IP用","分割
Total	Integer	异常规则的总数 示例值：1
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取L7转发规则健康检查异常结果

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeNewL7RulesErrHealth
<公共请求参数>
```

```
{
  "Business": "bgpip"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Total": 1,
    "ErrHealths": [
      {
        "Key": "rule-00000001",
        "Value": "1.1.1.1 400,2.2.2.2 500"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

修改7层转发规则

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改7层转发规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyNewDomainRules。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
Id	是	String	资源ID 示例值：bgpip-000000xe
Rule	是	NewL7RuleEntry	域名转发规则

3. 输出参数

参数名称	类型	描述
Success	SuccessCode	成功码
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改7层转发规则

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyNewDomainRules
&Business=bgpip
&Id=bgpip-000000xe
&Rule.RuleName=name
&Rule.Protocol=http
&Rule.Domain=www.abc.com
&Rule.SourceType=2
&Rule.LbType=1
```

```
&Rule.KeepTime=300
&Rule.SourceList.0.Source=1.1.1.10
&Rule.SourceList.0.Weight=100
&Rule.SourceList.1.Source=1.1.1.20
&Rule.SourceList.1.Weight=100
&Rule.CertType=0
&Rule.VirtualPort=8443
&Rule.KeepEnable=0
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Success": {
      "Code": "Success",
      "Message": "Success"
    }
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
ResourceInUse	资源被占用。
UnsupportedOperation	操作不支持。

智能调度相关接口

创建调度的域名

最近更新时间：2025-04-25 01:08:01

- - - - -

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

创建一个域名，可用于在封堵时调度切换IP

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateSchedulingDomain。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Product	否	String	代表是否混合云本地化的产品。 hybrid: 宙斯盾本地化 不填写：其他 示例值：hybrid

3. 输出参数

参数名称	类型	描述
Domain	String	新创建的域名 示例值：test123.anever.cn
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建调度的域名

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateSchedulingDomain
&Product=hybrid
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Domain": "test123.anever.cn"
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。

获取智能调度域名列表

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取智能调度域名列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeListSchedulingDomain。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：1
Limit	是	Integer	一页条数，当Limit=0时，默认一页条数为20;最大取值为100 示例值：25
FilterDomain	否	String	调度域名搜索 示例值：test.com
Status	否	String	运行状态 0 代表未运行 1 正在运行 2 运行异常 示例值：1

3. 输出参数

参数名称	类型	描述
Total	Integer	总数 示例值：1
DomainList	Array of SchedulingDomainInfo	调度域名信息列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取智能调度域名列表

查询智能调度列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeListSchedulingDomain
<公共请求参数>
```

```
{
  "Limit": 1,
  "FilterDomain": "www.test.com",
  "Offset": 1
}
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "RequestId": "0506d138-ef0f-4ff4-83b0-f1d85e740afd",
    "DomainList": [
      {
        "Status": 1,
        "Domain": "www.test.com",
        "ModifyTime": "2020-09-22 00:00:00",
        "TTL": 1,
        "CreatedTime": "2020-09-22 00:00:00",
        "UsrDomainName": "www.test.com",
        "LineIPList": [
          {
            "Eip": "1.1.1.1",
            "Type": "bgp"
          }
        ],
        "Method": "priority"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

修改智能解析域名名称

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改智能解析域名名称

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyDomainUsrName。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
DomainName	是	String	用户CNAME 示例值：test.com
DomainUserName	是	String	域名名称 示例值：name

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改智能解析域名名称

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyDomainUsrName
&DomainName=test.com
&DomainUserName=name
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
```

```
}  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

黑洞解封相关接口

获取IP封堵列表

最近更新时间：2025-04-25 01:07:52

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取IP封堵列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeIpBlockList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。

3. 输出参数

参数名称	类型	描述
List	Array of IpBlockData	IP封堵列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取IP封堵列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeIpBlockList
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "List": [
      {
        "Status": "1",
        "Ip": "1.1.1.1",
```

```
"UnBlockTime": "2020-09-22 00:00:00",
"ActionType": "type",
"BlockTime": "2020-09-22 00:00:00",
"ProtectFlag": 1
},
"RequestId": "reqid"
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

统计报表相关接口

获取高防包流量折线图

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取高防包流量折线图

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBgpBizTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp-multip表示高防包） 示例值：bgp-multip
StartTime	是	String	统计开始时间。例：“2020-09-22 00:00:00”，注意该时间必须为5分钟的倍数 示例值：2020-09-22 00:00:00
EndTime	是	String	统计结束时间。例：“2020-09-22 00:00:00”，注意该时间必须为5分钟的倍数 示例值：2020-09-22 00:00:00
MetricName	是	String	统计维度，可取值intraffic, outtraffic, inpkg, outpkg；intraffic：入流量 outtraffic：出流量 inpkg：入包速率 outpkg：出包速率 示例值：outtraffic
InstanceId	是	String	资源实例ID 示例值：bgp-xxxxxxxx
Flag	是	Integer	0表示固定时间，1表示自定义时间 示例值：1

3. 输出参数

参数名称	类型	描述
DataList	Array of Integer	曲线图各个时间点的值 示例值：1
Total	Integer	曲线图取值个数 示例值：10
MetricName	String	统计纬度 示例值：intraffic

参数名称	类型	描述
MaxData	Integer	返回数组最大值 示例值：1
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取高防包流量折线图

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBgpBizTrend
<公共请求参数>

{
  "Business": "bgp-multip",
  "InstanceId": "bgp-000000001",
  "Flag": 1,
  "StartTime": "2022-11-29 00:00:00",
  "EndTime": "2022-11-29 01:00:00",
  "MetricName": "intraffice"
}
```

输出示例

```
{
  "Response": {
    "DataList": [
      1
    ],
    "Total": 1,
    "MaxData": 1,
    "RequestId": "f4b3915e-clbd-4ca4-aaef-e88eaa8aa66e",
    "MetricName": "intraffice"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取业务流量状态码统计列表

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取业务流量状态码统计列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBizHttpStatus。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Statistics	是	String	统计方式，仅支持sum 示例值：sum
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
Period	是	Integer	统计周期，可取值60，300，1800，3600，21600，86400，单位秒 示例值：300
StartTime	是	String	统计开始时间。如2020-02-01 12:04:12 示例值：2020-02-01 12:04:12
EndTime	是	String	统计结束时间。如2020-02-03 18:03:23 示例值：2020-02-03 18:03:23
Id	是	String	资源Id 示例值：bgpip-000000xe
Domain	否	String	特定域名查询 示例值：www.xx.com
ProtoInfo.N	否	Array of ProtocolPort	协议及端口列表，协议可取值TCP, UDP, HTTP, HTTPS，仅统计纬度为连接数时有效

3. 输出参数

参数名称	类型	描述
HttpStatusMap	HttpStatusMap	业务流量http状态码统计数据
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取业务流量状态码统计曲线

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBizHttpStatus
<公共请求参数>

{
  "Domain": "www.tencent.com",
  "Statistics": "sum",
  "Business": "bgpip",
  "ProtoInfo": [
    {
      "Protocol": "https",
      "Port": 1
    }
  ],
  "Period": 1,
  "StartTime": "2020-09-22 00:00:00",
  "EndTime": "2020-09-22 00:00:00",
  "Id": "bgpip-00001"
}
```

输出示例

```
{
  "Response": {
    "HttpStatusMap": {
      "SourceHttp2xx": [
        1,
        2,
        3,
        2
      ],
      "Http5xx": [
        1,
        2,
        3,
        2
      ],
      "SourceHttp5xx": [
        1,
        2,
        3,
        2
      ],
      "SourceHttp404": [
        1,
        2,
        3,

```

```
2
],
"Http4xx": [
1,
2,
3,
2
],
"SourceHttp4xx": [
1,
2,
3,
2
],
"Http2xx": [
1,
2,
3,
2
],
"Http404": [
1,
2,
3,
2
],
"SourceHttp3xx": [
1,
2,
3,
2
],
"Http3xx": [
1,
2,
3,
2
]
},
"RequestId": "asdasd-dasdasd"
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnknownParameter	未知参数错误。

获取高防IP业务监控流量曲线

最近更新时间：2025-04-25 01:08:01

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取高防IP业务监控流量曲线

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBizMonitorTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，此参数为可选参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgpip
StartTime	是	Timestamp	统计开始时间。例：“2020-09-22 00:00:00” 示例值：2020-09-22 00:00:00
EndTime	是	Timestamp	统计结束时间。例：“2020-09-22 00:00:00” 示例值：2020-09-22 00:00:00
Id	是	String	资源实例ID 示例值：bgpip-00000
MetricName	是	String	统计纬度，可取值intraffic outtraffic inpkg outpkg 示例值：inpkg
Period	是	Integer	时间粒度 60 300 3600 21600 86400 示例值：60

3. 输出参数

参数名称	类型	描述
DataList	Array of Float	曲线图各个时间点的值 示例值：1.2
MetricName	String	统计纬度 示例值：intraffic
MaxData	Integer	返回DataList中的最大值 示例值：10
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 请求入包速率

请求入包速率

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBizMonitorTrend
<公共请求参数>

{
  "Business": "bgpip",
  "Period": 60,
  "StartTime": "2023-06-21 10:00:00",
  "EndTime": "2023-06-21 10:30:00",
  "Id": "bgpip-0000xbc",
  "MetricName": "inpkg"
}
```

输出示例

```
{
  "Response": {
    "DataList": [
      1,
      12
    ],
    "MetricName": "inpkg",
    "MaxData": 12,
    "RequestId": "abc-as"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。

获取高防IP业务流量曲线

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

获取业务流量曲线

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBizTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Statistics	是	String	统计方式，可取值max, min, avg, sum, 如统计纬度是流量速率或包量速率，仅可取值max 示例值：max
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP） 示例值：bgp
Period	是	Integer	统计周期，可取值60，300，1800，3600，21600，86400，单位秒 示例值：1
StartTime	是	Timestamp	统计开始时间。例：“2020-09-22 00:00:00” 示例值：2020-09-22 00:00:00
EndTime	是	Timestamp	统计结束时间。例：“2020-09-22 00:00:00” 示例值：2020-09-22 00:00:00
Id	是	String	资源实例ID 示例值：bgp-0000
MetricName	是	String	统计纬度，可取值connum, new_conn, inactive_conn, intraffic, outtraffic, inpkg, outpkg, qps 示例值：intraffic
Domain	否	String	统计纬度为qps时，可选特定域名查询 示例值：test.tencentddos.com
ProtoInfo.N	否	Array of ProtocolPort	协议及端口列表，协议可取值TCP, UDP, HTTP, HTTPS，仅统计纬度为连接数时有效
BusinessType	否	String	业务类型可取值domain, port port：端口业务 domain：域名业务 示例值：domain

3. 输出参数

参数名称	类型	描述
DataList	Array of Float	曲线图各个时间点的值 示例值: [1.2,2.2]
MetricName	String	统计纬度 示例值: intratrafic
MaxData	Integer	返回DataList中的最大值 示例值: 0
RequestId	String	唯一请求 ID, 由服务端生成, 每次请求都会返回 (若请求因其他原因未能抵达服务端, 则该次请求不会获得 RequestId)。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取高防IP业务流量曲线

获取高防IP业务流量曲线

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBizTrend
<公共请求参数>

{
  "Domain": "test.com",
  "Statistics": "avg",
  "Business": "bgpip",
  "ProtoInfo": [
    {
      "Protocol": "TCP",
      "Port": 1
    }
  ],
  "Period": 1,
  "StartTime": "2020-09-22 00:00:00",
  "EndTime": "2020-09-22 00:00:00",
  "Id": "bgpip-00000001",
  "BusinessType": "domain",
  "MetricName": "qps"
}
```

输出示例

```
{
  "Response": {
    "DataList": [
      236,
      236,
      236,
      236,
      236,
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。

获取CC攻击指标曲线

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC攻击指标数据，包括总请求峰值(QPS)和攻击请求(QPS)以及总请求次数和攻击请求次数

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版；basic表示DDoS基础防护） 示例值：bgp
Ip	是	String	资源的IP 示例值：1.1.1.1
Period	是	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：3600
StartTime	是	Timestamp	统计开始时间 示例值：2018-08-2715:05:10
EndTime	是	Timestamp	统计结束时间 示例值：2018-08-2716:05:10
MetricName	是	String	指标，取值[inqps(总请求峰值，dropqps(攻击请求峰值))，incount(请求次数)，dropcount(攻击次数)] 示例值：inqps
Domain	否	String	域名，可选 示例值：www.xx.com
Id	否	String	资源实例ID，当Business为basic时，此字段不用填写（因为基础防护没有资源实例） 示例值：bgp-00000010

3. 输出参数

参数名称	类型	描述
Count	Integer	值个数 示例值：1
Business	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版；basic表示DDoS基础防护） 示例值：bgp

参数名称	类型	描述
Ip	String	资源的IP 示例值：3.3.3.3
Period	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：300
StartTime	Timestamp	统计开始时间 示例值：2018-08-27 15:05:10
EndTime	Timestamp	统计结束时间 示例值：2018-08-27 16:05:10
Data	Array of Integer	值数组 示例值：[1234]
Id	String	资源ID 示例值：bgp-00000010
MetricName	String	指标，取值[inqps(总请求峰值，dropqps(攻击请求峰值))，incount(请求次数)，dropcount(攻击次数)] 示例值：inqps
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC攻击指标数据

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCCTrend
<公共请求参数>

{
  "Domain": "www.test.com",
  "Business": "bgp",
  "Ip": "1.2.3.4",
  "Period": 0,
  "StartTime": "2020-09-22 00:00:00",
  "EndTime": "2020-09-22 00:00:00",
  "Id": "bgp-00000010",
  "MetricName": "inqps"
}
```

输出示例

```
{
  "Response": {
    "Business": "bgp",
    "Count": 1,
    "Data": [
      1234
    ],
  },
}
```

```
"EndTime": "2018-08-27 16:05:10",
"Id": "bgp-00000010",
"Ip": "3.3.3.3",
"MetricName": "inqps",
"Period": 300,
"RequestId": "eac6b301-a322-493a-8e36-83b295459397",
"StartTime": "2018-08-27 15:05:10"
}
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取DDoS攻击指标曲线

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

获取DDoS攻击流量带宽和攻击包速率数据

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeDDoS Trend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版；basic表示DDoS基础防护） 示例值：bgp
Ip	是	String	资源实例的IP 示例值：3.3.3.3
Period	是	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：3600
StartTime	是	Timestamp	统计开始时间 示例值：2018-08-2715:05:10
EndTime	是	Timestamp	统计结束时间 示例值：2018-08-2716:05:10
MetricName	是	String	指标，取值[bps(攻击流量带宽，pps(攻击包速率))] 示例值：bps
Id	否	String	资源实例ID，当Business为basic时，此字段不用填写（因为基础防护没有资源实例） 示例值：bgp-00000010

3. 输出参数

参数名称	类型	描述
Count	Integer	值个数 示例值：1
Business	String	DDoS防护子产品代号（bgpip表示高防IP；bgp表示独享包；bgp-multip表示共享包；net表示高防IP专业版；basic表示DDoS基础防护） 示例值：bgp
Ip	String	资源的IP 示例值：3.3.3.3

参数名称	类型	描述
Period	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：3600
StartTime	Timestamp	统计开始时间 示例值：2018-08-27 15:05:10
EndTime	Timestamp	统计结束时间 示例值：2018-08-27 16:05:10
Data	Array of Integer	值数组，攻击流量带宽单位为Mbps，包速率单位为pps 示例值：[1234]
Id	String	资源ID 示例值：bgp-00000010
MetricName	String	指标，取值[bps(攻击流量带宽，pps(攻击包速率))] 示例值：bps
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DDoS攻击指标数据

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeDDoS Trend
<公共请求参数>

{
  "Business": "bgpip",
  "Ip": "1.1.1.1",
  "Period": 330,
  "StartTime": "2022-10-10 01:08:00",
  "EndTime": "2022-10-10 01:13:00",
  "Id": "bgpip-00000xxx",
  "MetricName": "pps"
}
```

输出示例

```
{
  "Response": {
    "Business": "bgpip",
    "Count": 2,
    "Data": [
      30,
      20
    ],
    "EndTime": "2022-10-10 01:13:00",
    "Id": "bgpip-00000xxx",
  }
}
```

```
"Ip": "1.1.1.1",
"MetricName": "pps",
"Period": 300,
"RequestId": "2f44a9d8-2b6f-4a9a-b1da-9d05d9e6ad83",
"StartTime": "2022-10-10 01:05:00"
}
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

拉取防护概览攻击趋势

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

拉取防护概览攻击趋势

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeOverviewAttackTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Type	是	String	攻击类型，取值ddos，cc 示例值：ddos
Dimension	是	String	纬度，当前仅支持attackcount 示例值：attackcount
Period	是	Integer	周期，当前仅支持86400 示例值：86400
StartTime	是	String	防护概览攻击趋势开始时间 示例值：2021-03-17 12:32:12
EndTime	是	String	防护概览攻击趋势结束时间 示例值：2021-03-17 12:32:12

3. 输出参数

参数名称	类型	描述
Type	String	攻击类型 示例值：ddos
StartTime	String	防护概览攻击趋势起始时间 示例值：2021-05-24 14:23:54
EndTime	String	防护概览攻击趋势结束时间 示例值：2021-05-24 14:23:54
Period	Integer	周期 示例值：86400
Data	Array of Integer	每个周期点的攻击次数 示例值：[0, 0, 0, 0, 0, 0]

参数名称	类型	描述
Count	Integer	包含的周期点数 示例值：5
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 拉取防护概览攻击趋势

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeOverviewAttackTrend
<公共请求参数>

{
  "EndTime": "2020-02-03 18:03:23",
  "Type": "ddos",
  "Dimension": "attackcount",
  "StartTime": "2020-02-01 12:04:12",
  "Period": 86400
}
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Type": "ddos",
    "StartTime": "2020-02-01 12:04:12",
    "EndTime": "2020-02-03 18:03:23",
    "Period": 86400,
    "Data": [
      0,
      0
    ],
    "Count": 2
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取防护概览趋势图

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取防护概览总请求峰值(QPS)和攻击请求(QPS)以及总请求次数和攻击请求次数

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeOverviewCCTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Period	是	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：3600
StartTime	是	Timestamp	统计开始时间 示例值：2018-08-2715:05:10
EndTime	是	Timestamp	统计结束时间 示例值：2018-08-2716:05:10
MetricName	是	String	指标，取值[inqps(总请求峰值，dropqps(攻击请求峰值))，incount(请求次数)，dropcount(攻击次数)] 示例值：inqps
Business	否	String	DDoS防护子产品代号（bgpip表示高防IP；bgp-multip表示共享包；basic表示DDoS基础防护） 示例值：bgp
IpList.N	否	Array of String	资源的IP 示例值：1.1.1.1
Id	否	String	资源实例ID 示例值：bgp-00000010

3. 输出参数

参数名称	类型	描述
Count	Integer	值个数 示例值：1
Data	Array of Integer	值数组 示例值：[1234]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取防护概览趋势图

当Business可以不传，统计所有Business为 基础防护 + 高防ip + 高防包 的和

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeOverviewCCTrend
<公共请求参数>

{
  "Period": 3600,
  "StartTime": "2020-09-22 00:00:00",
  "EndTime": "2020-09-23 00:00:00",
  "MetricName": "dropqps"
}
```

输出示例

```
{
  "Response": {
    "Count": 1,
    "Data": [
      1234
    ],
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取防护概览的ddos攻击事件

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取防护概览的ddos攻击事件

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeOverviewDDoSEventList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
StartTime	是	String	起始时间 示例值：2021-03-17 12:32:12
EndTime	是	String	结束时间 示例值：2021-03-17 12:32:12
AttackStatus	否	String	可选按攻击状态过滤，start：攻击中；end：攻击结束 示例值：start
Offset	否	Integer	偏移量 示例值：0
Limit	否	Integer	记录条数 示例值：10

3. 输出参数

参数名称	类型	描述
Total	Integer	记录总数 示例值：1
EventList	Array of OverviewDDoSEvent	事件列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取防护概览的ddos攻击事件

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeOverviewDDoSEventList
&StartTime=2020-02-01 12:04:12
&EndTime=2020-02-03 18:03:23
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "Total": 1,
    "EventList": [
      {
        "Id": "ddosev-00000421",
        "Vip": "10.143.145.42",
        "StartTime": "2021-04-14 14:14:54",
        "EndTime": "2021-04-14 14:16:23",
        "AttackType": "SYNFLOOD",
        "AttackStatus": 0,
        "Mbps": 23414,
        "Pps": 2313,
        "Business": "bgpip",
        "InstanceId": "bgpip-00000143",
        "InstanceName": "name"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取防护概览DDoS攻击流量带宽和攻击包速率数据

最近更新时间：2025-04-25 01:08:00

1. 接口描述

接口请求域名： antiddos.tencentcloudapi.com 。

获取防护概览DDoS攻击流量带宽和攻击包速率数据

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeOverviewDDoSSTrend。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Period	是	Integer	统计粒度，取值[300(5分钟)，3600(小时)，86400(天)] 示例值：3600
StartTime	是	Timestamp	统计开始时间 示例值：2018-08-2715:05:10
EndTime	是	Timestamp	统计结束时间 示例值：2018-08-2716:05:10
MetricName	是	String	指标，取值[bps(攻击流量带宽，pps(攻击包速率))] 示例值：bps
Business	否	String	DDoS防护子产品代号（bgpip表示高防IP；bgp-multip表示高防包；basic表示DDoS基础防护） 示例值：bgp
IpList.N	否	Array of String	资源实例的IP列表 示例值：1.1.1.1
Id	否	String	资源实例ID 示例值：bgp-00000010

3. 输出参数

参数名称	类型	描述
Count	Integer	值个数 示例值：1
Data	Array of Integer	值数组，攻击流量带宽单位为Mbps，包速率单位为pps 示例值：[1234]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取防护概览DDoS攻击流量带宽和攻击包速率数据

当Business可以不传，统计所有Business为 基础防护 + 高防ip + 高防包 的和

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeOverviewDDoS Trend
<公共请求参数>

{
  "Business": "bgp",
  "Period": 0,
  "StartTime": "2020-09-22 00:00:00",
  "EndTime": "2020-09-22 00:00:00",
  "Id": "bgp-00000001",
  "MetricName": "bps"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "cb39db80-b483-4139-a00e-1177ed011579",
    "Count": 1,
    "Data": [
      12234
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

拉取防护概览指标

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

拉取防护概览指标

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeOverviewIndex。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
StartTime	是	String	拉取指标起始时间 示例值：2021-03-17 12:32:12
EndTime	是	String	拉取指标结束时间 示例值：2021-03-17 12:32:12

3. 输出参数

参数名称	类型	描述
AllIpCount	Integer	IP总数 示例值：100
AntiddosIpCount	Integer	高防IP总数（包含高防包+高防IP） 示例值：10
AttackIpCount	Integer	攻击IP总数 示例值：2
BlockIpCount	Integer	封堵IP总数 示例值：1
AntiddosDomainCount	Integer	高防域名总数 示例值：10
AttackDomainCount	Integer	攻击域名总数 示例值：2
MaxAttackFlow	Integer	攻击流量峰值 示例值：245
NewAttackTime	String	当前最近一条攻击中的起始时间 示例值：2021-05-15 21:43:15

参数名称	类型	描述
NewAttackIp	String	当前最近一条攻击中的IP 示例值：10.34.14.54
NewAttackType	String	当前最近一条攻击中的攻击类型 示例值：SYNFLOOD
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 拉取防护概览指标

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeOverviewIndex
&StartTime=2020-02-01 12:04:12
&EndTime=2020-02-03 18:03:23
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "eac6b301-a322-493a-8e36-83b295459397",
    "AllIpCount": 100,
    "AntiddosIpCount": 10,
    "AttackIpCount": 2,
    "BlockIpCount": 1,
    "AntiddosDomainCount": 10,
    "AttackDomainCount": 2,
    "MaxAttackFlow": 245,
    "NewAttackTime": "2021-05-15 21:43:15",
    "NewAttackIp": "10.34.14.54",
    "NewAttackType": "SYNFLOOD"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

查询账号维度待处理风险信息

最近更新时间：2025-04-25 01:07:59

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

查询账号维度待处理风险信息，包括是否为付费用户，查询攻击中、封堵中、过期资源数量等

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribePendingRiskInfo。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。

3. 输出参数

参数名称	类型	描述
IsPaidUsr	Boolean	是否为付费用户，true：付费用户，false：普通用户 示例值：true
AttackingCount	Integer	攻击中的资源数量 示例值：10
BlockingCount	Integer	封堵中的资源数量 示例值：10
ExpiredCount	Integer	已过期的资源数量 示例值：10
Total	Integer	所有待处理风险事件总数 示例值：30
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询账号维度待处理风险信息

查询账号维度待处理风险信息

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
```

```
X-TC-Action: DescribePendingRiskInfo
```

```
<公共请求参数>
```

```
{}
```

输出示例

```
{
  "Response": {
    "ExpiredCount": 144,
    "IsPaidUser": true,
    "AttackingCount": 0,
    "RequestId": "0506d138-ef0f-4ff4-83b0-f1d85e740afd",
    "Total": 146,
    "BlockingCount": 2
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

其他接口

新增CC精准防护策略

最近更新时间：2025-04-25 01:08:05

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

新增CC精准防护策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateCCPrecisionPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
Ip	是	String	IP值 示例值：1.2.1.3
Protocol	是	String	协议，可取值http，https，http/https 示例值：http
Domain	是	String	域名 示例值：1.acx.com
PolicyAction	是	String	策略方式，可取值alg表示人机校验，drop表示丢弃，trans表示放行 示例值：alg
PolicyList.N	是	Array of CCPrecisionPlyRecord	策略记录

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 新增CC精准防护策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=CreateCCPrecisionPolicy
&InstanceId=bgpip-0000011x
&Ip=1.2.1.3
&Protocol=HTTP
&Domain=1.acx.com
&PolicyAction=alg
&PolicyList.0.FieldType=value
&PolicyList.0.FieldName=cgi
&PolicyList.0.Value=/asd
&PolicyList.0.ValueOperator=equal
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。

错误码	描述
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

新增CC频率限制策略

最近更新时间：2025-04-25 01:08:05

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

新增CC频率限制策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：CreateCCReqLimitPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-00000235
Ip	是	String	IP值 示例值：1.2.1.5
Protocol	是	String	协议，可取值http, https, http/https 示例值：http
Domain	是	String	域名 示例值：1.csq.com
Policy	是	CCReqLimitPolicyRecord	策略项
IsGlobal	否	Integer	是否为兜底频控 0表示不是 1表示是 示例值：1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 新增CC频率限制策略

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: CreateCCReqLimitPolicy
<公共请求参数>
```

```
{
  "Domain": "www.baidu.com",
  "InstanceId": "bgpip-000005v7",
  "Ip": "1.1.1.1",
  "Policy": {
    "Action": "alg",
    "ExecuteDuration": 120,
    "Mode": "equal",
    "Period": 10,
    "RequestNum": 500,
    "Uri": "www.abc.com"
  },
  "Protocol": "http"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除CC分级策略

最近更新时间：2025-04-25 01:08:05

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC分级策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCCLevelPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
Ip	是	String	配置策略的IP 示例值：1.1.1.3
Domain	是	String	域名 示例值：test.com
Protocol	是	String	协议，可取值http 示例值：http

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC分级策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteCCLevelPolicy
&InstanceId=bgpip-0000011x
&Ip=1.1.1.3
&Domain=test.com
&Protocol=http
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除CC精准防护策略

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC精准防护策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCCPrecisionPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
PolicyId	是	String	策略Id 示例值：ccPrecs-00000324

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC精准防护策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteCCPrecisionPolicy
&InstanceId=bgpip-0000011x
&PolicyId=ccPrecs-00000324
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
```

```
}  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。

错误码	描述
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除CC频率限制策略

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC频率限制策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCCRequestLimitPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000023c
PolicyId	是	String	策略Id 示例值：ccRule-00000435

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC频率限制策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteCCRequestLimitPolicy
&InstanceId=bgpip-0000023c
&PolicyId=ccRule-00000435
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
```

```
}  
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。

错误码	描述
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除CC清洗阈值策略

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

删除CC清洗阈值策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteCCThresholdPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
Ip	是	String	配置策略的IP 示例值：1.1.1.3
Domain	是	String	域名 示例值：test.com
Protocol	是	String	协议，可取值http 示例值：http

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除CC清洗阈值策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DeleteCCThresholdPolicy
&InstanceId=bgpip-0000011x
&Ip=1.1.1.3
&Domain=test.com
&Protocol=http
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取基础防护攻击状态

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取基础防护攻击状态

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeBasicDeviceStatus。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
IpList.N	否	Array of String	IP 资源列表 示例值：["1.1.1.1"]
IdList.N	否	Array of String	域名化资源传id 示例值：id-xx
FilterRegion	否	Integer	地域名称 示例值：guangzhou
CnameWafIdList.N	否	Array of String	cnameWaf资源 示例值：id-xxx

3. 输出参数

参数名称	类型	描述
Data	Array of KeyValue	返回资源及状态，状态码： 1 - 封堵状态 2 - 正常状态 3 - 攻击状态
CLBData	Array of KeyValue	域名化资产的名称
CnameWafData	Array of KeyValue	cnamewaf资源状态
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取基础防护状态

获取基础防护状态

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeBasicDeviceStatus
<公共请求参数>

{
  "IpList": [
    "1.1.1.1"
  ],
  "IdList": [
    "222"
  ],
  "FilterRegion": 1
}
```

输出示例

```
{
  "Response": {
    "Data": [
      {
        "Key": "1.1.1.1",
        "Value": "2"
      }
    ],
    "CLBData": [
      {
        "Key": "1.1.1.1",
        "Value": "3"
      }
    ],
    "CnameWafData": [
      {
        "Key": "1.1.1.1",
        "Value": "1"
      }
    ],
    "RequestId": "07a620b1"
  }
}
```

示例2 获取基础防护攻击状态

无

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeBasicDeviceStatus
<公共请求参数>
```

```
{
  "IpList": [
    "127.0.0.1"
  ]
}
```

输出示例

```
{
  "Response": {
    "Data": [
      {
        "Key": "1.1.1.1",
        "Value": "1"
      }
    ],
    "CLBData": [
      {
        "Key": "1.1.1.1",
        "Value": "1"
      }
    ],
    "CnameWafData": [
      {
        "Key": "1.1.1.1",
        "Value": "1"
      }
    ],
    "RequestId": "8f7a9451-2372-4a88-a69d-9868b8991076"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取CC防护等级列表

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取边界防护CC防护等级列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCLevelList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp-multip表示高防包） 示例值：bgp-multip
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数 示例值：1
InstanceId	否	String	指定实例Id 示例值：bgp-xxxx

3. 输出参数

参数名称	类型	描述
Total	Integer	分级策略列表总数 示例值：1
LevelList	Array of CCLevelPolicy	分级策略列表总数
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取边界防护CC防护等级列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeCCLevelList
&Business=edge
```

```
&Offset=0
&Limit=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794",
    "Total": 1,
    "LevelList": [
      {
        "InstanceId": "edge-000000xe",
        "Ip": "1.2.1.1",
        "Protocol": "HTTP",
        "Domain": "1.ase.com",
        "Level": "loose",
        "CreateTime": "2020-03-27 14:32:12",
        "ModifyTime": "2020-03-27 14:32:32"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取CC分级策略

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC分级策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCLevelPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
Ip	是	String	IP值 示例值：1.3.2.5
Domain	是	String	域名 示例值：1.sca.com
Protocol	是	String	协议，可取值http、https、http/https 示例值：http

3. 输出参数

参数名称	类型	描述
Level	String	CC防护等级，可取值loose表示宽松，strict表示严格，normal表示适中，emergency表示攻击紧急，sup_loose表示超级宽松，default表示默认策略（无频控配置下发），customized表示自定义策略 示例值：loose
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC分级策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeCCLevelPolicy
&InstanceId=bgpip-0000011x
&Ip=1.3.2.5
```

```
&Domain=1.sca.com
&Protocol=HTTP
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794",
    "Level": "loose"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。

错误码	描述
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取CC精准防护列表

最近更新时间：2025-04-25 01:08:04

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC精准防护列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCPrecisionPlyList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgpip-multip：表示高防包；bgpip：表示高防IP） 示例值：bgp-multip
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：1
Limit	是	Integer	一页条数 示例值：1
InstanceId	否	String	指定特定实例Id 示例值：bgp-xxxx
Ip	否	String	IP地址，普通高防IP要传该字段 示例值：1.1.1.1
Domain	否	String	域名，普通高防IP要传该字段 示例值：www.qq.com
Protocol	否	String	协议，普通高防IP要传该字段 示例值：http

3. 输出参数

参数名称	类型	描述
Total	Integer	策略列表总数 示例值：1
PrecisionPolicyList	Array of CCPrecisionPolicy	策略列表详情
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC精准防护列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCCPrecisionPlyList
<公共请求参数>

{
  "Business": "bgp-multip",
  "InstanceId": "bgp-00000211",
  "Limit": 10,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "RequestId": "23a97c66-22d4-4522-b93b-388aac08777d",
    "Total": 1,
    "PrecisionPolicyList": [
      {
        "PolicyId": "ccPrecs-00000001",
        "InstanceId": "bgp-00000211",
        "Ip": "1.1.1.1",
        "Protocol": "http",
        "Domain": "www.test.com",
        "PolicyAction": "alg",
        "PolicyList": [
          {
            "FieldType": "value",
            "FieldName": "cgi",
            "Value": "value",
            "ValueOperator": "include"
          }
        ],
        "CreateTime": "2022-12-01 09:05:36",
        "ModifyTime": "2022-12-01 09:05:36"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取CC频率限制策略列表

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC频率限制策略列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCReqLimitPolicyList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp-multip表示高防包，bgpip表示高防IP） 示例值：bgp-multip
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：1
Limit	是	Integer	一页条数 示例值：1
InstanceId	否	String	指定实例Id 示例值：id-xxx
Ip	否	String	IP地址，普通高防IP要传该字段 示例值：1.1.1.1
Domain	否	String	域名，普通高防IP要传该字段 示例值：www.xx.com
Protocol	否	String	协议，普通高防IP要传该字段 示例值：http

3. 输出参数

参数名称	类型	描述
Total	Integer	频率限制列表总数 示例值：1
RequestLimitPolicyList	Array of CCReqLimitPolicy	频率限制列表详情
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC频率限制策略列表

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCCReqLimitPolicyList
<公共请求参数>

{
  "Business": "bgp-multip",
  "InstanceId": "bgp-00000001",
  "Limit": 10,
  "Offset": 0
}
```

输出示例

```
{
  "Response": {
    "RequestId": "ea775a73-7de5-4d36-aca0-29307e7c4f27",
    "Total": 1,
    "RequestLimitPolicyList": [
      {
        "PolicyId": "ccRule-00009xn0",
        "InstanceId": "bgp-00000001",
        "Ip": "1.1.1.1",
        "Protocol": "http",
        "Domain": "www.test.com",
        "PolicyRecord": {
          "Period": 10,
          "RequestNum": 500,
          "Action": "alg",
          "ExecuteDuration": 120,
          "Mode": "equal",
          "Uri": "www.a.com",
          "UserAgent": "",
          "Cookie": ""
        },
        "CreateTime": "2022-12-02 09:11:48",
        "ModifyTime": "2022-12-02 09:11:48"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取CC清洗阈值列表

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

获取CC清洗阈值列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DescribeCCThresholdList。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
Business	是	String	DDoS防护子产品代号（bgp-multip表示高防包） 示例值：bgp-multip
Offset	是	Integer	页起始偏移，取值为(页码-1)*一页条数 示例值：0
Limit	是	Integer	一页条数 示例值：1
InstanceId	否	String	指定实例Id 示例值：ins-xxx

3. 输出参数

参数名称	类型	描述
Total	Integer	清洗阈值策略列表总数 示例值：1
ThresholdList	Array of CCThresholdPolicy	清洗阈值策略列表详情
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取CC清洗阈值列表

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=DescribeCCThresholdList
&Business=bgp-multip
```

```
&Offset=0
&Limit=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794",
    "Total": 1,
    "ThresholdList": [
      {
        "InstanceId": "bgp-000000432",
        "Ip": "1.2.1.1",
        "Protocol": "HTTP",
        "Domain": "1.ase.com",
        "Threshold": 1000,
        "CreateTime": "2020-03-27 14:32:12",
        "ModifyTime": "2020-03-27 14:32:32"
      }
    ]
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

修改CC的防护等级

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改CC防护等级

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyCCLevelPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000023c
Ip	是	String	IP地址 示例值：1.3.2.6
Domain	是	String	域名 示例值：1.sdc.com
Protocol	是	String	协议，可取值http、https、http/https 示例值：http
Level	是	String	CC防护等级，可取值loose表示宽松，strict表示严格，normal表示适中，emergency表示攻击紧急，sup_loose表示超级宽松，default表示默认策略（无频控配置下发），customized表示自定义策略 示例值：loose

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改CC防护等级

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyCCLevelPolicy
&InstanceId=bgpip-0000023c
&Ip=1.3.2.6
```

```
&Domain=1.sdc.com
&Protocol=HTTP
&Level=loose
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。

错误码	描述
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

修改CC精准防护策略

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改CC精准防护策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyCCPrecisionPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000011x
PolicyId	是	String	策略Id 示例值：ccPrecs-00000326
PolicyAction	是	String	策略方式。可取值：alg、drop、trans。alg指返回验证码方式验证，drop表示该访问丢弃，trans表示该访问放行。 示例值：alg
PolicyList.N	是	Array of CCPrecisionPlyRecord	策略记录

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改CC精准防护策略

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyCCPrecisionPolicy
&InstanceId=bgpip-0000011x
&PolicyId=ccPrecs-00000326
&PolicyAction=alg
&PolicyList.0.FieldType=value
```

```
&PolicyList.0.FieldName=cgi
&PolicyList.0.Value=/cadwe
&PolicyList.0.ValueOperator=equal
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。

错误码	描述
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

修改CC频率限制策略

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改CC频率限制策略

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyCCReqLimitPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000023c
PolicyId	是	String	策略Id 示例值：ccRule-00000312
Policy	是	CCReqLimitPolicyRecord	策略项

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改CC频率限制策略

输入示例

```
POST / HTTP/1.1
Host: antiddos.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyCCReqLimitPolicy
<公共请求参数>

{
  "InstanceId": "bgpip-00000001",
  "Policy": {
```

```
"Action": "alg",
"ExecuteDuration": 100,
"Mode": "equal",
"Period": 10,
"RequestNum": 600,
"Uri": "www.a.com"
},
"PolicyId": "ccRule-00009001"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。

错误码	描述
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

修改CC清洗阈值

最近更新时间：2025-04-25 01:08:03

1. 接口描述

接口请求域名：antiddos.tencentcloudapi.com。

修改CC清洗阈值

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：ModifyCCThresholdPolicy。
Version	是	String	公共参数 ，本接口取值：2020-03-09。
Region	否	String	公共参数 ，本接口不需要传递此参数。
InstanceId	是	String	实例Id 示例值：bgpip-0000023c
Ip	是	String	IP地址 示例值：1.3.2.6
Domain	是	String	域名 示例值：1.sdc.com
Protocol	是	String	协议，可取值http，https，http/https 示例值：http
Threshold	是	Integer	清洗阈值，-1表示开启“默认”模式 示例值：1000

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改CC清洗阈值

输入示例

```
https://antiddos.tencentcloudapi.com/?Action=ModifyCCThresholdPolicy
&InstanceId=bgpip-0000023c
&Ip=1.3.2.6
&Domain=1.sdc.com
```

```
&Protocol=HTTP
&Threshold=1000
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5063ab0a-a8a7-41e8-ace2-263b2c1c8794"
  }
}
```

5. 开发者资源

腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#), [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#), [Gitee](#)

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。

错误码	描述
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

数据结构

最近更新时间：2025-03-31 01:07:35

AclConfig

基于端口的acl策略

被如下接口引用：CreatePortAclConfig, CreatePortAclConfigList, DeletePortAclConfig, DescribeListPortAclList, ModifyPortAclConfig。

名称	类型	必选	描述
ForwardProtocol	String	是	协议类型, 可取值tcp, udp, all 示例值: tcp
DPortStart	Integer	是	目的端口起始, 可取值范围0~65535 示例值: 0
DPortEnd	Integer	是	目的端口结束, 可取值范围0~65535 示例值: 65535
SPortStart	Integer	是	来源端口起始, 可取值范围0~65535 示例值: 0
SPortEnd	Integer	是	来源端口结束, 可取值范围0~65535 示例值: 65535
Action	String	是	动作, 可取值: drop, transmit, forward 示例值: 65535
Priority	Integer	否	策略优先级, 数字越小, 级别越高, 该规则越靠前匹配, 取值1~1000 示例值: 10

AclConfigRelation

端口acl策略配置与高防资源关联

被如下接口引用：DescribeListPortAclList。

名称	类型	描述
AclConfig	AclConfig	acl策略
InstanceDetailList	Array of InstanceRelation	实例列表

AnycastOutPackRelation

Anycast转外套详情

被如下接口引用：DescribeListBGPIPIInstances。

名称	类型	描述
NormalBandwidth	Integer	业务带宽(单位M) 示例值: 200
ForwardRulesLimit	Integer	转发规则数 示例值: 60
AutoRenewFlag	Integer	自动续费标记 示例值: 0
CurDeadline	String	到期时间 示例值: 2022-06-18 18:36:44

BGPIPIInstance

高防IP资产实例信息

被如下接口引用：DescribeListBGPIPIInstances。

名称	类型	描述
InstanceDetail	InstanceRelation	资产实例的详细信息
SpecificationLimit	BGPIPIInstanceSpecification	资产实例的规格信息
Usage	BGPIPIInstanceUsages	资产实例的使用统计信息
Region	RegionInfo	资产实例所在的地域
Status	String	资产实例的防护状态，状态码如下： "idle": 正常状态(无攻击) "attacking": 攻击中 "blocking": 封堵中 "creating": 创建中 "deblocking": 解封中 "isolate": 回收隔离中 示例值：idle
ExpiredTime	Timestamp	到期时间 示例值：2020-06-09 23:12:23
CreatedTime	Timestamp	购买时间 示例值：2020-05-09 23:12:23
Name	String	资产实例的名称 示例值：Test
PackInfo	PackInfo	资产实例所属的套餐包信息， 注意：当资产实例不是套餐包的实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。
StaticPackRelation	StaticPackRelation	资产实例所属的三网套餐包详情， 注意：当资产实例不是三网套餐包的实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。
Zoneld	Integer	区分高防IP境外线路 示例值：1
Tgw	Integer	区分集群 示例值：1
EipAddressStatus	String	高防弹性公网IP状态，包含'CREATING'(创建中),'BINDING'(绑定中),'BIND'(已绑定),'UNBINDING'(解绑中),'UNBIND'(已解绑),'OFFLINING'(释放中),'BIND_ENI'(绑定悬空弹性网卡)。只对高防弹性公网IP实例有效。 示例值：UNBIND
EipFlag	Integer	是否高防弹性公网IP实例，是为1，否为0。 示例值：1
EipAddressPackRelation	EipAddressPackRelation	资产实例所属的高防弹性公网IP套餐包详情， 注意：当资产实例不是高防弹性公网IP套餐包的实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。
EipAddressInfo	EipAddressRelation	高防弹性公网IP关联的实例信息。 注意：当资产实例不是高防弹性公网IP实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。
Domain	String	建议客户接入的域名，客户可使用域名接入。 示例值：www.8cname.com

名称	类型	描述
DamDDoSStatus	Integer	是否开启安全加速，是为1，否为0。 示例值：0
V6Flag	Integer	是否IPv6版本的IP，是为1，否为0 示例值：0
BGPIPChannelFlag	Integer	是否渠道版高防IP，是为1，否为0 示例值：0
TagInfoList	Array of TagInfo	资源关联标签 示例值：null
AnycastOutPackRelation	AnycastOutPackRelation	资产实例所属的全力防护套餐包详情， 注意：当资产实例不是全力防护套餐包的实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。
InstanceVersion	Integer	资源实例版本 示例值：0
ConvoyId	String	重保实例 示例值：id-xxx
ElasticBandwidth	Integer	带宽后付费 示例值：1000
EOFlag	Integer	是否为EO代播的ip: 1是，0不是 示例值：0

BGPIPInstanceSpecification

高防IP资产实例的规格信息

被如下接口引用：DescribeListBGPIPInstances。

名称	类型	描述
ProtectBandwidth	Integer	保底防护峰值，单位Mbps 示例值：2000
ProtectCCQPS	Integer	CC防护峰值，单位qps 示例值：2000
NormalBandwidth	Integer	正常业务带宽，单位Mbps 示例值：2000
ForwardRulesLimit	Integer	转发规则数，单位条 示例值：50
AutoRenewFlag	Integer	自动续费状态，取值[0：没有开启自动续费 1：开启了自动续费] 示例值：1
Line	Integer	高防IP线路，取值为[1：BGP线路 2：电信 3：联通 4：移动 99：第三方合作线路] 示例值：1

名称	类型	描述
ElasticBandwidth	Integer	弹性防护峰值，单位Mbps 示例值：1000

BGPIInstanceUsages

高防IP资产实例的使用信息统计

被如下接口引用：DescribeListBGPIInstances。

名称	类型	描述
PortRulesUsage	Integer	已使用的端口规则数，单位条 示例值：1
DomainRulesUsage	Integer	已使用的域名规则数，单位条 示例值：1
Last7DayAttackCount	Integer	最近7天的攻击次数，单位次 示例值：1

BGPIPL7RuleEntry

L7规则

被如下接口引用：DescribeBGPIPL7Rules。

名称	类型	必选	描述
Protocol	String	是	转发协议，取值[http, https] 示例值：80
Domain	String	是	转发域名 示例值：www.abc.com
LbType	Integer	是	负载均衡方式，取值[1(加权轮询)] 示例值：1
KeepEnable	Integer	是	会话保持开关，取值[0(会话保持关闭)，1(会话保持开启)] 示例值：1
KeepTime	Integer	是	会话保持时间，单位秒 示例值：12
SourceType	Integer	是	回源方式，取值[1(域名回源)，2(IP回源)] 示例值：1
SourceList	Array of L4RuleSource	是	回源列表
Region	Integer	否	区域码 示例值：1
Id	String	否	资源Id 示例值：id-xxx
Ip	String	否	资源Ip 示例值：1.1.1.1
RuleId	String	否	规则ID，当添加新规则时可以用不用填写此字段；当修改或者删除规则时需要填写此字段； 示例值：ruleid
RuleName	String	否	规则描述 示例值：name

名称	类型	必选	描述
CertType	Integer	否	证书来源，当转发协议为https时必须填，取值[2(腾讯云托管证书)]，当转发协议为http时也可以填0 示例值：1
SSLId	String	否	当证书来源为腾讯云托管证书时，此字段必须填写托管证书ID 示例值：id-xxx
Cert	String	否	当证书来源为自有证书时，此字段必须填写证书内容；(因已不再支持自有证书，此字段已弃用，请不用填写此字段) 示例值：cert
PrivateKey	String	否	当证书来源为自有证书时，此字段必须填写证书密钥；(因已不再支持自有证书，此字段已弃用，请不用填写此字段) 示例值：key
Status	Integer	否	规则状态，取值[0(规则配置成功)，1(规则配置生效中)，2(规则配置失败)，3(规则删除生效中)，5(规则删除失败)，6(规则等待配置)，7(规则等待删除)，8(规则待配置证书)] 示例值：1
CCStatus	Integer	否	cc防护状态，取值[0(关闭)，1(开启)] 示例值：1
CCEnable	Integer	否	HTTPS协议的CC防护状态，取值[0(关闭)，1(开启)] 示例值：1
CCThreshold	Integer	否	HTTPS协议的CC防护阈值（已废弃） 示例值：1
CCLevel	String	否	HTTPS协议的CC防护等级 示例值：宽松
ModifyTime	Timestamp	否	修改时间 示例值：2024.10.1
HttpsToHttpEnable	Integer	否	是否开启Https协议使用Http回源，取值[0(关闭)，1(开启)]，不填写默认是关闭 示例值：1
VirtualPort	Integer	否	接入端口值 示例值：80
RewriteHttps	Integer	否	http强制跳转https，1表示打开，0表示关闭 示例值：10
ErrCode	Integer	否	规则配置失败时的详细错误原因(仅当Status=2时有效)，1001证书不存在，1002证书获取失败，1003证书上传失败，1004证书已过期 示例值：10
Version	Integer	否	版本 示例值：2

BGPInstance

高防包资产实例信息

被如下接口引用：DescribeListBGPInstances。

名称	类型	描述
InstanceDetail	InstanceRelation	资产实例的详细信息 示例值：{"EipList":["175.27.164.158"],"InstanceId":"bgp-0000015a"}
SpecificationLimit	BGPInstanceSpecification	资产实例的规格信息 示例值： {"ProtectBandwidth":30000,"ProtectCountLimit":10,"ProtectIPNumbe

名称	类型	描述
Usage	BGPInstanceUsages	资产实例的使用统计信息 示例值： {"ProtectCountUsage":0,"ProtectIPNumberUsage":1,"Last7DayAttackC
Region	RegionInfo	资产实例所在的地域 示例值：{"Region":"ap-nanjing"}
Status	String	资产实例的防护状态，状态码如下： "idle"：正常状态(无攻击) "attacking"：攻击中 "blocking"：封堵中 "creating"：创建中 "deblocking"：解封中 "isolate"：回收隔离中 示例值：idle
CreatedTime	Timestamp	购买时间 示例值：2021-04-25 23:04:10
ExpiredTime	Timestamp	到期时间 示例值：2021-07-25 23:04:10
Name	String	资产实例的名称 示例值：“id-123”
PackInfo	PackInfo	资产实例所属的套餐包信息， 注意：当资产实例不是套餐包的实例时，此字段为null 注意：此字段可能返回 null，表示取不到有效值。 示例值：null
EipProductInfos	Array of EipProductInfo	高防包绑定的EIP属于的云产品信息 示例值：[{"Ip":"175.27.164.158","BizType":"public","DeviceType":"cvm","I
BoundStatus	String	高防包绑定状态，取值["idle"：绑定已完成 "bounding"：正在绑定中 "failed"：绑定失败] 示例值：idle
DDoSLevel	String	四层防护严格级别 示例值：defalut
CCEnable	Integer	CC防护开关 示例值：1
TagInfoList	Array of TagInfo	资源关联标签 示例值：null
IpCountNewFlag	Integer	新版本1ip高防包 示例值：1
VitalityVersion	Integer	攻击封堵套餐标记 示例值：1
Line	Integer	网络线路 示例值：1
FreeServiceBandwidth	Integer	不计费的业务带宽 示例值：10
ElasticServiceBandwidth	Integer	弹性业务带宽开关 示例值：1
GiftServiceBandWidth	Integer	赠送的业务带宽

名称	类型	描述
		示例值：50
ModifyTime	String	修改时间 示例值：2024.10.1
BasicPlusFlag	Integer	是否是基础防护加强版 0: 不是 1: 是 示例值：1
PlanCntFlag	Integer	是否标准版2.0 0: 包含标准版2.0 1: 只查询标准版2.0 2: 不查标准版2.0 示例值：1
TransRegionFlag	Integer	是否跨区域产品 0: 不包含跨区域产品 1: 中国大陆跨区域产品 2: 非中国大陆跨区域产品 示例值：1
SuperPackFlag	Integer	是否为超级高防包 示例值：1
Zoneld	Integer	所属Zoneld 示例值：1

BGPInstanceSpecification

高防包资产实例的规格信息

被如下接口引用：DescribeListBGPIstances。

名称	类型	描述
ProtectBandwidth	Integer	保底防护峰值，单位Gbps 示例值：1
ProtectCountLimit	Integer	防护次数，单位次 示例值：10
ProtectIPNumberLimit	Integer	防护IP数，单位个 示例值：10
AutoRenewFlag	Integer	自动续费状态，取值[0：没有开启自动续费 1：开启了自动续费] 示例值：0
UnionPackFlag	Integer	联合产品标记，0代表普通高防包，1代表联合高防包 示例值：0
ServiceBandWidth	Integer	业务带宽 示例值：0
BattleEditionFlag	Integer	战斗服版本标记，0表示普通高防包，1表示战斗服高防包 示例值：0
ChannelEditionFlag	Integer	渠道版标记，0表示普通高防包，1表示渠道版高防包 示例值：0
EnterpriseFlag	Integer	高防包企业版标记，0表示普通高防包；1表示企业版高防包 示例值：0
ElasticLimit	Integer	高防包企业版弹性阈值，0表示未开启；大于0为弹性防护阈值 示例值：0
DownGradeProtect	Integer	降配后的防护能力，单位Gbps 示例值：1

BGPInstanceUsages

高防包资产实例的使用信息统计

被如下接口引用：DescribeListBGPInstances。

名称	类型	描述
ProtectCountUsage	Integer	已使用的防护次数，单位次 示例值：1
ProtectIPNumberUsage	Integer	已防护的IP数，单位个 示例值：1
Last7DayAttackCount	Integer	最近7天的攻击次数，单位次 示例值：1

BlackWhitelPRelation

黑白名单IP

被如下接口引用：DescribeListBlackWhitelPList。

名称	类型	描述
Ip	String	IP地址 示例值：1.1.2.1
Type	String	IP类型，取值[black(黑IP)，white(白IP)] 示例值：black
InstanceDetailList	Array of InstanceRelation	黑白IP所属的实例
Mask	Integer	ip掩码，0表示32位完整ip 示例值：0
ModifyTime	String	修改时间 示例值：2021-11-23 14:56:59

BoundIpInfo

高防包绑定IP对象

被如下接口引用：CreateBoundIP。

名称	类型	必选	描述
Ip	String	是	IP地址 示例值：1.1.1.1
BizType	String	否	绑定的产品分类，绑定操作为必填项，解绑操作可不填。取值[public（CVM、CLB产品），bm（黑石产品），eni（弹性网卡），vpngw（VPN网关），natgw（NAT网关），waf（Web应用安全产品），fpc（金融产品），gaap（GAAP产品），other(托管IP)] 示例值：public
InstanceId	String	否	IP所属的资源实例ID，绑定操作为必填项，解绑操作可不填。例如是弹性网卡的IP，则InstanceId填写弹性网卡的ID(eni-*)；如果绑定的是托管IP没有对应的资源实例ID，请填写"none"； 示例值：none
DeviceType	String	否	产品分类下的子类型，绑定操作为必填项，解绑操作可不填。取值[cvm（CVM），lb（负载均衡器），eni（弹性网卡），vpngw（VPN），natgw（NAT），waf（WAF），fpc（金融），gaap（GAAP），other（托管IP），eip（弹性公网常规IP）] 示例值：cvm
IspCode	Integer	否	运营商，绑定操作为必填项，解绑操作可不填。0：电信；1：联通；2：移动；5：BGP 示例值：5

名称	类型	必选	描述
Domain	String	否	域名化资产对应的域名 示例值：www.abc.com

CCLevelPolicy

CC分级策略

被如下接口引用：DescribeCCLevelList。

名称	类型	描述
InstanceId	String	实例Id 示例值：bgp-xxx
Ip	String	Ip 示例值：192.168.1.1
Protocol	String	协议 示例值：http
Domain	String	域名 示例值：www.a.com
Level	String	防护等级，可取值default表示默认策略，loose表示宽松，strict表示严格 示例值：1
CreateTime	Timestamp	创建时间 示例值：2024-01-01
ModifyTime	Timestamp	修改时间 示例值：2024-01-01

CCPrecisionPlyRecord

CC精准防护配置项

被如下接口引用：CreateCCPrecisionPolicy, DescribeCCPrecisionPlyList, ModifyCCPrecisionPolicy。

名称	类型	必选	描述
FieldType	String	是	配置项类型，当前仅支持value 示例值：value
FieldName	String	是	配置字段，可取值cgi, ua, cookie, referer, accept, srcip 示例值：srcip
Value	String	是	配置取值 示例值：1.2.3.4
ValueOperator	String	是	配置项值对比方式，可取值equal：相等，not_equal：不相等，include：包含 示例值：equal

CCPrecisionPolicy

CC精准防护策略信息

被如下接口引用：DescribeCCPrecisionPlyList。

名称	类型	描述
PolicyId	String	策略Id 示例值：id-xxx

名称	类型	描述
InstanceId	String	实例Id 示例值：bgp-xxx
Ip	String	Ip地址 示例值：1.1.1.1
Protocol	String	协议 示例值：http
Domain	String	域名 示例值：www.a.com
PolicyAction	String	策略方式（丢弃或验证码） 示例值：drop
PolicyList	Array of CCPrecisionPlyRecord	策略列表 示例值：list
CreateTime	Timestamp	创建时间 示例值：2024-01-01
ModifyTime	Timestamp	修改时间 示例值：2024-01-01

CCReqLimitPolicy

CC频率限制策略

被如下接口引用：DescribeCCReqLimitPolicyList。

名称	类型	描述
PolicyId	String	策略Id 示例值：id-xxx
InstanceId	String	实例Id 示例值：bgp-xxx
Ip	String	Ip地址 示例值：1.1.1.1
Protocol	String	协议，可取值HTTP，HTTPS 示例值：http
Domain	String	域名 示例值：www.a.com
PolicyRecord	CCReqLimitPolicyRecord	策略项 示例值：record
CreateTime	Timestamp	创建时间 示例值：2024-01-01
ModifyTime	Timestamp	修改时间 示例值：2024-01-01

CCReqLimitPolicyRecord

CC频率限制策略项字段

被如下接口引用：CreateCCReqLimitPolicy, DescribeCCReqLimitPolicyList, ModifyCCReqLimitPolicy。

名称	类型	必选	描述
Period	Integer	是	统计周期，可取值1，10，30，60，单位秒 示例值：1
RequestNum	Integer	是	请求数，取值1~20000 示例值：1
Action	String	是	频率限制策略方式，可取值alg表示验证码，drop表示丢弃 示例值：drop
ExecuteDuration	Integer	是	频率限制策略时长，可取值1~86400，单位秒 示例值：1
Mode	String	是	策略项比对方式，可取值include表示包含，equal表示等于 示例值：include
Uri	String	否	Uri，三个策略项仅可填其中之一 示例值：www.a.com
UserAgent	String	否	User-Agent，三个策略项仅可填其中之一 示例值：Agent
Cookie	String	否	Cookie，三个策略项仅可填其中之一 示例值：Cookie

CCThresholdPolicy

CC清洗阈值策略

被如下接口引用：DescribeCCThresholdList。

名称	类型	描述
InstanceId	String	实例Id 示例值：bgp-0000000a
Ip	String	Ip地址 示例值：1.1.1.1
Protocol	String	协议 示例值：http
Domain	String	域名 示例值：www.test.com
Threshold	Integer	清洗阈值 示例值：1000
CreateTime	Timestamp	创建时间 示例值：2021-06-16 21:00:44
ModifyTime	Timestamp	修改时间 示例值：2021-06-17 14:38:12

CcBlackWhitelistPolicy

CC四层黑白名单列表

被如下接口引用：DescribeCcBlackWhitelist。

名称	类型	描述
PolicyId	String	策略Id 示例值：ccBlackWhite-0000000x

名称	类型	描述
InstanceId	String	实例Id 示例值: bgpip-0000011x
Ip	String	IP地址 示例值: 1.1.1.1
Domain	String	域名 示例值: www.xx.com
Protocol	String	协议 示例值: http
Type	String	IP类型, 取值[black(黑名单IP), white(白名单IP)] 示例值: black
BlackWhitelP	String	黑白名单IP地址 示例值: 1.1.1.1
Mask	Integer	掩码 示例值: 24
CreateTime	String	创建时间 示例值: 2021-10-12 00:00:00
ModifyTime	String	修改时间 示例值: 2021-10-12 00:00:00

CcGeoIPBlockConfig

DDoS防护的区域封禁配置

被如下接口引用: CreateCcGeoIPBlockConfig, DeleteCcGeoIPBlockConfig。

名称	类型	必选	描述
RegionType	String	是	区域类型, 取值[oversea(境外) china(国内) customized(自定义地区)] 示例值: oversea
Action	String	是	封禁动作, 取值[drop(拦截) alg(人机校验)] 示例值: drop
Id	String	否	配置ID, 配置添加成功后生成; 添加新配置时不用填写此字段, 修改或删除配置时需要填写配置ID 示例值: ccGeoip-00000111
AreaList	Array of Integer	否	当RegionType为customized时, 必须填写AreaList; 当RegionType为china或oversea时, AreaList为空 示例值: 100025

CcGeoIpPolicyNew

CC地域封禁列表详情

被如下接口引用: DescribeCcGeoIPBlockConfigList。

名称	类型	描述
PolicyId	String	策略Id 示例值: rule-0000011x
InstanceId	String	实例Id 示例值: bgpip-0000011x
Ip	String	IP地址 示例值: 1.1.1.1
Domain	String	域名 示例值: www.xx.com
Protocol	String	协议, 可取值HTTP, HTTPS 示例值: http
Action	String	用户动作, drop或alg 示例值: drop
RegionType	String	地域类型, 分为china, oversea与customized 示例值: china
AreaList	Array of Integer	用户选择封禁的地域ID列表 示例值: [0,1]
CreateTime	String	创建时间 示例值: 2021-10-12 00:00:00
ModifyTime	String	修改时间 示例值: 2021-10-12 00:00:00

CertIdInsL7Rules

使用证书的规则集合

被如下接口引用: DescribeL7RulesBySSLCertId。

名称	类型	描述
L7Rules	Array of InsL7Rules	使用证书的规则列表
CertId	String	证书ID 示例值: id-xxx

ConnectLimitConfig

连接抑制相关配置

被如下接口引用: CreateDDoSConnectLimit, DescribeDDoSConnectLimitList。

名称	类型	必选	描述
SdNewLimit	Integer	是	基于源IP+目的IP的每秒新建数限制 示例值: 1
DstNewLimit	Integer	是	基于目的IP的每秒新建数限制 示例值: 1
SdConnLimit	Integer	是	基于源IP+目的IP的并发连接控制 示例值: 1
DstConnLimit	Integer	是	基于目的IP+目的端口的并发连接控制 示例值: 1

名称	类型	必选	描述
BadConnThreshold	Integer	是	基于连接抑制触发阈值，取值范围[0,4294967295] 示例值：1
NullConnEnable	Integer	是	异常连接检测条件，空连接防护开关，，取值范围[0,1] 示例值：1
ConnTimeout	Integer	是	异常连接检测条件，连接超时，，取值范围[0,65535] 示例值：1
SynRate	Integer	是	异常连接检测条件，syn占比ack百分比，，取值范围[0,100] 示例值：1
SynLimit	Integer	是	异常连接检测条件，syn阈值，取值范围[0,100] 示例值：1

ConnectLimitRelation

连接抑制列表

被如下接口引用：DescribeDDoSConnectLimitList。

名称	类型	描述
ConnectLimitConfig	ConnectLimitConfig	连接抑制配置
InstanceDetailList	Array of InstanceRelation	连接抑制关联的实例信息

DDoSAIRelation

DDoS防护的AI防护开关

被如下接口引用：DescribeListDDoSAI。

名称	类型	描述
DDoSAI	String	AI防护开关，取值[on(开启) off(关闭)] 示例值：on
InstanceDetailList	Array of InstanceRelation	AI防护开关所属的资源实例 示例值：bgp-xxx

DDoSGeoIPBlockConfig

DDoS防护的区域封禁配置

被如下接口引用：CreateDDoSGeoIPBlockConfig, DeleteDDoSGeoIPBlockConfig, DescribeListDDoSGeoIPBlockConfig, ModifyDDoSGeoIPBlockConfig。

名称	类型	必选	描述
RegionType	String	是	区域类型，取值[oversea(境外) china(国内) customized(自定义地区)] 示例值：oversea
Action	String	是	封禁动作，取值[drop(拦截)

名称	类型	必选	描述
			trans(放行)] 示例值：drop
Id	String	否	配置ID，配置添加成功后生成；添加新配置时不用填写此字段，修改或删除配置时需要填写配置ID 示例值：0
AreaList	Array of Integer	否	当RegionType为customized时，必须填写AreaList，且最多填写128个； 示例值：100025

DDoSGeoIPBlockConfigRelation

DDoS区域封禁配置相关信息

被如下接口引用：DescribeListDDoSGeoIPBlockConfig。

名称	类型	描述
GeoIPBlockConfig	DDoSGeoIPBlockConfig	DDoS区域封禁配置
InstanceDetailList	Array of InstanceRelation	配置所属的资源实例

DDoSSpeedLimitConfig

DDoS访问限速配置

被如下接口引用：CreateDDoSSpeedLimitConfig, DeleteDDoSSpeedLimitConfig, DescribeListDDoSSpeedLimitConfig, ModifyDDoSSpeedLimitConfig。

名称	类型	必选	描述
Mode	Integer	是	限速模式，取值[1(基于源IP限速) 2(基于目的端口限速)] 示例值：1
SpeedValues	Array of SpeedValue	是	限速值，每种类型的限速值最多支持1个；该字段数组至少有一种限速值 示例值：1
DstPortScopes	Array of PortSegment	否	此字段已弃用，请填写新字段DstPortList。 示例值：DstPortList
Id	String	否	配置ID，配置添加成功后生成；添加新限制配置时不用填写此字段，修改或删除限速配置时需要填写配置ID 示例值：id-xx
ProtocolList	String	否	IP protocol numbers, 取值[ALL(所有协议) TCP(tcp协议) UDP(udp协议) SMP(smp协议) 1;2-100(自定义协议号范围,最多8个)] 注意：当自定义协议号范围时，只能填写协议号，多个范围;分隔；当填写ALL时不能再填写其他协议或协议号。 示例值：1.1.1.1
DstPortList	String	否	端口范围列表，最多8个，多个;分隔，范围表示用-；此端口范围必须填写；填写样式1:0-65535，样式2:80;443;1000-2000 示例值：80

DDoSSpeedLimitConfigRelation

DDoS访问限速配置相关信息

被如下接口引用：DescribeListDDoSSpeedLimitConfig。

名称	类型	描述
SpeedLimitConfig	DDoSSpeedLimitConfig	DDoS访问限速配置
InstanceDetailList	Array of InstanceRelation	配置所属的资源实例

DefaultAlarmThreshold

单IP默认告警阈值配置

被如下接口引用：CreateDefaultAlarmThreshold, DescribeDefaultAlarmThreshold。

名称	类型	必选	描述
AlarmType	Integer	否	告警阈值类型，取值[1(入流量告警阈值) 2(攻击清洗流量告警阈值)] 示例值：1
AlarmThreshold	Integer	否	告警阈值，单位Mbps，取值>=0；当作为输入参数时，设置0会删除告警阈值配置； 示例值：10

EipAddressPackRelation

Anycast高防套餐详情

被如下接口引用：DescribeListBGPIPIstances。

名称	类型	描述
IpCount	Integer	套餐IP数量 示例值：1
AutoRenewFlag	Integer	自动续费标记 示例值：0
CurDeadline	Timestamp	当前到期时间 示例值：2021-03-17 12:32:12

EipAddressRelation

高防弹性公网IP关联信息

被如下接口引用：DescribeListBGPIPIstances。

名称	类型	描述
EipAddressRegion	String	高防弹性公网IP绑定的实例地区，例如hk代表中国香港 示例值：hk
EipBoundRscIns	String	绑定的资源实例ID。可能是一个CVM。 示例值：ins-xx
EipBoundRscEni	String	绑定的弹性网卡ID 示例值：eni-xx
EipBoundRscVip	String	绑定的资源内网ip 示例值：x.x.x.x

名称	类型	描述
ModifyTime	String	修改时间 示例值：2021-01-01 00:00:00

EipProductInfo

EIP所属的云产品信息

被如下接口引用：DescribeListBGPInstances。

名称	类型	必选	描述
Ip	String	是	IP地址 示例值：1.1.1.1
BizType	String	是	云产品类型，取值[public（CVM产品）， bm（黑石产品）， eni（弹性网卡）， vpngw（VPN网关）， natgw（NAT网关）， waf（Web应用安全产品）， fpc（金融产品）， gaap（GAAP产品）， other(托管IP)] 示例值：cvm
DeviceType	String	是	云产品类型，取值[cvm（CVM），lb（负载均衡器），eni（弹性网卡），vpngw（VPN），natgw（NAT），waf（WAF），fpc（金融），gaap（GAAP），other（托管IP），eip（黑石弹性IP）] 示例值：cvm
InstanceId	String	是	IP所属的云产品实例ID，例如是弹性网卡的IP，InstanceId为弹性网卡的ID(eni-*); 如果是托管IP没有对应的资源实例ID,InstanceId为"" 示例值：id-xxx
Domain	String	否	域名化资产对应的域名 示例值：www.abc.com

ForwardListener

转发监听器

被如下接口引用：CreateWaterPrintConfig。

名称	类型	必选	描述
FrontendPort	Integer	是	转发监听端口下限，取值1~65535 示例值：80
ForwardProtocol	String	是	转发协议，取值[TCP UDP] 示例值：TCP
FrontendPortEnd	Integer	否	转发监听端口上限，取值1~65535 示例值：8080

HttpStatusMap

业务流量的http状态码聚合数据

被如下接口引用：DescribeBizHttpStatus。

名称	类型	描述
SourceHttp2xx	Array of Float	http2xx回源状态码 示例值：[1,2,3]
Http5xx	Array of Float	http5xx状态码 示例值：[1,2,3]
SourceHttp5xx	Array of Float	http5xx回源状态码 示例值：[1,2,3]
SourceHttp404	Array of Float	http404回源状态码 示例值：[1,2,3]
Http4xx	Array of Float	http4xx状态码 示例值：[1,2,3]
SourceHttp4xx	Array of Float	http4xx回源状态码 示例值：[1,2,3]
Http2xx	Array of Float	http2xx状态码 示例值：[1,2,3]
Http404	Array of Float	http404状态码 示例值：[1,2,3]
SourceHttp3xx	Array of Float	http3xx回源状态码 示例值：[1,2,3]
Http3xx	Array of Float	http3xx状态码 示例值：[1,2,3]

IPAlarmThresholdRelation

单IP告警阈值配置

被如下接口引用：CreateIPAlarmThresholdConfig, DescribeListIPAlarmConfig。

名称	类型	必选	描述
AlarmType	Integer	是	告警阈值类型，取值[1(入流量告警阈值) 2(攻击清洗流量告警阈值)] 示例值：1
AlarmThreshold	Integer	是	告警阈值，单位Mbps，取值>=0；当作为输入参数时，设置0会删除告警阈值配置； 示例值：1
InstanceDetailList	Array of InstanceRelation	是	告警阈值所属的资源实例 示例值：bgp-xxx

IPLineInfo

IP线路信息

被如下接口引用：DescribeListSchedulingDomain。

名称	类型	必选	描述
Type	String	是	IP线路类型，取值["bgp": BGP线路IP "ctcc": 电信线路IP "cucc": 联通线路IP

名称	类型	必选	描述
			"cmcc": 移动线路IP "abroad": 境外线路IP] 示例值: bgp
Eip	String	是	线路IP 示例值: 1.1.1.1
Cname	String	否	实例对应的cname 示例值: abcdefg
ResourceFlag	Integer	否	资源flag, 0: 高防包资源, 1: 高防IP资源, 2: 非高防资源IP 示例值: 0
Domain	String	否	域名化资产对应的域名 示例值: www.abc.com

InsL7Rules

实例7层规则

被如下接口引用: CreateL7RuleCerts, DescribeL7RulesBySSLCertId。

名称	类型	必选	描述
Status	Integer	是	规则在中间状态不可修改, 只可在 (0 , 2 , 8) 状态可编辑。 规则状态, 0: 正常运行中, 1: 配置规则中(配置生效中), 2: 配置规则失败 (配置生效失败), 3: 删除规则中(删除生效中), 5: 删除规则失败(删除失败), 6: 等待添加规则, 7: 等待删除规则, 8: 等待上传证书, 9: 规则对应的资源不存在, 被隔离, 10:等待修改规则, 11:配置修改中 示例值: 8
Domain	String	是	域名 示例值: www.xxx.com
Protocol	String	是	协议 示例值: https
InsId	String	是	实例ID 示例值: bgpip-000001eo
AppId	String	是	用户AppID 示例值: 251008915
VirtualPort	String	是	高防端口 示例值: 443
SSLId	String	是	证书ID 示例值: kneSCNrZ

InstanceRelation

资源实例IP信息

被如下接口引用: CreateIPAlarmThresholdConfig, DescribeDDoSConnectLimitList, DescribeListBGPIPInstances, DescribeListBGPIInstances, DescribeListBlackWhitelPList, DescribeListDDoSAI, DescribeListDDoSGeoIPBlockConfig, DescribeListDDoSSpeedLimitConfig, DescribeListIPAlarmConfig, DescribeListListener, DescribeListPacketFilterConfig, DescribeListPortAclList, DescribeListProtocolBlockConfig, DescribeListWaterPrintConfig。

名称	类型	必选	描述
EipList	Array of String	是	资源实例的IP 示例值: [1.1.1.1]

名称	类型	必选	描述
InstanceId	String	是	资源实例的ID 示例值：bgpip-0000011x

IpBlockData

IP封堵记录

被如下接口引用：DescribeIpBlockList。

名称	类型	描述
Status	String	状态（Blocked：被封堵；UnBlocking：解封中；UnBlockFailed：解封失败） 示例值：Blocked
Ip	String	资源IP 示例值：1.1.1.1
BlockTime	Timestamp	封堵时间 示例值：2024.10.1
UnBlockTime	Timestamp	解封时间（预计解封时间） 示例值：2024.10.2
ActionType	String	解封类型（user：自助解封；auto：自动解封；update：升级解封；bind：绑定高防包解封） 示例值：user
ProtectFlag	Integer	高防标记，0：非高防，1：高防 示例值：10000

IpSegment

ip段数据结构

被如下接口引用：CreateCcBlackWhitelist, CreateDDoSBlackWhitelist, DeleteDDoSBlackWhitelist, DescribeDDoSBlackWhitelist, ModifyCcBlackWhitelist, ModifyDDoSBlackWhitelist。

名称	类型	必选	描述
Ip	String	是	ip地址 示例值：1.1.2.1
Mask	Integer	是	ip掩码，如果为32位ip，填0 示例值：0

KeyValue

字段值，K-V形式

被如下接口引用：DescribeBasicDeviceStatus, DescribeNewL7RulesErrHealth。

名称	类型	描述
Key	String	字段名称 示例值：key
Value	String	字段取值 示例值：value

L4RuleSource

L4规则回源列表

被如下接口引用：CreateNewL7Rules, DescribeBGPIPL7Rules, DescribeNewL7Rules, ModifyNewDomainRules。

名称	类型	必选	描述
Source	String	是	回源IP或域名 示例值：source
Weight	Integer	是	权重值，取值[0,100]，暂不支持 示例值：100
Port	Integer	否	8000 注意：此字段可能返回 null，表示取不到有效值。 示例值：8000
Backup	Integer	否	备份源站，1: 备份源站，0: 普通源站 示例值：backup

L7RuleEntry

L7规则

被如下接口引用：CreateNewL7Rules。

名称	类型	必选	描述
KeepTime	Integer	是	会话保持时间，单位秒 示例值：1
Domain	String	是	转发域名 示例值：www.abc.com
Protocol	String	是	转发协议，取值[http, https] 示例值：80
SourceType	Integer	是	回源方式，取值[1(域名回源)，2(IP回源)] 示例值：1
LbType	Integer	是	负载均衡方式，取值[1(加权轮询)] 示例值：1
SourceList	Array of L4RuleSource	是	回源列表
KeepEnable	Integer	是	会话保持开关，取值[0(会话保持关闭)，1(会话保持开启)] 示例值：1
Status	Integer	否	规则状态，取值[0(规则配置成功)，1(规则配置生效中)，2(规则配置失败)，3(规则删除生效中)，5(规则删除失败)，6(规则等待配置)，7(规则等待删除)，8(规则待配置证书)] 示例值：1
RuleId	String	否	规则ID，当添加新规则时可以不用填写此字段；当修改或者删除规则时需要填写此字段； 示例值：id-xxx
CCThreshold	Integer	否	HTTPS协议的CC防护阈值 示例值：1
PrivateKey	String	否	当证书来源为自有证书时，此字段必须填写证书密钥；(因已不再支持自有证书，此字段已弃用，请不用填写此字段) 示例值：key
CCEnable	Integer	否	HTTPS协议的CC防护状态，取值[0(关闭)，1(开启)] 示例值：1
HttpsToHttpEnable	Integer	否	是否开启Https协议使用Http回源，取值[0(关闭)，1(开启)]，不填写默认是关闭 示例值：1

名称	类型	必选	描述
CertType	Integer	否	证书来源，当转发协议为https时必须填，取值[2(腾讯云托管证书)]，当转发协议为http时也可以填0 示例值：1
Cert	String	否	当证书来源为自有证书时，此字段必须填写证书内容；(因已不再支持自有证书，此字段已弃用，请不用填写此字段) 示例值：cert
CCLevel	String	否	HTTPS协议的CC防护等级 示例值：1
RuleName	String	否	规则描述 示例值：name
CCStatus	Integer	否	cc防护状态，取值[0(关闭), 1(开启)] 示例值：1
VirtualPort	Integer	否	接入端口值 示例值：80
SSLId	String	否	当证书来源为腾讯云托管证书时，此字段必须填写托管证书ID 示例值：id-xxx
Id	String	否	同ruleId 示例值：id-xxx
CCAIEnable	Integer	否	智能cc开关，取值[0(关闭), 1(开启)] 示例值：1

L7RuleHealth

L7规则健康检查参数

被如下接口引用：DescribeBGPIPL7Rules, DescribeNewL7Rules。

名称	类型	必选	描述
Status	Integer	是	配置状态，0：正常，1：配置中，2：配置失败 示例值：1
Enable	Integer	是	=1表示开启；=0表示关闭 示例值：1
RuleId	String	是	规则ID 示例值：id-xxx
Url	String	是	检查目录的URL，默认为/ 示例值：url
Interval	Integer	是	检测间隔时间，单位秒 示例值：1
AliveNum	Integer	是	健康阈值，单位次 示例值：10
KickNum	Integer	是	不健康阈值，单位次 示例值：2
Method	String	是	HTTP请求方式，取值[HEAD,GET] 示例值：GET
StatusCode	Integer	是	健康检查判定正常状态码，1xx=1, 2xx=2, 3xx=4, 4xx=8, 5xx=16，多个状态码值加和 示例值：1

名称	类型	必选	描述
ProtocolFlag	Integer	否	是否同时下发http和https规则健康检查配置 示例值：1
PassiveEnable	Integer	否	被动探测开关，=1表示开启；=0表示关闭 示例值：1
BlockInter	Integer	否	被动探测不健康屏蔽时间 示例值：1
FailedCountInter	Integer	否	被动探测不健康统计间隔 示例值：1
FailedThreshold	Integer	否	被动探测不健康阈值 示例值：1
PassiveStatusCode	Integer	否	被动探测判定正常状态码，1xx=1, 2xx=2, 3xx=4, 4xx=8, 5xx=16，多个状态码值加和 示例值：1
PassiveStatus	Integer	否	被动探测配置状态，0：正常，1：配置中，2：配置失败 示例值：1

Layer4Rule

4层转发规则

被如下接口引用：DescribeListListener。

名称	类型	必选	描述
BackendPort	Integer	是	源站端口，取值1~65535 示例值：80
FrontendPort	Integer	是	转发端口，取值1~65535 示例值：80
Protocol	String	是	转发协议，取值[TCP(TCP协议) UDP(UDP协议)] 示例值：TCP
RealServers	Array of SourceServer	是	源站列表 示例值：xxx
InstanceDetails	Array of InstanceRelation	是	资源实例 示例值：xxx
InstanceDetailRule	Array of RuleInstanceRelation	否	规则所属的资源实例 示例值：xxx

Layer7Rule

7层转发规则

被如下接口引用：DescribeListListener。

名称	类型	必选	描述
Domain	String	是	域名 示例值：www.baidu.com
ProxyTypeList	Array of ProxyTypeInfo	是	转发类型列表 示例值：1

名称	类型	必选	描述
RealServers	Array of SourceServer	是	源站列表 示例值：1.1.1.1
InstanceDetails	Array of InstanceRelation	是	资源实例 示例值：detail
InstanceDetailRule	Array of RuleInstanceRelation	否	规则所属的资源实例 示例值：detail
Protocol	String	否	协议 示例值：http
Vport	Integer	否	端口号 示例值：80

NewL7RuleEntry

L7规则

被如下接口引用：DescribeNewL7Rules, ModifyNewDomainRules。

名称	类型	必选	描述
Protocol	String	是	转发协议，取值[http, https] 示例值：http
Domain	String	是	转发域名 示例值：www.abc.com
LbType	Integer	是	负载均衡方式，取值[1(加权轮询)] 示例值：1
KeepEnable	Integer	是	会话保持开关，取值[0(会话保持关闭)，1(会话保持开启)] 示例值：1
KeepTime	Integer	是	会话保持时间，单位秒 示例值：15
SourceType	Integer	是	回源方式，取值[1(域名回源)，2(IP回源)] 示例值：1
SourceList	Array of L4RuleSource	是	回源列表
Region	Integer	否	区域码 示例值：1
Id	String	否	资源Id 示例值：id-xxx
Ip	String	否	资源Ip 示例值：1.1.1.1
RuleId	String	否	规则ID，当添加新规则时可以用不用填写此字段；当修改或者删除规则时需要填写此字段； 示例值：id-xxx
RuleName	String	否	规则描述 示例值：name
CertType	Integer	否	证书来源，当转发协议为https时必须填，取值[2(腾讯云托管证书)]，当转发协议为http时也可以填0 示例值：1
SSLId	String	否	当证书来源为腾讯云托管证书时，此字段必须填写托管证书ID 示例值：id-xx

名称	类型	必选	描述
Cert	String	否	当证书来源为自有证书时，此字段必须填写证书内容；（因已不再支持自有证书，此字段已弃用，请不用填写此字段） 示例值：cert
PrivateKey	String	否	当证书来源为自有证书时，此字段必须填写证书密钥；（因已不再支持自有证书，此字段已弃用，请不用填写此字段） 示例值：key
Status	Integer	否	规则状态，取值[0(规则配置成功)，1(规则配置生效中)，2(规则配置失败)，3(规则删除生效中)，5(规则删除失败)，6(规则等待配置)，7(规则等待删除)，8(规则待配置证书)] 示例值：1
CCStatus	Integer	否	cc防护状态，取值[0(关闭)，1(开启)] 示例值：1
CCEnable	Integer	否	HTTPS协议的CC防护状态，取值[0(关闭)，1(开启)] 示例值：1
CCThreshold	Integer	否	HTTPS协议的CC防护阈值（已废弃） 示例值：1
CCThresholdNew	Integer	否	HTTPS协议的CC防护阈值 -1：默认防御阈值 0：关闭 大于0：自定义防护阈值 示例值：1
CCLevel	String	否	HTTPS协议的CC防护等级 示例值：loose
ModifyTime	Timestamp	否	修改时间 示例值：2024.10.01
HttpsToHttpEnable	Integer	否	是否开启Https协议使用Http回源，取值[0(关闭)，1(开启)]，不填写默认是关闭 示例值：1
VirtualPort	Integer	否	接入端口值 示例值：80
RewriteHttps	Integer	否	http强制跳转https，1表示打开，0表示关闭 示例值：1
ErrCode	Integer	否	规则配置失败时的详细错误原因(仅当Status=2时有效)，1001证书不存在，1002证书获取失败，1003证书上传失败，1004证书已过期 示例值：100
Version	Integer	否	版本 示例值：2

OverviewDDoSEvent

防护概览DDoS攻击事件

被如下接口引用：DescribeOverviewDDoSEventList。

名称	类型	描述
Id	String	事件Id 示例值：ddosev-00000424
Vip	String	ip 示例值：10.24.145.26
StartTime	String	开始时间 示例值：2021-03-17 12:32:12

名称	类型	描述
EndTime	String	结束时间 示例值：2021-03-17 12:32:12
AttackType	String	攻击类型 示例值：SYNFLOOD
AttackStatus	Integer	攻击状态，0：攻击中；1：攻击结束 示例值：0
Mbps	Integer	攻击流量，单位Mbps 示例值：23526
Pps	Integer	攻击包量，单位pps 示例值：235623
Business	String	业务类型，bgp-multip：高防包；bgpip：高防ip；basic：基础防护 示例值：bgpip
InstanceId	String	高防实例Id 示例值：bgpip-00001001
InstanceName	String	高防实例名称 示例值：高防实例名称

PackInfo

套餐包信息

被如下接口引用：DescribeListBGPIPInstances, DescribeListBGPInstances。

名称	类型	必选	描述
PackType	String	是	套餐包的类型，取值[staticpack：高防IP三网套餐包 insurance：保险套餐包] 示例值：staticpack
PackId	String	是	套餐包的ID 示例值：bgp-xxx

PacketFilterConfig

特征过滤配置

被如下接口引用：CreatePacketFilterConfig, DeletePacketFilterConfig, DescribeListPacketFilterConfig, ModifyPacketFilterConfig。

名称	类型	必选	描述
Protocol	String	是	协议，取值[tcp udp icmp all] 示例值：tcp
SportStart	Integer	是	起始源端口，取值0~65535 示例值：1000
SportEnd	Integer	是	结束源端口，取值1~65535，必须大于等于起始源端口 示例值：1001
DportStart	Integer	是	起始目的端口，取值0~65535 示例值：1000
DportEnd	Integer	是	结束目的端口，取值1~65535，必须大于等于起始目的端口 示例值：1001

名称	类型	必选	描述
PktlenMin	Integer	是	最小报文长度，取值1~1500 示例值：300
PktlenMax	Integer	是	最大报文长度，取值1~1500，必须大于等于最小报文长度 示例值：500
Action	String	是	动作，取值[drop(丢弃) transmit(放行) drop_black(丢弃并拉黑) drop_rst(拦截)（已废弃，不支持drop_rst） drop_black_rst(拦截并拉黑)（已废弃，不支持drop_black_rst） forward(继续防护)] 示例值：drop
MatchBegin	String	否	检测位置，取值[begin_l3(IP头) begin_l4(TCP/UDP头) begin_l5(T载荷) no_match(不匹配)] 示例值：begin_l5
MatchType	String	否	检测类型，取值[sunday(关键字) pcre(正则表达式)（已废弃，仅支持sunday）] 示例值：sunday
Str	String	否	检测值，关键字字符串或正则表达式,取值[当检测类型为sunday时，请填写字符串或者16进制字节码，例如 \x313233对应的是字符串"123"的16进制字节码；最多支持63位；] 示例值：\x313233
Depth	Integer	否	从检测位置开始的检测深度，取值[0,1500] 示例值：500
Offset	Integer	否	从检测位置开始的偏移量，取值范围[0,Depth] 示例值：100
IsNot	Integer	否	是否包含检测值，取值[0(包含) 1(不包含)（已废弃，仅支持0）] 示例值：0
MatchLogic	String	否	当有第二个检测条件时，与第一检测条件的且或关系，取值[and(且的关系)（已废弃，仅支持none） none(当没有第二个检测条件时填写此值)] 示例值：none
MatchBegin2	String	否	（已废弃） 示例值：begin_l5
MatchType2	String	否	（已废弃） 示例值：sunday
Str2	String	否	（已废弃） 示例值：\x313233
Depth2	Integer	否	（已废弃） 示例值：500

名称	类型	必选	描述
Offset2	Integer	否	(已废弃) 示例值: 200
IsNot2	Integer	否	(已废弃) 示例值: 0
Id	String	否	特征过滤配置添加成功后自动生成的规则ID, 当添加新特征过滤配置时, 此字段不用填写; 当修改/删除新特征过滤配置时, 此字段必填; 示例值: id-xxx
PktLenGT	Integer	否	大于报文长度, 取值1+ 示例值: 50

PacketFilterRelation

特征过滤相关信息

被如下接口引用: DescribeListPacketFilterConfig。

名称	类型	描述
PacketFilterConfig	PacketFilterConfig	特征过滤配置
InstanceDetailList	Array of InstanceRelation	特征过滤配置所属的实例
ModifyTime	String	修改时间 示例值: 2021-11-23 14:56:59

PortSegment

端口段信息

被如下接口引用: CreateDDoSSpeedLimitConfig, DeleteDDoSSpeedLimitConfig, ModifyDDoSSpeedLimitConfig。

名称	类型	必选	描述
BeginPort	Integer	是	起始端口, 取值1~65535 示例值: 1
EndPort	Integer	是	结束端口, 取值1~65535, 必须不小于起始端口 示例值: 1

ProtocolBlockConfig

协议封禁配置

被如下接口引用: CreateProtocolBlockConfig, DescribeListProtocolBlockConfig。

名称	类型	必选	描述
DropTcp	Integer	是	TCP封禁, 取值[0(封禁关), 1(封禁开)] 示例值: 1
DropUdp	Integer	是	UDP封禁, 取值[0(封禁关), 1(封禁开)] 示例值: 1
DropIcmp	Integer	是	ICMP封禁, 取值[0(封禁关), 1(封禁开)] 示例值: 1
DropOther	Integer	是	其他协议封禁, 取值[0(封禁关), 1(封禁开)] 示例值: 1

名称	类型	必选	描述
CheckExceptNullConnect	Integer	是	异常空连接防护，取值[0(防护关)，1(防护开)] 示例值：1
PingOfDeath	Integer	否	ping of death防护，取值[0(防护关)，1(防护开)] 示例值：1
TearDrop	Integer	否	tear drop防护，取值[0(防护关)，1(防护开)] 示例值：1

ProtocolBlockRelation

协议封禁相关信息

被如下接口引用：DescribeListProtocolBlockConfig。

名称	类型	描述
ProtocolBlockConfig	ProtocolBlockConfig	协议封禁配置
InstanceDetailList	Array of InstanceRelation	协议封禁配置所属的实例

ProtocolPort

Protocol、Port参数

被如下接口引用：DescribeBizHttpStatus, DescribeBizTrend。

名称	类型	必选	描述
Protocol	String	是	协议（tcp；udp） 示例值：tcp
Port	Integer	是	端口 示例值：80

ProxyTypeInfo

转发类型

被如下接口引用：DescribeListListener。

名称	类型	必选	描述
ProxyPorts	Array of Integer	是	转发监听端口列表，端口取值1~65535 示例值：80
ProxyType	String	是	转发协议，取值[http(HTTP协议) https(HTTPS协议)] 示例值：http

RegionInfo

地域信息

被如下接口引用：DescribeListBGPIPInstances, DescribeListBGPInstances。

名称	类型	必选	描述
Region	String	是	地域名称，例如，ap-guangzhou 示例值：ap-guangzhou

RuleInstanceRelation

四七层规则对应实例与IP的关系

被如下接口引用：DescribeListListener。

名称	类型	必选	描述
EipList	Array of String	否	资源实例的IP 示例值：[1.1.1.1]
InstanceId	String	否	资源实例的ID 示例值：bgpip-0000011x
Cname	String	否	资源实例的Cname 示例值：abcdefg

SchedulingDomainInfo

调度域名信息

被如下接口引用：DescribeListSchedulingDomain。

名称	类型	描述
Domain	String	调度域名 示例值：www.abc.com
LineIPList	Array of IPLineInfo	线路IP列表 示例值：[1]
Method	String	调度方式，当前仅支持优先级的方式，取值[priority] 示例值：priority
TTL	Integer	调度域名解析记录的TTL值 示例值：1
Status	Integer	运行状态，取值[0：未运行 1：运行中 2：运行异常] 示例值：1
CreatedTime	Timestamp	创建时间 示例值：2024.12.1
ModifyTime	Timestamp	最后修改时间 示例值：2024.12.2
UsrDomainName	String	域名名称 示例值：name

SourceServer

源站信息

被如下接口引用：DescribeListListener。

名称	类型	必选	描述
RealServer	String	是	源站的地址（IP或者域名） 示例值：1.1.1.1
RsType	Integer	是	源站的地址类型，取值[1(域名地址) 2(IP地址)] 示例值：1
Weight	Integer	是	源站的回源权重，取值1~100 示例值：100
Port	Integer	否	端口号：0~65535 示例值：80

SpeedValue

限速值类型，例如：包速率pps、带宽bps

被如下接口引用：CreateDDoSSpeedLimitConfig, DeleteDDoSSpeedLimitConfig, ModifyDDoSSpeedLimitConfig。

名称	类型	必选	描述
Type	Integer	是	限速值类型，取值[1(包速率pps) 2(带宽bps)] 示例值：1
Value	Integer	是	值大小 示例值：1

StaticPackRelation

三网高防套餐详情

被如下接口引用：DescribeListBGPIPIstances。

名称	类型	描述
ProtectBandwidth	Integer	保底带宽 示例值：10
NormalBandwidth	Integer	业务带宽 示例值：1
ForwardRulesLimit	Integer	转发规则 示例值：1
AutoRenewFlag	Integer	自动续费标记 示例值：1
CurDeadline	String	到期时间 示例值："2024.12.1"

SuccessCode

操作返回码，只用于返回成功的情况

被如下接口引用：CreateBoundIP, CreateL7RuleCerts, CreateNewL7Rules, ModifyDDoSThreshold, ModifyNewDomainRules。

名称	类型	描述
Message	String	描述 示例值：msg
Code	String	成功/错误码 示例值：200

TagFilter

标签类型

被如下接口引用：DescribeListBGPIPInstances, DescribeListBGPInstances。

名称	类型	必选	描述
TagKey	String	是	标签键 示例值：232
TagValue	Array of String	是	标签键值列表 示例值：232

TagInfo

标签信息，用于资源列表返回关联的标签

被如下接口引用：DescribeListBGPIPInstances, DescribeListBGPInstances。

名称	类型	描述
TagKey	String	标签键 示例值：sdf
TagValue	String	标签值 示例值：sdfss

WaterPrintConfig

水印防护配置

被如下接口引用：CreateWaterPrintConfig, DescribeListWaterPrintConfig。

名称	类型	必选	描述
Offset	Integer	是	水印偏移量，取值范围[0, 100) 示例值：0
OpenStatus	Integer	是	是否开启，取值[0（手动开启） 1（立即运行）] 示例值：0
Listeners	Array of ForwardListener	是	水印所属的转发监听器列表 示例值：[{80,TCP}]
Keys	Array of WaterPrintKey	否	水印添加成功后生成的水印密钥列表，一条水印最少1个密钥，最多2个密钥 示例值：["key"]
Verify	String	否	水印检查模式，取值[checkall（普通模式） shortfpcheckall（精简模式）] 示例值：checkall

名称	类型	必选	描述
CloudSdkProxy	Integer	否	是否开启代理，1开启则忽略IP+端口校验；0关闭则需要IP+端口校验 示例值：1

WaterPrintKey

生成的水印密钥

被如下接口引用：CreateWaterPrintConfig。

名称	类型	必选	描述
KeyVersion	String	是	密钥版本号 示例值：1
KeyContent	String	是	密钥内容 示例值：0090-12301-1212
KeyId	String	是	密钥ID 示例值：id-xxx
KeyOpenStatus	Integer	是	密钥启用状态，只有一个取值1(启用) 示例值：1
CreateTime	Timestamp	是	密钥生成时间 示例值：2024.10.1

WaterPrintRelation

水印配置相关信息

被如下接口引用：DescribeListWaterPrintConfig。

名称	类型	描述
WaterPrintConfig	WaterPrintConfig	水印配置
InstanceDetailList	Array of InstanceRelation	水印配置所属的资源实例

错误码

最近更新时间：2024-12-20 01:04:39

功能说明

如果返回结果中存在 Error 字段，则表示调用 API 接口失败。例如：

```
{
  "Response": {
    "Error": {
      "Code": "AuthFailure.SignatureFailure",
      "Message": "The provided credentials could not be validated. Please check your signature is correct."
    },
    "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
  }
}
```

Error 中的 Code 表示错误码，Message 表示该错误的具体信息。

错误码列表

公共错误码

错误码	说明
ActionOffline	接口已下线。
AuthFailure.InvalidAuthorization	请求头部的 Authorization 不符合腾讯云标准。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。
AuthFailure.MFAFailure	MFA 错误。
AuthFailure.SecretIdNotFound	密钥不存在。请在 控制台 检查密钥是否已被删除或者禁用，如状态正常，请检查密钥是否填写正确，注意前后不得有空格。
AuthFailure.SignatureExpire	签名过期。Timestamp 和服务器时间相差不得超过五分钟，请检查本地时间是否和标准时间同步。
AuthFailure.SignatureFailure	签名错误。签名计算错误，请对照调用方式中的签名方法文档检查签名计算过程。
AuthFailure.TokenFailure	token 错误。
AuthFailure.UnauthorizedOperation	请求未授权。请参考 CAM 文档对鉴权的说明。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidAction	接口不存在。
InvalidParameter	参数错误（包括参数格式、类型等错误）。
InvalidParameterValue	参数取值错误。
InvalidRequest	请求 body 的 multipart 格式错误。
IpInBlacklist	IP 地址在黑名单中。
IpNotInWhitelist	IP 地址不在白名单中。

错误码	说明
LimitExceeded	超过配额限制。
MissingParameter	缺少参数。
NoSuchProduct	产品不存在
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
RequestLimitExceeded.GlobalRegionUinLimitExceeded	主账号超过频率限制。
RequestLimitExceeded.IPLimitExceeded	IP 限频。
RequestLimitExceeded.UinLimitExceeded	主账号限频。
RequestSizeLimitExceeded	请求包超过限制大小。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResponseSizeLimitExceeded	返回包超过限制大小。
ServiceUnavailable	当前服务暂时不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误，用户多传未定义的参数会导致错误。
UnsupportedOperation	操作不支持。
UnsupportedProtocol	http(s) 请求协议错误，只支持 GET 和 POST 请求。
UnsupportedRegion	接口不支持所传地域。

业务错误码

错误码	说明
AuthFailure	CAM签名/鉴权错误。
OperationDenied	操作被拒绝。
ResourcesSoldOut	资源售罄。

附录

RegionId列表

最近更新时间：2025-03-18 01:08:04

列表说明

当RegionType为customized时，必须填写AreaList，且最多填写128个
下列为地区名称相对应的RegionId。

RegionId	地区名称
100001	亚洲-中国北京
100002	亚洲-中国上海
100003	亚洲-中国云南
100004	亚洲-中国内蒙古
100005	亚洲-中国吉林
100006	亚洲-中国四川
100007	亚洲-中国天津
100008	亚洲-中国宁夏
100009	亚洲-中国安徽
100010	亚洲-中国山东
100011	亚洲-中国广东
100012	亚洲-中国广西
100013	亚洲-中国新疆
100014	亚洲-中国江苏
100015	亚洲-中国江西
100016	亚洲-中国河北
100017	亚洲-中国河南
100018	亚洲-中国浙江
100019	亚洲-中国海南
100020	亚洲-中国湖北
100021	亚洲-中国湖南
100022	亚洲-中国甘肃
100023	亚洲-中国福建
100024	亚洲-中国西藏
100025	亚洲-中国贵州
100026	亚洲-中国辽宁
100027	亚洲-中国重庆
100028	亚洲-中国陕西

RegionId	地区名称
100029	亚洲-中国青海
100030	亚洲-中国黑龙江
100031	亚洲-中国山西
100032	亚洲-中国台湾
100033	亚洲-中国澳门
100034	亚洲-中国香港
100035	亚洲-中国其他
100101	亚洲-也门
100102	亚洲-伊拉克
100103	亚洲-沙特阿拉伯
100104	亚洲-伊朗
100105	亚洲-叙利亚
100106	亚洲-亚美尼亚
100107	亚洲-约旦
100108	亚洲-黎巴嫩
100109	亚洲-科威特
100110	亚洲-阿曼
100111	亚洲-卡塔尔
100112	亚洲-巴林
100113	亚洲-阿联酋
100114	亚洲-以色列
100115	亚洲-土耳其
100116	亚洲-阿塞拜疆
100117	亚洲-格鲁吉亚
100118	亚洲-阿富汗
100119	亚洲-巴基斯坦
100120	亚洲-孟加拉
100121	亚洲-土库曼斯坦
100122	亚洲-塔吉克斯坦
100123	亚洲-斯里兰卡
100124	亚洲-不丹
100126	亚洲-马尔代夫
100127	亚洲-英属印度洋领地
100128	亚洲-尼泊尔

RegionId	地区名称
100129	亚洲-缅甸
100130	亚洲-乌兹别克斯坦
100131	亚洲-哈萨克斯坦
100132	亚洲-吉尔吉克斯坦
100133	亚洲-科科斯群岛
100134	亚洲-越南
100135	亚洲-泰国
100136	亚洲-印度尼西亚
100137	亚洲-老挝
100138	亚洲-菲律宾
100139	亚洲-马来西亚
100140	亚洲-文莱
100141	亚洲-柬埔寨
100142	亚洲-韩国
100143	亚洲-日本
100144	亚洲-朝鲜
100145	亚洲-新加坡
100146	亚洲-蒙古
100147	亚洲-圣诞岛
100148	亚洲-巴勒斯坦领土
100149	亚洲-阿联酋
200001	北美洲-巴巴多斯
200002	北美洲-圣皮埃尔和密克隆
200003	北美洲-格陵兰
200004	北美洲-牙买加
200005	北美洲-多米尼加共和国
200006	北美洲-古巴
200007	北美洲-马提尼克群岛
200008	北美洲-巴哈马
200009	北美洲-百慕大
200010	北美洲-安圭拉
200011	北美洲-特立尼达和多巴哥
200012	北美洲-圣基茨和尼维斯
200013	北美洲-多米尼加

RegionId	地区名称
200014	北美洲-安提瓜和巴布达
200015	北美洲-圣卢西亚
200016	北美洲-特克斯和凯科斯群岛
200017	北美洲-阿鲁巴
200018	北美洲-英属维京群岛
200019	北美洲-圣文森特和格林纳丁斯
200020	北美洲-蒙塞拉特群岛
200021	北美洲-圣马丁
200022	北美洲-圣巴泰勒米
200023	北美洲-瓜德罗普岛
200024	北美洲-格林纳达
200025	北美洲-开曼群岛
200026	北美洲-伯利兹
200027	北美洲-萨尔瓦多
200028	北美洲-危地马拉
200029	北美洲-洪都拉斯
200030	北美洲-尼加拉瓜
200031	北美洲-哥斯达黎加
200032	北美洲-巴拿马
200033	北美洲-海地
200034	北美洲-墨西哥
200035	北美洲-波多黎各
200037	北美洲-美国
200038	北美洲-圣马丁岛
200039	北美洲-库拉索
300001	南美洲-圭亚那
300002	南美洲-法属圭亚那
300003	南美洲-苏里南
300004	南美洲-巴拉圭
300005	南美洲-乌拉圭
300006	南美洲-巴西
300007	南美洲-福克兰群岛
300008	南美洲-委内瑞拉
300009	南美洲-厄瓜多尔

RegionId	地区名称
300010	南美洲-哥伦比亚
300011	南美洲-阿根廷
300012	南美洲-智利
300013	南美洲-玻利维亚
300014	南美洲-秘鲁
400001	欧洲-塞浦路斯
400002	欧洲-希腊
400003	欧洲-爱沙尼亚
400004	欧洲-拉脱维亚
400005	欧洲-立陶宛
400006	欧洲-摩尔多瓦
400008	欧洲-芬兰
400009	欧洲-奥兰群岛
400010	欧洲-乌克兰
400011	欧洲-马其顿
400012	欧洲-匈牙利
400013	欧洲-保加利亚
400014	欧洲-阿尔巴尼亚
400015	欧洲-波兰
400016	欧洲-罗马尼亚
400017	欧洲-科索沃
400019	欧洲-葡萄牙
400020	欧洲-直布罗陀
400021	欧洲-西班牙
400022	欧洲-马耳他
400023	欧洲-法罗群岛
400024	欧洲-丹麦
400025	欧洲-冰岛
400026	欧洲-英国
400027	欧洲-瑞士
400028	欧洲-瑞典
400029	欧洲-荷兰
400030	欧洲-奥地利
400031	欧洲-比利时

RegionId	地区名称
400032	欧洲-德国
400033	欧洲-卢森堡
400034	欧洲-爱尔兰
400035	欧洲-摩纳哥
400036	欧洲-法国
400037	欧洲-安道尔
400038	欧洲-列支敦士登
400039	欧洲-泽西岛
400040	欧洲-曼岛
400041	欧洲-格恩西岛
400042	欧洲-斯洛伐克
400043	欧洲-捷克
400044	欧洲-挪威
400045	欧洲-梵蒂冈
400046	欧洲-圣马力诺
400047	欧洲-意大利
400048	欧洲-斯洛文尼亚
400049	欧洲-黑山共和国
400050	欧洲-克罗地亚
400051	欧洲-波黑
400052	欧洲-塞尔维亚
500001	大洋洲-帕劳
500002	大洋洲-库克群岛
500003	大洋洲-东帝汶
500004	大洋洲-澳大利亚
500005	大洋洲-马绍尔群岛
500006	大洋洲-密克罗尼西亚
500007	大洋洲-巴布亚新几内亚
500008	大洋洲-所罗门群岛
500009	大洋洲-图瓦卢
500010	大洋洲-瑙鲁
500011	大洋洲-瓦努阿图
500012	大洋洲-新喀里多尼亚
500013	大洋洲-诺福克岛

RegionId	地区名称
500014	大洋洲-新西兰
500015	大洋洲-斐济
500016	大洋洲-法属波利尼西亚
500017	大洋洲-皮特凯恩群岛
500018	大洋洲-基里巴斯
500019	大洋洲-托克劳
500020	大洋洲-汤加
500021	大洋洲-瓦利斯和富图纳
500022	大洋洲-萨摩亚
500023	大洋洲-纽埃
500024	大洋洲-北马里亚纳群岛
500025	大洋洲-关岛
500026	大洋洲-美属萨摩亚
600001	非洲-卢旺达
600002	非洲-索马里
600003	非洲-坦桑尼亚
600004	非洲-肯尼亚
600005	非洲-扎伊尔
600006	非洲-吉布提
600007	非洲-乌干达
600008	非洲-中非
600009	非洲-塞舌尔
600010	非洲-埃塞俄比亚
600011	非洲-厄立特里亚
600012	非洲-埃及
600013	非洲-苏丹
600014	非洲-布隆迪
600015	非洲-津巴布韦
600016	非洲-赞比亚
600017	非洲-科摩罗
600018	非洲-马拉维
600019	非洲-莱索托
600020	非洲-博茨瓦纳
600021	非洲-毛里求斯

RegionId	地区名称
600022	非洲-斯威士兰
600023	非洲-留尼汪
600024	非洲-南非
600025	非洲-马约特
600026	非洲-莫桑比克
600027	非洲-马达加斯加
600028	非洲-利比亚
600029	非洲-喀麦隆
600030	非洲-塞内加尔
600031	非洲-刚果（布）
600032	非洲-利比里亚
600033	非洲-象牙海岸
600034	非洲-加纳
600035	非洲-赤道几内亚
600036	非洲-尼日利亚
600037	非洲-布基纳法索
600038	非洲-多哥
600039	非洲-几内亚比绍
600040	非洲-毛里塔尼亚
600041	非洲-贝宁
600042	非洲-加蓬
600043	非洲-塞拉利昂
600044	非洲-圣多美和普林西比
600045	非洲-冈比亚
600046	非洲-几内亚
600047	非洲-乍得
600048	非洲-尼日尔
600049	非洲-马里
600050	非洲-西撒哈拉
600051	非洲-突尼斯
600052	非洲-摩洛哥
600053	非洲-阿尔及利亚
600054	非洲-安哥拉
600055	非洲-纳米比亚

RegionId	地区名称
600056	非洲-圣赫勒拿
600057	非洲-佛得角
600058	非洲-南苏丹
600059	非洲-刚果(金)
700001	南极洲-法属南部领土
700002	南极洲-南极洲