

云联络中心 语音通话服务端 API



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

语音通话服务端 API

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

座席相关接口

创建 SDK 登录 Token

创建客服账号

删除坐席信息

修改客服账号

获取坐席信息列表

获取坐席实时状态统计指标

修改座席密码

恢复客服上线

强制客服下线

技能组相关接口

新建技能组

删除客服技能组

更新技能组

获取技能组信息列表

绑定座席所属技能组

解绑坐席所属技能组

号码相关接口

绑定号码外呼技能组

绑定号码呼入回调接口

解绑号码外呼技能组

查询号码列表

停用号码

申请运营商白名单号码

查询运营商白名单号码申请

查询生效运营商白名单规则

创建客户自携号码接入审核

修改客户自携号码审批单

双呼相关接口

创建外呼会话（当前仅支持双呼）

自动外呼相关接口

创建自动外呼任务

批量查询自动外呼任务

停止自动外呼任务

查询自动外呼任务详情

停止预测式外呼任务

创建预测式外呼任务

删除预测式外呼任务

查询座席巡航式外呼任务

查询预测式外呼任务

查询预测式外呼任务列表

查询预测式外呼呼叫列表

暂停预测式外呼任务

恢复预测式外呼任务

更新预测式外呼任务

话机相关接口

创建话机账号

删除话机账号

修改话机账号(绑定技能组、绑定坐席账号)

获取话机信息

查询话机列表信息

重置话机注册密码

电话服务记录相关接口

获取电话服务记录与录音

获取主被叫受保护的电话服务记录与录音

获取 PSTN 活动会话列表

获取呼入实时数据统计指标

获取 PSTN 会话信息

按实例获取电话消耗统计

购买相关接口

获取用户购买信息列表

外呼相关接口

停止座席巡航式外呼任务

创建座席巡航式外呼任务

CreateIVRSession

企业资质相关接口

创建企业资质申请

查询企业资质申请列表

修改企业资质申请

语音智能体相关接口

获取 AI 耗时

创建智能体通话

创建AI通话

查询 AI 通话内容提取结果

AI服务相关接口

控制AI对话

其他接口

创建用户数据签名

挂断电话

拉取会话录音转文本信息

创建管理端访问链接

查询IVR音频文件列表

上传IVR音频文件

转接人工

数据结构

错误码

语音通话服务端 API 更新历史

最近更新时间：2025-07-07 01:11:50

第 102 次发布

发布时间：2025-07-07 01:11:38

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [ServerPushText](#)
 - 新增成员：AddHistory

第 101 次发布

发布时间：2025-07-04 01:11:58

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [ControlAIConversation](#)

新增数据结构：

- [ServerPushText](#)

第 100 次发布

发布时间：2025-07-02 01:16:05

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyStaff](#)
 - 新增入参：ForwardingConfig

第 99 次发布

发布时间：2025-06-24 01:11:16

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- [ForwardingConfig](#)
- [ForwardingTarget](#)

修改数据结构：

- [StaffInfo](#)
 - 新增成员：ForwardingConfig

第 98 次发布

发布时间：2025-06-11 11:58:13

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [BindNumberCallInInterface](#)

新增数据结构：

- [ClientInfo](#)
- [Interface](#)

删除数据结构：

- Client

修改数据结构：

- [StaffStatusMetrics](#)
 - 修改成员：ClientInfo

第 97 次发布

发布时间：2025-06-10 01:13:29

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- Client

修改数据结构：

- [StaffStatusMetrics](#)
 - 新增成员：ClientInfo

第 96 次发布

发布时间：2025-04-28 01:13:29

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [AllLatencyDetail](#)
 - 新增成员：LLMFirstTokenLatency

第 95 次发布

发布时间：2025-04-25 01:12:19

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [StaffInfo](#)
 - 新增成员：RoleList
 - 废弃成员：RoleIdList

第 94 次发布

发布时间：2025-04-23 01:09:58

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：CallType

第 93 次发布

发布时间：2025-04-22 01:10:05

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DeleteCCCSkillGroup](#)

修改接口：

- [CreateAIAgentCall](#)
 - 废弃入参：PromptVariables
- [CreateAICall](#)
 - 废弃入参：PromptVariables

第 92 次发布

发布时间：2025-04-14 01:13:13

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeAILatency](#)
- [ForceMemberOffline](#)
- [RestoreMemberOnline](#)

新增数据结构：

- [AILatencyDetail](#)
- [AILatencyStatistics](#)
- [AILatencyStatisticsInfo](#)

第 91 次发布

发布时间：2025-03-31 01:09:33

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [AbortAgentCruiseDialingCampaign](#)
- [CreateAgentCruiseDialingCampaign](#)
- [DescribeAgentCruiseDialingCampaign](#)
- [TransferToManual](#)

第 90 次发布

发布时间：2025-03-28 01:13:29

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAIAgentCall](#)
 - 新增入参：Variables
- [CreateAICall](#)
 - 新增入参：Variables
 - 修改入参：SystemPrompt, Model

第 89 次发布

发布时间：2025-03-27 01:09:01

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAutoCalloutTask](#)
 - 新增入参：TimeZone, AvailableTime, AIAgentId
 - 修改入参：IvrId
- [CreatePredictiveDialingCampaign](#)
 - 新增入参：TimeZone, AvailableTime

新增数据结构：

- [TimeRange](#)

修改数据结构：

- [StaffInfo](#)
 - 新增成员：RoleIdList
 - 废弃成员：RoleId

第 88 次发布

发布时间：2025-03-11 01:13:31

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAICall](#)
 - 新增入参：Temperature

第 87 次发布

发布时间：2025-01-24 01:09:27

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeAICallExtractResult](#)

修改接口：

- [CreateAICall](#)
 - 新增入参：WelcomeMessagePriority, ExtractConfig

新增数据结构：

- [AICallExtractConfigElement](#)
- [AICallExtractResultElement](#)
- [AICallExtractResultInfo](#)

第 86 次发布

发布时间：2025-01-15 01:09:19

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateAIAgentCall](#)

修改接口：

- [CreateAICall](#)
 - 新增入参：VadSilenceTime

第 85 次发布

发布时间：2025-01-08 01:11:38

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAICall](#)
 - 新增入参：TransferFunctionEnable, TransferItems, PromptVariables

新增数据结构：

- [AITransferItem](#)

第 84 次发布

发布时间：2024-12-20 01:10:00

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyStaff](#)
 - 新增入参：ExtensionNumber

修改数据结构：

- [SeatUserInfo](#)
 - 新增成员：ExtensionNumber
- [StaffInfo](#)
 - 新增成员：ExtensionNumber

第 83 次发布

发布时间：2024-12-19 01:14:58

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAICall](#)

- 新增入参：NotifyMaxCount

第 82 次发布

发布时间：2024-11-19 01:13:32

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreatePredictiveDialingCampaign](#)
 - 新增入参：Variables, UUI, CalleeAttributes
- [UpdatePredictiveDialingCampaign](#)
 - 新增入参：Variables, UUI, CalleeAttributes

第 81 次发布

发布时间：2024-11-15 01:14:07

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [CompanyStateInfo](#)
 - 新增成员：ContractNo

第 80 次发布

发布时间：2024-11-01 01:10:08

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [CarrierPrivilegeNumberApplicant](#)
 - 修改成员：SdkAppld, ApplicantId, Callers, Callees, Description, State, CreateTime, UpdateTime

第 79 次发布

发布时间：2024-10-28 01:14:26

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeTelRecordAsr](#)

新增数据结构：

- [AsrData](#)

修改数据结构：

- [IVRKeyPressedElement](#)
 - 新增成员：Timestamp, NodeLabel, OriginalContent, TTSPrompt

第 78 次发布

发布时间：2024-10-24 01:09:50

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [OwnNumberApplyDetailItem](#)
 - 新增成员：OutboundCalleeFormat

第 77 次发布

发布时间：2024-10-16 01:05:10

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAiCall](#)
 - 新增入参：NotifyDuration, NotifyMessage, CustomTTSTConfig
 - 修改入参：VoiceType

第 76 次发布

发布时间：2024-09-30 01:10:03

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateOwnNumberApply](#)
- [ModifyOwnNumberApply](#)

新增数据结构：

- [OwnNumberApplyDetailItem](#)

第 75 次发布

发布时间：2024-09-27 01:05:32

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [IMCdrInfo](#)
 - 新增成员：ClientUserId

第 74 次发布

发布时间：2024-09-24 13:17:26

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateAiCall](#)

第 73 次发布

发布时间：2024-08-22 01:10:57

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：AsrStatus

第 72 次发布

发布时间：2024-08-20 01:11:11

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeChatMessages](#)
 - 修改入参：SdkAppId

第 71 次发布

发布时间：2024-08-16 01:11:55

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyStaff](#)
 - 新增入参：StaffNo

第 70 次发布

发布时间：2024-07-23 01:10:10

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SkillGroupInfoItem](#)
 - 新增成员：Alias

第 69 次发布

发布时间：2024-06-06 01:17:16

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateIVRSession](#)

第 68 次发布

发布时间：2024-05-20 01:15:04

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [CompanyApplyInfo](#)
 - 新增成员：IsEqualTencentCloud, CorporationMobile, CorporationMobilePicUrl, UseDescribeFileUrl, CompanyAuthLetterPicUrl, OperatorMobile, OperatorEmail, OperatorMobilePicUrl
 - 删除成员：BusinessScope

- [CompanyStateInfo](#)
 - 新增成员：ModifyTime

第 67 次发布

发布时间：2024-04-24 01:06:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeChatMessages](#)
 - 废弃入参：CdrId

第 66 次发布

发布时间：2024-04-23 01:09:39

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [ModifyStaffPassword](#)

第 65 次发布

发布时间：2024-04-10 01:09:30

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [UploadIvrAudio](#)
 - 新增出参：SuccessFileList

第 64 次发布

发布时间：2024-04-08 01:09:16

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeIvrAudioList](#)
- [UploadIvrAudio](#)

新增数据结构：

- [AudioFileInfo](#)
- [UploadAudioInfo](#)
- [UploadIvrAudioFailedInfo](#)

第 63 次发布

发布时间：2024-03-28 01:09:50

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCompanyApply](#)
- [DescribeCompanyList](#)
- [ModifyCompanyApply](#)

新增数据结构：

- [CompanyApplyInfo](#)
- [CompanyStateInfo](#)

第 62 次发布

发布时间：2024-03-13 01:09:37

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SdkAppldBuyInfo](#)
 - 修改成员： SipBuyNum
- [ServeParticipant](#)
 - 修改成员： TransferFromType
- [SkillGroupInfoItem](#)
 - 修改成员： SkillGroupId, SkillGroupName, Type, RoutePolicy, UsingLastSeat, MaxConcurrency, LastModifyTimestamp, SkillGroupType
- [StaffBuyInfo](#)
 - 修改成员： SipNum
- [StaffStatusMetrics](#)
 - 修改成员： LastOnlineTimestamp, LastStatusTimestamp

第 61 次发布

发布时间：2024-03-07 01:09:17

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [StaffInfo](#)
 - 新增成员： RoleId

第 60 次发布

发布时间：2024-03-05 01:09:20

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [UpdateCCCSkillGroup](#)

第 59 次发布

发布时间：2024-03-01 01:09:52

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [AbortPredictiveDialingCampaign](#)
- [CreatePredictiveDialingCampaign](#)
- [DeletePredictiveDialingCampaign](#)

- [DescribePredictiveDialingCampaign](#)
- [DescribePredictiveDialingCampaigns](#)
- [DescribePredictiveDialingSessions](#)
- [PausePredictiveDialingCampaign](#)
- [ResumePredictiveDialingCampaign](#)
- [UpdatePredictiveDialingCampaign](#)

新增数据结构：

- [DescribePredictiveDialingCampaignsElement](#)

第 58 次发布

发布时间：2023-11-03 01:08:07

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SeatUserInfo](#)
 - 新增成员：Role
- [TelCdrInfo](#)
 - 新增成员：RecordId

第 57 次发布

发布时间：2023-10-13 01:09:06

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeIMCdrList](#)

修改接口：

- [DescribeChatMessages](#)
 - 废弃入参：Instanceld
- [DescribeIMCdrs](#)
 - 废弃入参：Instanceld
 - 新增出参：IMCdrList
 - 废弃出参：IMCdrs

第 56 次发布

发布时间：2023-10-12 01:16:25

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateCallOutSession](#)
 - 新增入参：UUI
 - 废弃入参：Uui
- [DescribeProtectedTelCdr](#)
 - 新增出参：TelCdrList
 - 废弃出参：TelCdrs
- [DescribeTelCallInfo](#)
 - 新增出参：VOIPCallInCount
 - 废弃出参：VoipCallInCount

- [DescribeTelCdr](#)
 - 废弃入参：Instanceld
 - 新增出参：TelCdrList
 - 废弃出参：TelCdrs

修改数据结构：

- [NumberInfo](#)
 - 新增成员：State
 - 修改成员：Number, CallOutSkillGroupIds
- [TelCdrInfo](#)
 - 新增成员：UUI
 - 废弃成员：Uui

第 55 次发布

发布时间：2023-09-07 01:15:54

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [BindStaffSkillGroupList](#)
 - 新增入参：StaffSkillGroupList
 - 修改入参：SkillGroupList
 - 废弃入参：SkillGroupList

新增数据结构：

- [StaffSkillGroupList](#)

第 54 次发布

发布时间：2023-08-23 01:10:26

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateSDKLoginToken](#)
 - 新增入参：OnlyOnce

第 53 次发布

发布时间：2023-08-08 01:09:59

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateAdminURL](#)

第 52 次发布

发布时间：2023-07-28 01:10:59

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SdkAppldBuyInfo](#)
 - 新增成员：SipBuyNum
- [StaffBuyInfo](#)
 - 新增成员：SipNum

第 51 次发布

发布时间：2023-06-30 01:11:07

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateExtension](#)
 - 新增入参：SkillGroupIds, Relation

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：VoicemailAsrURL
 - 修改成员：Caller, Callee, Time, Direction, Duration, RecordURL, SeatUser, EndStatus, SkillGroup, CallerLocation, IVRDuration, RingTimestamp, AcceptTimestamp, EndedTimestamp, IVRKeyPressed, HungUpSide, ServeParticipants, SkillGroupId, EndStatusString, StartTimestamp, QueuedTimestamp, PostIVRKeyPressed, QueuedSkillGroupId, SessionId, ProtectedCaller, ProtectedCallee, Uui, IVRKeyPressedEx, AsrUrl, CustomRecordURL, Remark, QueuedSkillGroupName, VoicemailRecordURL

第 50 次发布

发布时间：2023-06-08 01:08:30

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateCallOutSession](#)
 - 新增入参：Callers

第 49 次发布

发布时间：2023-05-17 01:08:49

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAutoCalloutTask](#)
 - 新增入参：UUI, CalleeAttributes

新增数据结构：

- [CalleeAttribute](#)

第 48 次发布

发布时间：2023-05-08 01:11:11

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [ServeParticipant](#)

- 新增成员：TransferFromType

第 47 次发布

发布时间：2023-04-24 01:06:05

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeCallInMetrics](#)
 - 新增入参：GroupIdList
- [DescribeStaffStatusMetrics](#)
 - 新增入参：GroupIdList, StatusList

第 46 次发布

发布时间：2023-03-28 01:09:13

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeTelCallInfo](#)
 - 新增出参：AsrOfflineCount, AsrRealtimeCount

第 45 次发布

发布时间：2023-03-24 01:13:32

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SeatUserInfo](#)
 - 修改成员：Name, Mail, StaffNumber

第 44 次发布

发布时间：2023-03-23 01:14:04

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [StaffStatusMetrics](#)
 - 新增成员：LastOnlineTimestamp, LastStatusTimestamp

第 43 次发布

发布时间：2023-03-08 01:06:07

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DisableCCCPhoneNumber](#)
 - 修改入参：SdkAppId

第 42 次发布

发布时间：2023-03-07 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：

- [DescribeTelCallInfo](#)
 - 新增出参：VoipCallInCount

第 41 次发布

发布时间：2023-02-09 01:09:05

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [BindNumberCallOutSkillGroup](#)
- [DescribeNumbers](#)
- [UnbindNumberCallOutSkillGroup](#)

新增数据结构：

- [NumberInfo](#)

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：VoicemailRecordURL

第 40 次发布

发布时间：2023-01-09 01:07:34

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：QueuedSkillGroupName

第 39 次发布

发布时间：2022-12-16 01:10:35

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [HangUpCall](#)

第 38 次发布

发布时间：2022-12-08 01:17:22

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：Remark
 - 修改成员：Caller, Callee, Time, Direction, Duration, RecordURL, SeatUser, EndStatus, SkillGroup, CallerLocation, IVRDuration, RingTimestamp, AcceptTimestamp, EndedTimestamp, IVRKeyPressed, HungUpSide, ServeParticipants, SkillGroupId, EndStatusString, StartTimestamp, QueuedTimestamp, PostIVRKeyPressed, QueuedSkillGroupId, SessionId, ProtectedCaller, ProtectedCallee, Uui, IVRKeyPressedEx, AsrUrl, CustomRecordURL

第 37 次发布

发布时间：2022-12-07 06:21:07

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SkillGroupInfoItem](#)
 - 新增成员：SkillGroupType
 - 修改成员：SkillGroupId, SkillGroupName, Type, RoutePolicy, UsingLastSeat, MaxConcurrency, LastModifyTimestamp

第 36 次发布

发布时间：2022-11-21 06:12:39

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCarrierPrivilegeNumberApplicant](#)
- [DescribeActiveCarrierPrivilegeNumber](#)
- [DescribeCarrierPrivilegeNumberApplicants](#)

新增数据结构：

- [ActiveCarrierPrivilegeNumber](#)
- [CarrierPrivilegeNumberApplicant](#)
- [Filter](#)

第 35 次发布

发布时间：2022-11-16 06:11:21

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateStaff](#)
 - 新增入参：SendPassword

第 34 次发布

发布时间：2022-11-08 06:12:05

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- [IMSatisfaction](#)

修改数据结构：

- [IMCdrInfo](#)
 - 新增成员：Satisfaction

第 33 次发布

发布时间：2022-10-31 06:13:20

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateExtension](#)
- [DeleteExtension](#)
- [DescribeExtension](#)
- [DescribeExtensions](#)
- [ModifyExtension](#)
- [ResetExtensionPassword](#)

新增数据结构：

- [ExtensionInfo](#)

第 32 次发布

发布时间：2022-10-26 06:15:12

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateAutoCalloutTask](#)
 - 新增入参：Variables

新增数据结构：

- [Variable](#)

第 31 次发布

发布时间：2022-09-22 06:12:48

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DisableCCCPhoneNumber](#)

第 30 次发布

发布时间：2022-09-06 06:13:56

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [AutoCalloutTaskCalleeInfo](#)
 - 新增成员：Sessions

第 29 次发布

发布时间：2022-08-17 06:11:23

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeStaffInfoList](#)
 - 新增入参：SkillGroupId

第 28 次发布

发布时间：2022-08-11 06:14:26

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DescribeSeatUserList

修改接口：

- [DescribeSkillGroupInfoList](#)
 - 新增入参：SkillGroupName
- [ModifyStaff](#)
 - 新增入参：UseMobileCallOut, UseMobileAccept

修改数据结构：

- [StaffStatusMetrics](#)
 - 新增成员：UseMobileAccept, UseMobileCallOut

第 27 次发布

发布时间：2022-07-01 06:06:59

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [ModifyStaff](#)

第 26 次发布

发布时间：2022-06-20 06:03:42

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCCCSkillGroup](#)

第 25 次发布

发布时间：2022-05-24 06:07:44

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [AutoCalloutTaskInfo](#)
 - 新增成员：TaskId

第 24 次发布

发布时间：2022-04-15 06:03:13

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeProtectedTelCdr](#)

第 23 次发布

发布时间：2022-04-13 06:12:26

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeTelCdr](#)
 - 修改入参：Limit, Offset

第 22 次发布

发布时间：2022-03-04 08:05:02

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [ServeParticipant](#)
 - 新增成员：CustomRecordURL
- [TelCdrInfo](#)
 - 新增成员：CustomRecordURL

第 21 次发布

发布时间：2022-02-14 08:06:57

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateAutoCalloutTask](#)
- [DescribeAutoCalloutTask](#)
- [StopAutoCalloutTask](#)

新增数据结构：

- [AutoCalloutTaskCalleeInfo](#)

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：AsrUrl

第 20 次发布

发布时间：2022-01-24 08:05:51

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateCallOutSession](#)
- [DescribeAutoCalloutTasks](#)

新增数据结构：

- [AutoCalloutTaskInfo](#)

修改数据结构：

- [IMCdrInfo](#)
 - 新增成员：SessionId, SkillGroupId, SkillGroupName

第 19 次发布

发布时间：2021-12-10 08:02:29

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：IVRKeyPressedEx

第 18 次发布

发布时间：2021-12-06 08:02:18

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeCCCBuyInfoList](#)

新增数据结构：

- [PackageBuyInfo](#)
- [PhoneNumBuyInfo](#)
- [SdkAppIdBuyInfo](#)
- [StaffBuyInfo](#)

第 17 次发布

发布时间：2021-11-16 08:06:45

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：Uui

第 16 次发布

发布时间：2021-11-01 08:06:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeChatMessages](#)

- 新增入参：SessionId
- 修改入参：CdrId
- [DescribeTelCdr](#)
 - 新增入参：SessionIds

修改数据结构：

- [PSTNSession](#)
 - 新增成员：ProtectedCaller, ProtectedCallee
- [PSTNSessionInfo](#)
 - 新增成员：ProtectedCaller, ProtectedCallee
- [TelCdrInfo](#)
 - 新增成员：ProtectedCaller, ProtectedCallee

第 15 次发布

发布时间：2021-08-10 08:01:44

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeCallInMetrics](#)
- [DescribeStaffStatusMetrics](#)

新增数据结构：

- [CallInMetrics](#)
- [CallInNumberMetrics](#)
- [CallInSkillGroupMetrics](#)
- [StaffStatusExtra](#)
- [StaffStatusMetrics](#)

第 14 次发布

发布时间：2021-07-08 08:01:24

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateUserSig](#)

第 13 次发布

发布时间：2021-07-07 09:55:44

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeTelCdr](#)
 - 新增入参：Phones

第 12 次发布

发布时间：2021-07-06 08:01:37

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [BindStaffSkillGroupList](#)
- [DeleteStaff](#)
- [UnbindStaffSkillGroupList](#)

修改接口：

- [CreateStaff](#)
 - 新增出参：ErrorStaffList

新增数据结构：

- [ErrStaffitem](#)

第 11 次发布

发布时间：2021-04-07 08:00:41

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeTelSession](#)

新增数据结构：

- [PSTNSession](#)

第 10 次发布

发布时间：2021-03-16 08:00:43

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：SessionId

第 9 次发布

发布时间：2021-03-02 08:00:36

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeSkillGroupInfoList](#)
- [DescribeStaffInfoList](#)

新增数据结构：

- [SkillGroupInfoItem](#)
- [SkillGroupItem](#)
- [StaffInfo](#)

第 8 次发布

发布时间：2021-01-28 08:00:34

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeSeatUserList](#)

修改接口：

- [DescribeTelCdr](#)
 - 新增入参：PageSize, PageNumber

新增数据结构：

- [IVRKeyPressedElement](#)
- [ServeParticipant](#)

修改数据结构：

- [TelCdrInfo](#)
 - 新增成员：IVRDuration, RingTimestamp, AcceptTimestamp, EndedTimestamp, IVRKeyPressed, HungUpSide, ServeParticipants, SkillGroupId, EndStatusString, StartTimestamp, QueuedTimestamp, PostIVRKeyPressed, QueuedSkillGroupId

第 7 次发布

发布时间：2020-12-17 08:00:22

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribePSTNActiveSessionList](#)

新增数据结构：

- [PSTNSessionInfo](#)

第 6 次发布

发布时间：2020-11-23 08:00:28

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeTelCallInfo](#)

第 5 次发布

发布时间：2020-11-16 08:00:24

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateSDKLoginToken](#)
 - 新增出参：SdkURL

第 4 次发布

发布时间：2020-11-05 08:00:21

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateStaff](#)
- [DescribeChatMessages](#)
- [DescribeIMCdrs](#)

修改接口：

- [DescribeTelCdr](#)
 - 新增入参：SdkAppId
 - 修改入参：InstanceId

新增数据结构：

- [IMCdrInfo](#)
- [Message](#)
- [MessageBody](#)

第 3 次发布

发布时间：2020-10-23 08:00:22

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [SeatUserInfo](#)
 - 新增成员：StaffNumber
 - 修改成员：Phone

第 2 次发布

发布时间：2020-10-20 08:00:12

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateSDKLoginToken](#)

第 1 次发布

发布时间：2020-08-21 17:50:27

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeTelCdr](#)

新增数据结构：

- [SeatUserInfo](#)
- [TelCdrInfo](#)

简介

最近更新时间：2025-04-25 01:12:34

腾讯云联络中心（Tencent Cloud Contact Center，CCC）是深度融合电话、即时通信、音视频等多种通信能力为一体的融合通信平台，支持电话、微信公众号、小程序等渠道接入。您可以使用融合通信工作台统一接待用户呼入，轻松设置全渠道服务流程，轻松实现各类场景需求。

API 概览

最近更新时间：2025-07-04 01:12:42

.....

座席相关接口

接口名称	接口功能	频率限制（次/秒）
CreateSDKLoginToken	创建 SDK 登录 Token	20
CreateStaff	创建客服账号	20
DeleteStaff	删除坐席信息	20
ModifyStaff	修改客服账号	20
DescribeStaffInfoList	获取坐席信息列表	20
DescribeStaffStatusMetrics	获取坐席实时状态统计指标	20
ModifyStaffPassword	修改座席密码	20
ForceMemberOffline	强制客服下线	20
RestoreMemberOnline	恢复客服上线	20

技能组相关接口

接口名称	接口功能	频率限制（次/秒）
CreateCCCSkillGroup	新建技能组	20
DeleteCCCSkillGroup	删除客服技能组	20
UpdateCCCSkillGroup	更新技能组	20
DescribeSkillGroupInfoList	获取技能组信息列表	20
BindStaffSkillGroupList	绑定座席所属技能组	20
UnbindStaffSkillGroupList	解绑坐席所属技能组	20

号码相关接口

接口名称	接口功能	频率限制（次/秒）
BindNumberCallInInterface	绑定号码呼入回调接口	20
BindNumberCallOutSkillGroup	绑定号码外呼技能组	20
UnbindNumberCallOutSkillGroup	解绑号码外呼技能组	20
DescribeNumbers	查询号码列表	20
DisableCCCPhoneNumber	停用号码	20
CreateCarrierPrivilegeNumberApplicant	申请运营商白名单号码	20
DescribeCarrierPrivilegeNumberApplicants	查询运营商白名单号码申请	20
DescribeActiveCarrierPrivilegeNumber	查询生效运营商白名单规则	20
CreateOwnNumberApply	创建客户自携号码接入审核	20
ModifyOwnNumberApply	修改客户自携号码审批单	20

双呼相关接口

接口名称	接口功能	频率限制（次/秒）
CreateCallOutSession	创建外呼会话（当前仅支持双呼）	20

自动外呼相关接口

接口名称	接口功能	频率限制（次/秒）
CreateAutoCalloutTask	创建自动外呼任务	20
DescribeAutoCalloutTasks	批量查询自动外呼任务	20
StopAutoCalloutTask	停止自动外呼任务	20
DescribeAutoCalloutTask	查询自动外呼任务详情	20
AbortPredictiveDialingCampaign	停止预测式外呼任务	20
CreatePredictiveDialingCampaign	创建预测式外呼任务	20
DeletePredictiveDialingCampaign	删除预测式外呼任务	20
DescribeAgentCruiseDialingCampaign	查询座席巡航式外呼任务	20
DescribePredictiveDialingCampaign	查询预测式外呼任务	20
DescribePredictiveDialingCampaigns	查询预测式外呼任务列表	20
DescribePredictiveDialingSessions	查询预测式外呼呼叫列表	20
PausePredictiveDialingCampaign	暂停预测式外呼任务	20
ResumePredictiveDialingCampaign	恢复预测式外呼任务	20
UpdatePredictiveDialingCampaign	更新预测式外呼任务	20

话机相关接口

接口名称	接口功能	频率限制（次/秒）
CreateExtension	创建话机账号	20
DeleteExtension	删除话机账号	20
ModifyExtension	修改话机账号(绑定技能组、绑定坐席账号)	20
DescribeExtension	获取话机信息	20
DescribeExtensions	查询话机列表信息	20
ResetExtensionPassword	重置话机注册密码	20

电话服务记录相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeTelCdr	获取电话服务记录与录音	20
DescribeProtectedTelCdr	获取主被叫受保护的电话服务记录与录音	20
DescribePSTNActiveSessionList	获取 PSTN 活动会话列表	20
DescribeCallInMetrics	获取呼入实时数据统计指标	20
DescribeTelSession	获取 PSTN 会话信息	20
DescribeTelCallInfo	按实例获取电话消耗统计	20

在线服务记录相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeIMCdrList	查询在线客服记录	20

接口名称	接口功能	频率限制（次/秒）
DescribeIMCdrs	查询在线客服记录(已废弃)	20
DescribeChatMessages	查询指定会话（SessionId）的聊天记录	20

购买相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeCCCBuyInfoList	获取用户购买信息列表	20

其他接口

接口名称	接口功能	频率限制（次/秒）
CreateUserSig	创建用户数据签名	20
HangUpCall	挂断电话	20
DescribeTelRecordAsr	拉取会话录音转文本信息	20
CreateAdminURL	创建管理端访问链接	20
DescribeIvrAudioList	查询IVR音频文件列表	20
UploadIvrAudio	上传IVR音频文件	1
TransferToManual	转接人工	20

外呼相关接口

接口名称	接口功能	频率限制（次/秒）
AbortAgentCruiseDialingCampaign	停止座席巡航式外呼任务	20
CreateAgentCruiseDialingCampaign	创建座席巡航式外呼任务	20
CreateIVRSession	CreateIVRSession	20

企业资质相关接口

接口名称	接口功能	频率限制（次/秒）
CreateCompanyApply	创建企业资质申请	1
DescribeCompanyList	查询企业资质申请列表	20
ModifyCompanyApply	修改企业资质申请	20

语音智能体相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeAILatency	获取 AI 耗时	20
CreateAIAgentCall	创建智能体通话	20
CreateAICall	创建AI通话	20
DescribeAICallExtractResult	查询 AI 通话内容提取结果	20

AI服务相关接口

接口名称	接口功能	频率限制（次/秒）
ControlAIConversation	控制AI对话	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 01:10:15

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 `ccc.tencentcloudapi.com`，也支持指定地域域名访问，例如广州地域的域名为 `ccc.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 `ccc.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）	<code>ccc.tencentcloudapi.com</code>
华南地区（广州）	<code>ccc.ap-guangzhou.tencentcloudapi.com</code>
华东地区（上海）	<code>ccc.ap-shanghai.tencentcloudapi.com</code>
华东地区（南京）	<code>ccc.ap-nanjing.tencentcloudapi.com</code>
华北地区（北京）	<code>ccc.ap-beijing.tencentcloudapi.com</code>
西南地区（成都）	<code>ccc.ap-chengdu.tencentcloudapi.com</code>
西南地区（重庆）	<code>ccc.ap-chongqing.tencentcloudapi.com</code>
港澳台地区（中国香港）	<code>ccc.ap-hongkong.tencentcloudapi.com</code>
亚太东南（新加坡）	<code>ccc.ap-singapore.tencentcloudapi.com</code>
亚太东南（雅加达）	<code>ccc.ap-jakarta.tencentcloudapi.com</code>
亚太东南（曼谷）	<code>ccc.ap-bangkok.tencentcloudapi.com</code>
亚太东北（首尔）	<code>ccc.ap-seoul.tencentcloudapi.com</code>
亚太东北（东京）	<code>ccc.ap-tokyo.tencentcloudapi.com</code>
美国东部（弗吉尼亚）	<code>ccc.na-ashburn.tencentcloudapi.com</code>
美国西部（硅谷）	<code>ccc.na-siliconvalley.tencentcloudapi.com</code>
南美地区（圣保罗）	<code>ccc.sa-saopaulo.tencentcloudapi.com</code>
欧洲地区（法兰克福）	<code>ccc.eu-frankfurt.tencentcloudapi.com</code>

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）	<code>ccc.ap-shanghai-fsi.tencentcloudapi.com</code>
华南地区（深圳金融）	<code>ccc.ap-shenzhen-fsi.tencentcloudapi.com</code>

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 01:14:19

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输入公共参数 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 ccc；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, SignedHeaders=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, SignedHeaders=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。
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 。

参数名称	类型	必选	描述
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=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&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=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&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
亚太东南（新加坡）	ap-singapore

签名方法 v3

最近更新時間：2024-12-25 01:14:44

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 *****。用户想查看广州区云服务器名为“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： 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

字段名称	解释
	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，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。
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";
$hashedCanonicalRequest = hash("SHA256", $canonicalRequest);
$stringToSign = $algorithm."\n"
.$timestamp."\n"
.$credentialScope."\n"
.$hashedCanonicalRequest;
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}\nxc-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}, Signature=#{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, 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++)
{
    snprintf(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++)
    {
        sprintf(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 OPENSSSL_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 OPENSSSL_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* 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, "\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", "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 01:14:44

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
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-gu
angzhou&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&R
egion=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 编码前需要先以 UTF-8 进行编码。

注意：有些编程语言的网络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=7RAM2xfNMO9EiVTNmPg06MRnCvQ%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("evm.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");
$params["Nonce"] = 11886;//rand();
$params["Timestamp"] = 1465185768;//time();
$params["Region"] = "ap-guangzhou";
$params["SecretId"] = $secretId;
$params["Version"] = "2017-03-12";
$params["Action"] = "DescribeInstances";
$params["InstanceIds.0"] = "ins-09dx96dg";
$params["Limit"] = 20;
$params["Offset"] = 0;

ksort($params);

$signStr = "GETcvm.tencentcloudapi.com/?";
foreach ( $params 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
// $params["Signature"] = $signature;
// $url = "https://cvm.tencentcloudapi.com/?".http_build_query($params);
// 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 sigInParam = MakeSignPlainText(headers, "GET", endpoint, "/");
    string sigOutParam = Sign(SECRET_KEY, sigInParam);
    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 01:13:00

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：2022-08-10 06:17:46

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。

座席相关接口

创建 SDK 登录 Token

最近更新时间：2025-04-25 01:12:30

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

创建 SDK 登录 Token。

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：CreateSDKLoginToken。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SeatUserId	是	String	座席账号。 示例值：FooOrBar@tencent.com
OnlyOnce	否	Boolean	生成的token是否一次性校验 示例值：false

3. 输出参数

参数名称	类型	描述
Token	String	SDK 登录 Token。 示例值：some_token
ExpiredTime	Integer	过期时间戳，Unix 时间戳。 示例值：1601371974
SdkURL	String	SDK 加载路径会随着 SDK 的发布而变动。 示例值： https://tccc/sdk.js
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 SDK 登录 Token

创建 SDK 登录 Token

输入示例

```
https://ccc.tencentcloudapi.com/?Action=CreateSDKLoginToken
&SdkAppId=1400000000
```

```
&SeatUserId=FooOrBar@tencent.com
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "Token": "6bb56a09278740bc80c5dc6dab783eff",
    "SdkURL": "https://29294-22989-29805-29810.cdn-go.cn/tccc-agent-sdk/latest/",
    "ExpiredTime": 1601371974
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.AccountNotExist	账号不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。
OperationDenied.UinDisabled	账号已被禁用。

创建客服账号

最近更新时间：2025-04-25 01:12:30

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建客服账号。

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：CreateStaff。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Staffs.N	是	Array of SeatUserInfo	客服信息，个数不超过 10 示例值：[{"Name":"John Doe","Mail":"john.doe@example.com","Phone":"00861234567890","StaffNumber":"123456","Role":{"Name":"Jane Smith","Mail":"jane.smith@example.com","Phone":"00861234567890","StaffNumber":"789012","R
SendPassword	否	Boolean	是否发送密码邮件，默认true 示例值：false

3. 输出参数

参数名称	类型	描述
ErrorStaffList	Array of ErrStaffItem	错误坐席列表及错误信息 示例值：[{"StaffEmail": "foobar@example.com"}]
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 创建客服示例

创建客服示例

输入示例

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

```
{
  "Staffs": [
    {
      "Phone": "联系电话",
      "Mail": "联系人邮箱",
      "StaffNumber": "001",
      "Name": "小军"
    }
  ],
  "SdkAppId": 1400000000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "ErrorStaffList": []
  }
}
```

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.DuplicatedAccount	重复账号。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。

错误码	描述
InvalidParameterValue.InstanceNotExist	实例不存在。
LimitExceeded	超过配额限制。

删除坐席信息

最近更新时间：2025-04-25 01:12:30

-

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

删除坐席信息

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：DeleteStaff。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
StaffList.N	是	Array of String	待删除客服邮箱列表，一次最大支持200个。 示例值：["staff1@xxx.com","staff2@xxx.com"]

3. 输出参数

参数名称	类型	描述
OnlineStaffList	Array of String	无法删除的状态为在线的客服列表 示例值：["staff1@xxx.com","staff2@xxx.com"]
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 删除坐席信息示例

输入示例

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

{
  "SdkAppId": 1400000000,
  "StaffList": [
    "staff1@xxx.com",
    "staff2@xxx.com"
  ]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "48edd236-7ef1-45af-9e12-fc376ba355bf",
    "OnlineStaffList": []
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue.InstanceNotExist	实例不存在。

修改客服账号

最近更新时间：2025-07-02 01:16:21

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

修改客服账号

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：ModifyStaff。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000
Email	是	String	座席账户 示例值：xx@tencent.com
Name	否	String	座席名称 示例值：小王
Phone	否	String	座席手机号（带0086前缀,示例：008618011111111） 示例值：008618088888888
Nick	否	String	座席昵称 示例值：小管家
StaffNo	否	String	座席工号 示例值：114514
SkillGroupIds.N	否	Array of Integer	绑定技能组ID列表 示例值：[123,134]
UseMobileCallOut	否	Boolean	是否开启手机外呼开关 示例值：true
UseMobileAccept	否	Integer	手机接听模式 0 - 关闭 1 - 仅离线 2 - 始终 示例值：0
ExtensionNumber	否	String	座席分机号（1 到 8 打头，4 - 6 位） 示例值：1003
ForwardingConfig	否	ForwardingConfig	呼叫转移配置 示例值：{"Enabled":true,"Condition":2,"Target":{"Type":1,"StaffUserId":"foo@tencent.com"}}

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 修改客服示例

修改客服示例

输入示例

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

{
  "Phone": "008618092688910",
  "Email": "xxx@tencent.com",
  "SdkAppId": 1400000000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.AccountNotExist	账号不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。
InvalidParameterValue.PhoneNumIsBoundOtherAccount	号码已经绑定别的账号。
InvalidParameterValue.SkillGroupError	技能组错误。
OperationDenied.NotInWhiteList	不在白名单中。

获取坐席信息列表

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取坐席信息列表

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：DescribeStaffInfoList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	是	Integer	分页尺寸，上限 9999 示例值：9999
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0
StaffMail	否	String	坐席账号，查询单个坐席时使用 示例值：102992@qq.com
ModifiedTime	否	Integer	查询修改时间大于等于ModifiedTime的坐席时使用 示例值：1611631881
SkillGroupId	否	Integer	技能组ID 示例值：1145

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	坐席用户总数 示例值：10
StaffList	Array of StaffInfo	坐席用户信息列表 示例值：[{ "Name": "n", "Mail": "m@e.com", "Phone": "123", "Nick": "t", "StaffNumber": "1", "RoleId": "1", "SkillGroupList": [{"SkillGroupId": "1", "SkillGroupName": "sg", "Priority": "1", "Type": "IM"}], "LastModifyTimestamp": "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 获取坐席列表示例

获取坐席列表示例

输入示例

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

{
  "ModifiedTime": 1590147606,
  "PageSize": 10,
  "PageNumber": 0,
  "StaffMail": "“121223@qq.com”",
  "SdkAppId": "1400000000"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9896bd",
    "TotalCount": 602,
    "StaffList": [
      {
        "Name": "xiao",
        "Mail": "1000273@qq.com",
        "Phone": "008617636049517",
        "Nick": "测试3",
        "StaffNumber": "125",
        "RoleId": 1,
        "SkillGroupList": [
          {
            "SkillGroupId": 53,
            "SkillGroupName": "ALL-dingoding-测试",
            "Priority": 3,
            "Type": "ALL"
          },
          {
            "SkillGroupId": 82,
            "SkillGroupName": "uu",
            "Priority": 1,
            "Type": "IM"
          }
        ],
        "LastModifyTimestamp": 1613988825
      }
    ]
  }
}
```

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.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

获取坐席实时状态统计指标

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取坐席实时状态统计指标

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：DescribeStaffStatusMetrics。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
StaffList.N	否	Array of String	筛选坐席列表，默认不传返回全部坐席信息 示例值：[a@tencent.com,b@tencent.com,c@tencent.com]
GroupIdList.N	否	Array of Integer	筛选技能组ID列表 示例值：[1,2,3,4,5,6]
StatusList.N	否	Array of String	筛选坐席状态列表 座席状态 free 示闲 busy 忙碌 rest 小休 notReady 示忙 afterCallWork 话后调整 offline 离线 示例值：["free"]

3. 输出参数

参数名称	类型	描述
Metrics	Array of StaffStatusMetrics	坐席状态实时信息
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 获取坐席实时状态统计指标示例

获取坐席实时状态统计指标示例。

输入示例

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

```
{
  "SdkAppId": 0,
  "StaffList": [
    "aaa@abc.com"
  ],
  "GroupIdList": [
    0
  ],
  "StatusList": [
    "free"
  ]
}
```

输出示例

```
{
  "Response": {
    "Metrics": [
      {
        "Email": "aaa@abc.com",
        "Status": "free",
        "StatusExtra": {
          "Type": "tel",
          "Direct": "callin"
        },
        "OnlineDuration": 0,
        "FreeDuration": 0,
        "BusyDuration": 0,
        "NotReadyDuration": 0,
        "RestDuration": 0,
        "AfterCallWorkDuration": 0,
        "Reason": "会议",
        "ReserveRest": true,
        "ReserveNotReady": true,
        "UseMobileAccept": 0,
        "UseMobileCallOut": true,
        "LastOnlineTimestamp": 0,
        "LastStatusTimestamp": 0
      }
    ],
    "RequestId": "abc"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

修改座席密码

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

修改座席的密码

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：ModifyStaffPassword。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：160000000
Email	是	String	座席邮箱 示例值：a@qq.com
Password	是	String	设置的密码 示例值：abcdefg

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 修改密码

输入示例

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

{
  "SdkAppId": 1600000000,
  "Email": "a@qq.com",
  "Password": "foobar"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "abc"
  }
}
```

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.AccountNotExist	账号不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

恢复客服上线

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

恢复客服上线

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：RestoreMemberOnline。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
UserId	是	String	客服ID 示例值：user@example.com

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 恢复客服上线

输入示例

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

{
  "SdkAppId": 1400000000,
  "UserId": "user@example.com"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9896cd"
  }
}
```

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。
InternalServerError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

强制客服下线

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

强制客服下线

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：ForceMemberOffline。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
UserId	是	String	客服ID 示例值：user@example.com

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 强制客服下线

输入示例

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

{
  "SdkAppId": 1400000000,
  "UserId": "user@example.com"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9896bd"
  }
}
```

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。
InternalServerError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

技能组相关接口

新建技能组

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

新建技能组

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：CreateCCCSkillGroup。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SkillGroupName	是	String	技能组名称 示例值：foobar技能组
SkillGroupType	是	Integer	技能组类型0-电话，1-在线，3-音频，4-视频 示例值：0
MaxConcurrency	否	Integer	技能组接待人数上限（该技能组中1个座席可接待的人数上限）默认为1。1、若技能组类型为在线，则接待上限可设置为1及以上 2、若技能组类型为电话、音频、视频，则接待上线必须只能为1 示例值：1

3. 输出参数

参数名称	类型	描述
SkillGroupId	Integer	技能组ID 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 创建技能组实例

输入示例

```
https://ccc.tencentcloudapi.com/?Action=CreateCCCSkillGroup
&SdkAppId=1400000000
&SkillGroupName="test"
&SkillGroupType=0
```

```
&MaxConcurrency=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9896bd",
    "SkillGroupId": 602
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.SkillGroupExist	技能组已存在。

删除客服技能组

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

删除技能组

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：DeleteCCCSkillGroup。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SkillGroupId	是	Integer	技能组ID 示例值：58832

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 删除技能组

输入示例

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

{
  "SdkAppId": 1400000000,
  "SkillGroupId": 58832
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9812bd"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
ResourceUnavailable	资源不可用。

更新技能组

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

更新技能组

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：UpdateCCCSkillGroup。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SkillGroupId	是	Integer	技能组ID 示例值：12345678
SkillGroupName	否	String	修改后的技能组名字 示例值：客户一组
MaxConcurrency	否	Integer	修改后的最大并发数,同振最大为2 示例值：1
RingAll	否	Boolean	true同振，false顺振 示例值：true

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 更新技能组信息

输入示例

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

{
```

```
"SdkAppId": 1400000,
"SkillGroupId": 1234,
"SkillGroupName": "客户一组",
"MaxConcurrency": 1,
"RingAll": true
}
```

输出示例

```
{
  "Response": {
    "RequestId": "abc"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.SkillGroupExist	技能组已存在。

获取技能组信息列表

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取技能组信息列表

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：DescribeSkillGroupInfoList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	是	Integer	分页尺寸，上限 100 示例值：100
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0
SkillGroupId	否	Integer	技能组ID，查询单个技能组时使用 示例值：12
ModifiedTime	否	Integer	查询修改时间大于等于ModifiedTime的技能组时使用 示例值：1611631881
SkillGroupName	否	String	技能组名称 示例值：销售组

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	技能组总数 示例值：10
SkillGroupList	Array of SkillGroupInfoItem	技能组信息列表 示例值： [{"SkillGroupId":1115,"SkillGroupName":"lulutt","Type":"TEL","RoutePolicy":"firstCreate","UsingLastS
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）

4. 示例

示例1 获取技能组列表示例

获取技能组列表示例

输入示例

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

```
{
  "PageSize": 10,
  "PageNumber": 0,
  "ModifiedTime": 1590147606,
  "SdkAppId": 1400000000,
  "SkillGroupId": 12
}
```

输出示例

```
{
  "Response": {
    "RequestId": "5ac74e13-ef15-41a6-9639-f1bc8c9896bd",
    "TotalCount": 2,
    "SkillGroupList": [
      {
        "SkillGroupId": 1115,
        "SkillGroupName": "luluttt",
        "SkillGroupType": 1,
        "Type": "TEL",
        "RoutePolicy": "firstCreate",
        "UsingLastSeat": 0,
        "MaxConcurrency": 1,
        "LastModifyTimestamp": 1613976392,
        "Alias": "6666"
      }
    ]
  }
}
```

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
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

绑定座席所属技能组

最近更新时间：2025-04-25 01:12:29

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

绑定座席所属技能组

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：BindStaffSkillGroupList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
StaffEmail	是	String	座席邮箱 示例值：staff1@xx.com
StaffSkillGroupList.N	否	Array of StaffSkillGroupList	绑定技能组列表(必填) 示例值：[{"SkillGroupId":100,"Priority":1},{"SkillGroupId":200,"Priority":2}]

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 绑定坐席所属技能组示例

输入示例

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

{
  "SdkAppId": 1400000000,
  "StaffEmail": "staff1@xxx.com",
  "StaffSkillGroupList": [
    {
      "SkillGroupId": 100,
```

```
"Priority": 1
}
]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "48edd236-7ef1-45af-9e12-fc376ba355bf"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

解绑坐席所属技能组

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

解绑坐席所属技能组

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：UnbindStaffSkillGroupList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
StaffEmail	是	String	客服邮箱 示例值：staff1@xx.com
SkillGroupList.N	是	Array of Integer	解绑技能组列表 示例值：[1,2,3,4,5,6]

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 解绑坐席所属技能组示例

输入示例

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

{
  "SdkAppId": 1400000000,
  "StaffEmail": "staff1@xxx.com",
  "SkillGroupList": [
    100,
    101
```



```
]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "48edd236-7ef1-45af-9e12-fc376ba355bf"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

号码相关接口

绑定号码外呼技能组

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

绑定号码外呼技能组

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：BindNumberCallOutSkillGroup。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Number	是	String	待绑定的号码 示例值："0086075512345678"
SkillGroupIds.N	是	Array of Integer	待绑定的技能组Id列表 示例值：[1,2,3]

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 绑定号码和技能组

绑定号码和技能组

输入示例

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

{
  "Number": "0086075512345678",
  "SkillGroupIds": [
```

```
1,  
2,  
3  
],  
"SdkAppId": 1400000000  
}
```

输出示例

```
{  
  "Response": {  
    "RequestId": "abc"  
  }  
}
```

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	内部数据库访问失败。

绑定号码呼入回调接口

最近更新时间：2025-06-11 11:58:41

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

绑定号码呼入回调接口

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：BindNumberCallInInterface。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Number	是	String	待绑定的号码 示例值："0086075512345678"
CallInInterface	否	Interface	待绑定的回调地址

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 绑定号码呼入回调接口

输入示例

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

{
  "SdkAppId": 1400000000,
  "Number": "0086075586013388",
  "CallInInterface": {
    "URL": "https://cloud.tencent.com/"
```

```
}  
}
```

输出示例

```
{  
  "Response": {  
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"  
  }  
}
```

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	内部数据库访问失败。

解绑号码外呼技能组

最近更新时间：2025-04-25 01:12:31

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

解绑号码外呼技能组

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：UnbindNumberCallOutSkillGroup。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Number	是	String	待解绑的号码 示例值："0086075512345678"
SkillGroupIds.N	是	Array of Integer	待解绑的技能组Id列表 示例值：[1,2,3]

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 解绑号码外呼技能组

解绑号码外呼技能组

输入示例

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

{
  "Number": "0086075512345678",
  "SkillGroupIds": [
    1,
    2,
```

```
3
],
"SdkAppId": 1400000000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "abc"
  }
}
```

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](#)。

查询号码列表

最近更新时间：2025-04-25 01:12:31

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询号码列表

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：DescribeNumbers。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageNumber	否	Integer	页数，从0开始 示例值：0
PageSize	否	Integer	分页大小，默认20 示例值：20

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	总数量 示例值：1
Numbers	Array of NumberInfo	号码列表 示例值：[{"Number":"0086075512345678","State":1,"CallOutSkillGroupIds":[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 查询号码列表

查询号码列表

输入示例

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



```
{
  "SdkAppId": 1400000000,
  "PageNumber": 0,
  "PageSize": 10
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "Numbers": [
      {
        "Number": "0086075512345678",
        "State": 1,
        "CallOutSkillGroupIds": [
          1
        ]
      }
    ],
    "RequestId": "abc"
  }
}
```

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](#)。

停用号码

最近更新时间：2025-04-25 01:12:31

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

停用号码

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：DisableCCCPhoneNumber。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
PhoneNumbers.N	是	Array of String	号码列表，0086开头 示例值：['00860755XXXX','0086021XXXX']
Disabled	是	Integer	停用开关，0启用 1停用 示例值：1
SdkAppId	否	Integer	TCCC 实例应用 ID 示例值：1400000000

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 停用号码示例

停用号码。

输入示例

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

{
  "SdkAppId": 1,
  "PhoneNumbers": [
    "008602112345678"
  ],
}
```

```
"Disabled": 0
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3651cda6-6501-4482-9f4e-8d0c9548a4db"
  }
}
```

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.CurStateNotAllowModify	当前号码状态不能被修改。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter.DuplicateAddress	重复的地址
InvalidParameter.DuplicatePhoneNumber	重复的号码
InvalidParameter.DuplicateSipAccount	重复的SIP账号
InvalidParameter.IllegalAddress	非法的地址
InvalidParameter.IllegalPhoneNumber	非法的号码
InvalidParameter.InvalidAddress	无效的地址
InvalidParameter.InvalidIP	无效的IP信息
InvalidParameter.InvalidPhoneNumber	无效的号码
InvalidParameter.InvalidPort	无效的端口信息

错误码	描述
InvalidParameter.SipAccountPasswordFormat	密码不合法(长度大于等于八位，必须包含大小写字母以及数字)
InvalidParameter.SipAccountUserFormat	用户名不合法(只能包含A-Z、a-z、以及数字)
InvalidParameter.SipTrunkInUsed	SIP通道仍在使用中
InvalidParameter.SipTrunkNotFound	未找到SIP通道信息
InvalidParameterValue	参数取值错误。

申请运营商白名单号码

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

用于无限频率地呼叫坐席手机

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：CreateCarrierPrivilegeNumberApplicant。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：140000000
Callers.N	是	Array of String	主叫号码，必须为实例中存在的号码，格式为0086xxxx（暂时只支持国内号码） 示例值：["008601012345678"]
Callees.N	是	Array of String	被叫号码，必须为实例中坐席绑定的手机号码，格式为0086xxxx（暂时只支持国内号码） 示例值：["008618612345678"]
Description	否	String	描述 示例值：foobar

3. 输出参数

参数名称	类型	描述
ApplicantId	Integer	申请单Id 示例值：1543
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 申请白名单号码

审批完成后，号码A呼叫号码B不会被运营商限制频率

输入示例

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

```
{
  "SdkAppId": 14000000,
  "Callers": [
    "008601012345678"
  ],
  "Callees": [
    "008618612345678"
  ]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "abc",
    "ApplicantId": 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](#)。

查询运营商白名单号码申请

最近更新时间：2025-04-25 01:12:31

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询单状态

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：DescribeCarrierPrivilegeNumberApplicants。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageNumber	否	Integer	默认0，从0开始 示例值：0
PageSize	否	Integer	默认10，最大100 示例值：10
Filters.N	否	Array of Filter	筛选条件,Name支持ApplicantId,PhoneNumber(按号码模糊查找) 示例值：[{"Name": "ApplicantId", "Values": "1"}]

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	筛选出的总申请单数量 示例值：13
Applicants	Array of CarrierPrivilegeNumberApplicant	申请单列表 示例值：[{"SdkAppId": "1400000000", "ApplicantId": "3", "Callers": ["008601012345678"], "Callees": ["008618612345678"], "State": "1", "CreateTime": "1623145120", "UpdateTime": "1623145120"}]
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 查询申请单

查询申请单

输入示例

```
POST / HTTP/1.1
Host: ccc.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeCarrierPrivilegeNumberApplicants
<公共请求参数>

{
  "SdkAppId": 14000000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "xnasdlgfas",
    "TotalCount": 1,
    "Applicants": [
      {
        "SdkAppId": 1400000000,
        "ApplicantId": 3,
        "Callers": [
          "008601012345678"
        ],
        "Callees": [
          "008618612345678"
        ],
        "State": 1,
        "Description": "foobar",
        "CreateTime": 1623145120,
        "UpdateTime": 1623145120
      }
    ]
  }
}
```

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](#)。

查询生效运营商白名单规则

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询生效运营商白名单规则

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：DescribeActiveCarrierPrivilegeNumber。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageNumber	否	Integer	默认0 示例值：0
PageSize	否	Integer	默认10，最大100 示例值：10
Filters.N	否	Array of Filter	筛选条件 Name支持PhoneNumber(按号码模糊查找) 示例值：[]

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	总数量 示例值：1
ActiveCarrierPrivilegeNumbers	Array of ActiveCarrierPrivilegeNumber	生效列表 示例值： [{"SdkAppId":1,"Caller":"008601012345678","Callee":"008601012345679"}]
PendingApplicantIds	Array of Integer	待审核单号 示例值：[1,2]
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，！ RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询生效的白名单规则

查询生效的白名单规则

输入示例

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

{
```

```
"SdkAppId": 1400000000
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "ActiveCarrierPrivilegeNumbers": [
      {
        "SdkAppId": 1,
        "Caller": "008601012345678",
        "Callee": "008601012345679",
        "CreateTime": 1729905740
      }
    ],
    "PendingApplicantIds": [
      1
    ],
    "RequestId": "abc"
  }
}
```

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](#)。

创建客户自携号码接入审核

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建客户自携号码接入审核

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：CreateOwnNumberApply。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SipTrunkId	是	Integer	SIP通道ID 示例值：1
DetailList.N	是	Array of OwnNumberApplyDetailItem	线路相关参数
Prefix	否	String	送号前缀 示例值：0

3. 输出参数

参数名称	类型	描述
ApplyId	Integer	审批单号 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）。定位问题时需提供该次请求的 RequestId。

4. 示例

示例1 创建客户自携号码接入审核实例

输入示例

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

{
  "SdkAppId": 1,
```

```
"Prefix": "1145",
"SipTrunkId": 12,
"DetailList": [
{
"CallType": 2,
"PhoneNumber": "02012345678",
"MaxCallCount": 1,
"MaxCallPSec": 1
}
]
}
```

输出示例

```
{
"Response": {
"RequestId": "foo-bar",
"ApplyId": 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

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.PermissionDenied	权限不足。
InternalServerError	内部错误。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。

错误码	描述
InvalidParameterValue	参数取值错误。
InvalidParameterValue.ApplyIDNotExist	审批单号不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。
InvalidParameterValue.PhoneNumInvalid	号码状态无效。
InvalidParameterValue.WaitingApprovalOverflow	待审核单已经超限，暂时不能再提交

修改客户自携号码审批单

最近更新时间：2025-04-25 01:12:31

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

修改客户自携号码审批单

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：ModifyOwnNumberApply。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
DetailList.N	是	Array of OwnNumberApplyDetailItem	线路相关参数
ApplyId	否	Integer	审批单号 示例值：1
Prefix	否	String	送号前缀 示例值：0

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 修改客户自携号码审批单示例

输入示例

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

{
  "SdkAppId": 1,
  "ApplyId": 1,
  "Prefix": "1145",
```

```
"DetailList": [  
  {  
    "CallType": 2,  
    "PhoneNumber": "02012345678",  
    "MaxCallCount": 1,  
    "MaxCallPSec": 1  
  }  
]  
}
```

输出示例

```
{  
  "Response": {  
    "RequestId": "foo-bar"  
  }  
}
```

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.PermissionDenied	权限不足。
InternalServerError	内部错误。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.ApplyIDNotExist	审批单号不存在。

错误码	描述
InvalidParameterValue.ErrorApplyStatus	审核单状态错误。
InvalidParameterValue.InstanceNotExist	实例不存在。
InvalidParameterValue.PhoneNumInvalid	号码状态无效。

双呼相关接口

创建外呼会话（当前仅支持双呼）

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建外呼会话，当前仅支持双呼，即先使用平台号码呼出到座席手机上，座席接听后，然后再外呼用户，而且由于运营商频率限制，座席手机号必须先加白名单，避免频控导致外呼失败。所以调用此接口前，下述操作均已完成

- 1. UserId 指定的座席已经[绑定手机号](#)
- 2. 座席绑定的手机号已经[申请并通过了外呼白名单](#)
- 3. 当前座席侧只能呼叫其手机，所以 IsForceMobile 字段当前必须为 true
- 4. 被叫不要填当前 UserId 所绑定的手机号，否则会造成占线呼叫失败

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：CreateCallOutSession。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID 示例值：14000000000
UserId	是	String	客服用户 ID，一般为客服邮箱，确保已经绑定了手机号 https://cloud.tencent.com/document/product/679/76067#.E6.AD.A5.E9.AA.A42.EF.BC.9A.E5.AE.8A 示例值：foo@tencent.com
Callee	是	String	被叫号码，须带 0086 前缀 示例值：008613012345678
Caller	否	String	主叫号码（废弃，使用Callers），须带 0086 前缀 示例值：008610086
Callers.N	否	Array of String	指定主叫号码列表，如果前面的号码失败了会自动换成下一个号码，须带 0086 前缀 示例值：["0086075512345678","0086075512345679"]
IsForceUseMobile	否	Boolean	是否强制使用手机外呼，当前只支持 true，若为 true 请确保已配置白名单 https://cloud.tencent.com/document/product/679/76744#.E6.93.8D.E4.BD.9C.E6.AD.A5.E9.AA.A4 示例值：true
UUI	否	String	自定义数据，长度限制 1024 字节 示例值：dGhpcyBpcyBhIGV4YW1wbGU

3. 输出参数

参数名称	类型	描述
SessionId	String	新创建的会话 ID 示例值：adc61b8-dbb9-4c20-87bf-c0744c04bade
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

参数名称	类型	描述
		要提供该次请求的 RequestId。

4. 示例

示例1 外呼接口调用示例

在不加载前端 SDK 的场景下，可以通过在后台触发此接口进行外呼，当前只支持通过手机端回拨外呼（先拨通客服手机），并且确认已经申请并通过添加外呼白名单。

输入示例

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

{
  "IsForceUseMobile": "true",
  "Callee": "008612300000000",
  "UI": "fooandbar",
  "UserId": "FooOrBar@tencent.com",
  "SdkAppId": 1400000000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "SessionId": "6bb56a09278740bc80c5dc6dab783eff"
  }
}
```

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.CallOutFailed	外呼失败。
FailedOperation.CalleelsLimited	被叫号码外呼受限。
FailedOperation.CallerOverFrequency	主叫号码外呼超频。
FailedOperation.NoCallOutNumber	无可用的外呼号码。
FailedOperation.SeatStatusBusy	坐席处于忙碌状态。
InternalError.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.AccountNotExist	账号不存在。
OperationDenied.NotInWhiteList	不在白名单中。
UnsupportedOperation	操作不支持。

自动外呼相关接口

创建自动外呼任务

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

创建自动外呼任务

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：CreateAutoCalloutTask。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
NotBefore	是	Integer	任务起始时间戳，Unix 秒级时间戳 示例值：1642500621
Callees.N	是	Array of String	被叫号码列表 示例值：[008613912341234,008613912341235]
Callers.N	是	Array of String	主叫号码列表 示例值：[008601012345678]
IvrId	否	Integer	呼叫使用的 IVR Id，不填时需要填写 AIAgentId 示例值：3
Name	否	String	任务名 示例值：任务1
Description	否	String	任务描述 示例值：任务描述1
NotAfter	否	Integer	任务停止时间戳，Unix 秒级时间戳 示例值：1642500622
Tries	否	Integer	最大尝试次数，1-3 次 示例值：1
Variables.N	否	Array of Variable	自定义变量（仅高级版支持） 示例值：{"a":1}
UUI	否	String	UUI 示例值：foobar
CalleeAttributes.N	否	Array of CalleeAttribute	被叫属性 示例值：[{"Callee":"foobar","UUI":"11","Variables":{"foo":"bar"}}]
TimeZone	否	String	IANA 时区名称，参考 https://datatracker.ietf.org/doc/html/draft-ietf-netmod-iana-timezones

参数名称	必选	类型	描述
			示例值：Asia/Shanghai
AvailableTime.N	否	Array of TimeRange	可用时间段 示例值：[{"StartTime":"09:00:00","EndTime":"11:30:00"}, {"StartTime":"14:00:00","EndTime":"18:00:00"}]
AIAgentId	否	Integer	智能体 ID，不填写时需要填写 IvrlId 示例值：5845

3. 输出参数

参数名称	类型	描述
TaskId	Integer	任务Id 示例值：2
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 创建任务

创建任务

输入示例

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

{
  "SdkAppId": 1400123455,
  "NotBefore": 1642500621,
  "Callees": [
    "008613012345678"
  ],
  "Callers": [
    "008602012345678"
  ],
  "IvrId": 8
}
```

输出示例

```
{
  "Response": {
    "TaskId": 1,
    "RequestId": "a561d5e7-2680-481f-b86e-a9ad7316f4d5"
  }
}
```

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.PermissionDenied	权限不足。
InternalServerError	内部错误。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.ExperienceAccount	使用体验账号的智能体不支持此类操作

批量查询自动外呼任务

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

批量查询自动外呼任务

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：DescribeAutoCalloutTasks。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	是	Integer	分页大小 示例值：10
PageNumber	是	Integer	页数 示例值：0

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	总数 示例值：5
Tasks	Array of AutoCalloutTaskInfo	任务列表 示例值： [{"TaskId": "3241", "IvrId": "1", "CalleeCount": "1", "Name": "foobar", "NotBefore": "0", "NotAfter": "0", "State": "1", "Caller": "foobar"}]
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 示例

输入示例

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



```
{
  "PageNumber": 0,
  "PageSize": 10,
  "SdkAppId": 1
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "Tasks": [
      {
        "TaskId": 3241,
        "IvrId": 1,
        "CalleeCount": 1,
        "Name": "foobar",
        "NotBefore": 0,
        "NotAfter": 0,
        "State": 1,
        "Callers": [
          "008612345123456"
        ]
      }
    ],
    "RequestId": "foo-bar"
  }
}
```

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
InternalError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。

停止自动外呼任务

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

停止自动外呼任务

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：StopAutoCalloutTask。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
TaskId	是	Integer	任务Id 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 停止任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "TaskId": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "xxx"
```

```
}  
}
```

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	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
UnsupportedOperation	操作不支持。

查询自动外呼任务详情

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询自动外呼任务详情

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：DescribeAutoCalloutTask。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
TaskId	是	Integer	任务Id 示例值：3

3. 输出参数

参数名称	类型	描述
Name	String	任务名 示例值：任务1
Description	String	任务描述 示例值：任务描述
NotBefore	Integer	任务起始时间戳 示例值：1729905740
NotAfter	Integer	任务结束时间戳 注意：此字段可能返回 null，表示取不到有效值。 示例值：1729905750
Callers	Array of String	主叫列表 示例值：["008610086"]
Callees	Array of AutoCalloutTaskCalleeInfo	被叫信息列表 示例值：[{"State":1,"Callee":"foobar","Sessions":["foobar"]}]]
IvrId	Integer	任务使用的IvrId 示例值：2
State	Integer	任务状态 0初始 1运行中 2已完成 3结束中 4已终止 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 查询任务

输入示例

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

{
  "SdkAppId": 1,
  "TaskId": 2
}
```

输出示例

```
{
  "Response": {
    "IvrId": 1,
    "Name": "foobar",
    "NotBefore": 1729905740,
    "NotAfter": 1729905750,
    "State": 1,
    "Callers": [
      "008610086"
    ],
    "RequestId": "foobar",
    "Callees": [
      {
        "State": 1,
        "Callee": "foobar",
        "Sessions": [
          "foobar"
        ]
      }
    ],
    "Description": "foobar"
  }
}
```

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	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。

停止预测式外呼任务

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

停止预测式外呼任务

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：AbortPredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

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 停止预测式外呼任务示例

停止预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例


```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InvalidParameterValue	参数取值错误。

创建预测式外呼任务

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建预测式外呼任务

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：CreatePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Name	是	String	任务名称 示例值：新能源SUV推荐
Callees.N	是	Array of String	被叫列表，支持 E.164 或不带国家码形式的号码 示例值：+8613012345678
Callers.N	是	Array of String	主叫列表，使用管理端展示的号码格式 示例值：0086075586013388
CallOrder	是	Integer	被叫呼叫顺序 0 随机 1 顺序 示例值：0
SkillGroupId	是	Integer	使用的座席技能组 ID 示例值：255
Priority	是	Integer	相同应用内多个任务运行优先级，从高到底 1 - 5 示例值：3
ExpectedAbandonRate	是	Integer	预期呼损率，百分比，5 - 50 示例值：10
RetryInterval	是	Integer	呼叫重试间隔时间，单位秒，60 - 86400 示例值：1800
StartTime	是	Integer	任务启动时间，Unix 时间戳，到此时间后会启动任务 示例值：1708483433
EndTime	是	Integer	任务结束时间，Unix 时间戳，到此时间后会终止任务 示例值：1708485433
IVRId	否	Integer	指定的 IVR Id 示例值：4600
RetryTimes	否	Integer	呼叫重试次数，0 - 2 示例值：2
Variables.N	否	Array of Variable	自定义变量 示例值：[{"Key":"foo","Value":"bar"}]

参数名称	必选	类型	描述
UUI	否	String	UUI 示例值: foobar
CalleeAttributes.N	否	Array of CalleeAttribute	被叫属性 示例值: [{"Callee": "13012345678", "UUI": "foobar", "Variables": [{"Key": "foo": "Value": "bar"}]}]
TimeZone	否	String	IANA 时区名称, 参考 https://datatracker.ietf.org/doc/html/draft-ietf-netmod-iana-timezones 示例值: Asia/Shanghai
AvailableTime.N	否	Array of TimeRange	可用时间段 示例值: [{"StartTime": "09:00:00", "EndTime": "11:30:00"}, {"StartTime": "14:00:00", "EndTime": "18:00:00"}]

3. 输出参数

参数名称	类型	描述
CampaignId	Integer	生成的任务 ID 示例值: 2569
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 创建预测式外呼任务示例

创建预测式外呼任务

输入示例

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

```
{
  "SdkAppId": 1400000000,
  "Name": "新能源SUV推荐",
  "Callees": [
    "+8613012345678",
    "+8613012345679",
    "+8613012345670"
  ],
  "Callers": [
    "0086075586013388"
  ],
  "CallOrder": 0,
  "SkillGroupId": 255,
  "IVRId": 4600,
  "Priority": 3,
  "ExpectedAbandonRate": 10,
  "RetryTimes": 1,
  "RetryInterval": 1800,
  "StartTime": 1708483433,
```

```
"EndTime": 1708485433
}
```

输出示例

```
{
  "Response": {
    "CampaignId": 2569,
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
LimitExceeded.OutOfCountLimit	超出数量限制。

删除预测式外呼任务

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

删除预测式外呼任务

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：DeletePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

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 删除预测式外呼任务示例

删除预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

查询座席巡航式外呼任务

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询 座席巡航式外呼任务

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：DescribeAgentCruiseDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

3. 输出参数

参数名称	类型	描述
Name	String	任务名称 示例值：新能源SUV推荐
Agent	String	座席账号 示例值：foo@tencent.com
ConcurrencyNumber	Integer	单轮并发呼叫量 1-20 示例值：3
StartTime	Integer	任务启动时间，Unix 时间戳，到此时间后会自动启动任务 示例值：1708483433
EndTime	Integer	任务结束时间，Unix 时间戳，到此时间后会自动终止任务 示例值：1708485433
CallOrder	Integer	被叫呼叫顺序 0 随机 1 顺序 示例值：0
UII	String	调用方自定义数据，最大长度 1024 示例值：foobar
State	Integer	任务状态 0 未启动 1 运行中 2 已完成 3 已终止 示例值：2
TotalCalleeCount	Integer	被叫总数 示例值：200
CalledCalleeCount	Integer	已呼被叫数 示例值：200

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 查询 座席巡航式外呼任务示例

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "Name": "新能源SUV推荐",
    "Agent": "foo@tencent.com",
    "ConcurrencyNumber": 3,
    "CallOrder": 0,
    "StartTime": 1708483433,
    "EndTime": 1708485433,
    "UUI": "",
    "State": 2,
    "TotalCalleeCount": 200,
    "CalledCalleeCount": 200,
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

查询预测式外呼任务

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询预测式外呼任务

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：DescribePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

3. 输出参数

参数名称	类型	描述
CampaignId	Integer	任务 ID 示例值：2569
Name	String	任务名称 示例值：新能源SUV推荐
CallOrder	Integer	被叫呼叫顺序 0 随机 1 顺序 示例值：0
SkillGroupId	Integer	使用的座席技能组 ID 示例值：255
IVRId	Integer	指定的 IVR ID 示例值：4600
Priority	Integer	相同应用内多个任务运行优先级，从高到底 1 - 5 示例值：3
ExpectedAbandonRate	Integer	预期呼损率，百分比，5 - 50 示例值：10
RetryTimes	Integer	呼叫重试次数，0 - 2 示例值：2
RetryInterval	Integer	呼叫重试间隔时间，单位秒，60 - 86400 示例值：1800
StartTime	Integer	任务启动时间，Unix 时间戳，到此时间后会自动启动任务 示例值：1708483433

参数名称	类型	描述
EndTime	Integer	任务结束时间，Unix 时间戳，到此时间后会自动终止任务 示例值：1708485433
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 查询预测式外呼任务示例

查询预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "CampaignId": 2569,
    "Name": "新能源SUV推荐",
    "CallOrder": 0,
    "SkillGroupId": 255,
    "IVRId": 4600,
    "Priority": 3,
    "ExpectedAbandonRate": 10,
    "RetryTimes": 1,
    "RetryInterval": 1800,
    "StartTime": 1708483433,
    "EndTime": 1708485433,
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

查询预测式外呼任务列表

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询预测式外呼任务列表

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：DescribePredictiveDialingCampaigns。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	是	Integer	分页尺寸，最大为 100 示例值：25
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0
Name	否	String	查询任务列表名称关键字 示例值：推荐
SkillGroupId	否	Integer	查询任务列表技能组 ID 示例值：255

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	数据总量 示例值：66
CampaignList	Array of DescribePredictiveDialingCampaignsElement	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 查询预测式外呼任务列表示例

查询预测式外呼任务列表

输入示例

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

```
{
  "SdkAppId": 1400000000,
  "PageSize": 25,
  "PageNumber": 0
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 66,
    "CampaignList": [
      {
        "CampaignId": 2567,
        "Name": "新能源SUV推荐1",
        "Status": 0,
        "StatusReason": 0,
        "CalleeCount": 1000,
        "FinishedCalleeCount": 0,
        "Priority": 3,
        "SkillGroupId": 255
      },
      {
        "CampaignId": 2568,
        "Name": "新能源SUV推荐2",
        "Status": 0,
        "StatusReason": 0,
        "CalleeCount": 1000,
        "FinishedCalleeCount": 0,
        "Priority": 3,
        "SkillGroupId": 255
      },
      {
        "CampaignId": 2569,
        "Name": "新能源SUV推荐3",
        "Status": 0,
        "StatusReason": 0,
        "CalleeCount": 1000,
        "FinishedCalleeCount": 0,
        "Priority": 3,
        "SkillGroupId": 255
      }
    ],
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

查询预测式外呼呼叫列表

最近更新时间：2025-04-25 01:12:26

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询预测式外呼呼叫列表

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：DescribePredictiveDialingSessions。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	生成的任务 ID 示例值：2569
PageSize	是	Integer	分页尺寸，最大为 1000 示例值：25
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	数据总量 示例值：3000
SessionList	Array of String	呼叫的 session id 列表，通过 https://cloud.tencent.com/document/product/679/47714 可以批量获取呼叫详细话单 示例值：["e4b2d3f1-8c27-4a9d-bef1-0b412a7c5e8d", "a3e8f4c9-2b7d-4e1f-9d82-6c5a8b0d3f1e", "f7d2b4e9-1c8a-5d9e-b0f3-2e1a3c4b5d6f"]
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 查询预测式外呼呼叫列表示例

查询预测式外呼呼叫列表

输入示例

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



```
X-TC-Action: DescribePredictiveDialingSessions
```

<公共请求参数>

```
{
  "SdkAppId": 1400000000,
  "CampaignId": 2569,
  "PageSize": 25,
  "PageNumber": 0
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 66,
    "SessionList": [
      "a5be1044-e8b0-4f02-9b25-64baee24374b",
      "22c3e170-f307-47c8-9f10-1b77413a646f",
      "e8557b17-d68b-4475-9a97-639a1108887b"
    ],
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

暂停预测式外呼任务

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

暂停预测式外呼任务

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：PausePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

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 暂停预测式外呼任务示例

暂停预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InvalidParameterValue	参数取值错误。

恢复预测式外呼任务

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

恢复预测式外呼任务

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：ResumePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

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 恢复预测式外呼任务示例

恢复预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InvalidParameterValue	参数取值错误。

更新预测式外呼任务

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

任务未启动前，更新预测式外呼任务。

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：UpdatePredictiveDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	生成的任务 ID 示例值：2569
Name	是	String	任务名称 示例值：新能源SUV推荐
Callees.N	是	Array of String	被叫列表，支持 E.164 或不带国家码形式的号码 示例值：+8613012345678
Callers.N	是	Array of String	主叫列表，使用管理端展示的号码格式 示例值：0086075586013388
CallOrder	是	Integer	被叫呼叫顺序 0 随机 1 顺序 示例值：0
SkillGroupId	是	Integer	使用的座席技能组 ID 示例值：255
Priority	是	Integer	相同应用内多个任务运行优先级，从高到底 1 - 5 示例值：3
ExpectedAbandonRate	是	Integer	预期呼损率，百分比，5 - 50 示例值：10
RetryInterval	是	Integer	呼叫重试间隔时间，单位秒，60 - 86400 示例值：1800
StartTime	是	Integer	任务启动时间，Unix 时间戳，到此时间后会自动启动任务 示例值：1708483433
EndTime	是	Integer	任务结束时间，Unix 时间戳，到此时间后会自动终止任务 示例值：1708485433
IVRId	否	Integer	指定的 IVR ID 示例值：4600
RetryTimes	否	Integer	呼叫重试次数，0 - 2 示例值：2

参数名称	必选	类型	描述
Variables.N	否	Array of Variable	自定义变量 示例值: [{"Key":"foo","Value":"bar"}]
UUI	否	String	UUI 示例值: foobar
CalleeAttributes.N	否	Array of CalleeAttribute	被叫属性 示例值: [{"Callee":"13012345678","UUI":"foobar","Variables":[{"Key":"foo":"Value":"bar"}]}]

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 更新预测式外呼任务示例

更新预测式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569,
  "Name": "新能源SUV推荐",
  "Callees": [
    "+8613012345678",
    "+8613012345679",
    "+8613012345670"
  ],
  "Callers": [
    "0086075586013388"
  ],
  "CallOrder": 0,
  "SkillGroupId": 255,
  "IVRId": 4600,
  "Priority": 3,
  "ExpectedAbandonRate": 10,
  "RetryTimes": 1,
  "RetryInterval": 1800,
  "StartTime": 1708483433,
  "EndTime": 1708485433
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
UnsupportedOperation	操作不支持。

话机相关接口

创建话机账号

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

创建话机账号

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：CreateExtension。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
ExtensionId	是	String	分机号 示例值：1001
ExtensionName	是	String	分机名称 示例值：分机1
SkillGroupIds.N	否	Array of Integer	绑定的技能组列表 示例值：[1,2,3,4,5]
Relation	否	String	绑定的坐席邮箱 示例值：aaa@bbb.com

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 创建话机账号

创建话机账号示例

输入示例

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

<公共请求参数>

```
{
  "SdkAppId": 140000000,
  "ExtensionId": "1001",
  "ExtensionName": "分机1"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127"
  }
}
```

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	内部数据库访问失败。
InvalidParameterValue.InstanceNotExist	实例不存在。

删除话机账号

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

删除话机账号

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：DeleteExtension。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
ExtensionId	是	String	分机号 示例值：1001

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 删除话机账号

输入示例

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

{
  "SdkAppId": 1400000000,
  "ExtensionId": "1001"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127"
  }
}
```

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](#)。

修改话机账号(绑定技能组、绑定坐席账号)

最近更新时间：2025-04-25 01:12:24

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

修改话机账号(绑定技能组、绑定坐席账号)

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：ModifyExtension。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
ExtensionId	是	String	分机号 示例值：1001
ExtensionName	否	String	分机名称 示例值：分机1
SkillGroupIds.N	否	Array of Integer	所属技能组列表 示例值：[1,2,3,4,5,6]
Relation	否	String	绑定坐席邮箱账号 示例值：xxx@xxx.com

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 修改话机账号

输入示例

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

{
```

```
"SdkAppId": 140000000,
"ExtensionId": "1001",
"ExtensionName": "分机1",
"SkillGroupIds": [
  1000
],
"Relation": "xxx@xxx.com"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127"
  }
}
```

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
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue.InstanceNotExist	实例不存在。

获取话机信息

最近更新时间：2025-04-25 01:12:25

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取话机信息

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：DescribeExtension。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
ExtensionId	是	String	分机号 示例值：1001

3. 输出参数

参数名称	类型	描述
ExtensionId	String	分机号 示例值：1001
ExtensionDomain	String	域名 示例值：14000000.tccc.qcloud.com
Password	String	注册密码 示例值：foobarfoobar
OutboundProxy	String	代理服务器地址 示例值：sip.tcc.qcloud.com
Transport	String	传输协议 示例值：tls
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 获取话机信息

输入示例

```
POST / HTTP/1.1
Host: ccc.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeExtension
<公共请求参数>

{
  "SdkAppId": 140000000,
  "ExtensionId": "1001"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127",
    "ExtensionId": "1001",
    "ExtensionDomain": "1400000214.tccc.qcloud.com",
    "Password": "0pMnxfdsdLT6G",
    "OutboundProxy": "sip.tccc.qcloud.com:5061",
    "Transport": "tls"
  }
}
```

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	内部数据库访问失败。

查询话机列表信息

最近更新时间：2025-04-25 01:12:24

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询话机列表信息

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：DescribeExtensions。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageNumber	是	Integer	分页页号（从0开始） 示例值：0
ExtensionIds.N	否	Array of String	筛选分机号列表 示例值：["1001","1002"]
PageSize	否	Integer	分页大小 示例值：50
FuzzingKeyWord	否	String	模糊查询字段（模糊查询分机号、分机名称、坐席邮箱、坐席名称） 示例值：foobar
IsNeedStatus	否	Boolean	是否需要返回话机当前状态 示例值：false

3. 输出参数

参数名称	类型	描述
Total	Integer	查询总数 示例值：1
ExtensionList	Array of ExtensionInfo	话机信息列表 示例值： [{"SdkAppId":1400000014,"FullExtensionId":"1001@140000014.tccc.qcloud.com","ExtensionId":"1001","Sk
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

4. 示例

示例1 查询话机列表信息

输入示例

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

```
{
  "PageNumber": 0,
  "PageSize": 35,
  "SdkAppId": 1400264214,
  "ExtensionIds": [
    "1001"
  ]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127",
    "Total": 1,
    "ExtensionList": [
      {
        "SdkAppId": 1400000014,
        "FullExtensionId": "1001@1400000014.tccc.qcloud.com",
        "ExtensionId": "1001",
        "SkillGroupId": "1532",
        "ExtensionName": "lulu",
        "CreateTime": 0,
        "ModifyTime": 0,
        "Status": 0,
        "Register": false,
        "Relation": "foobar@tencent.com",
        "RelationName": "Foobar"
      }
    ]
  }
}
```

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.DBError	内部数据库访问失败。
InvalidParameterValue.InstanceNotExist	实例不存在。

重置话机注册密码

最近更新时间：2025-04-25 01:12:24

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

重置话机注册密码

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：ResetExtensionPassword。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
ExtensionId	是	String	分机号 示例值：1001

3. 输出参数

参数名称	类型	描述
Password	String	重置后密码 示例值：foobar
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 重置话机注册密码

输入示例

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

{
  "SdkAppId": 1400000000,
  "ExtensionId": "1001"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127",
    "Password": "xxxxxxxxxxxxxxxxxx"
  }
}
```

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](#)。

电话服务记录相关接口

获取电话服务记录与录音

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

获取电话服务记录与录音

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：DescribeTelCdr。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
StartTimeStamp	是	Integer	起始时间戳，Unix 秒级时间戳，最大支持近180天。 示例值：1611631881
EndTimeStamp	是	Integer	结束时间戳，Unix 秒级时间戳，结束时间与开始时间的区间范围小于90天。 示例值：1611631881
Limit	否	Integer	返回数据条数，上限（废弃） 示例值：0
Offset	否	Integer	偏移（废弃） 示例值：0
SdkAppId	否	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	否	Integer	分页尺寸（必填），上限 100 示例值：100
PageNumber	否	Integer	分页页码（必填），从 0 开始 示例值：0
Phones.N	否	Array of String	按手机号筛选 示例值：["008613012345678"]
SessionIds.N	否	Array of String	按SessionId筛选 示例值：["7cb26a84-83b2-42d8-a19b-7dae78e9b78e"]

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	话单记录总数 示例值：2
TelCdrList	Array of TelCdrInfo	话单记录

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 获取电话话单示例

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribeTelCdr
&SdkAppId=1400000000
&StartTimeStamp=1590547606
&EndTimeStamp=1590147606
&PageSize=10
&PageNumber=0
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "TelCdrList": [
      {
        "AcceptTimestamp": 1729756307,
        "AsrStatus": "Complete",
        "AsrUrl": "https://xxxxxx",
        "Callee": "00860200000000",
        "Caller": "0086075500000",
        "CallerLocation": "广东深圳固话",
        "CustomRecordURL": "https://xxxxxxx.mp3",
        "Direction": 0,
        "Duration": 8,
        "EndStatus": 1,
        "EndStatusString": "ok",
        "EndedTimestamp": 1729756315,
        "HungUpSide": "seat",
        "IVRDuration": 15,
        "IVRKeyPressed": [
          "香蕉"
        ],
        "IVRKeyPressedEx": [
          {
            "Key": "香蕉",
            "Label": "香蕉",
            "NodeLabel": "语音识别导航",
            "OriginalContent": "香蕉。",
            "TTSPrompt": "输入您喜欢吃的水果",
            "Timestamp": 1729756303211
          }
        ],
        "PostIVRKeyPressed": [
          {
            "Key": "1",
            "Label": "不满意",
```

```
"NodeLabel": "",
"OriginalContent": "",
"TTSPrompt": "",
"Timestamp": 0
},
{
  "ProtectedCallee": "",
  "ProtectedCaller": "",
  "QueuedSkillGroupId": 2522,
  "QueuedSkillGroupName": "test-tel",
  "QueuedTimestamp": 1729756306,
  "RecordId": "e9fdbff5-2e35-4480-83f7-81917e723fa5",
  "RecordURL": "https://xxxxxxx.mp3",
  "Remark": "skillGroupId--34588",
  "RingTimestamp": 1729756306,
  "SeatUser": {
    "Mail": "xxxxxxx@tencent.com",
    "Name": "xiaoming",
    "Nick": "guanjia",
    "Phone": "00861580000000",
    "SkillGroupNameList": [
      "1539",
      "2522",
      "3551"
    ],
    "StaffNumber": "20012",
    "UserId": "xxxxxxx@tencent.com"
  },
  "ServeParticipants": [
    {
      "AcceptTimestamp": 1729756306,
      "CustomRecordURL": "https://xxxxx.mp3",
      "EndStatusString": "ok",
      "EndedTimestamp": 1729756312,
      "Mail": "xxxxxxx@tencent.com",
      "Phone": "",
      "RecordId": "96eaa575-be4c-48f1-99b8-9722f2f816ec",
      "RecordURL": "https://xxxxxx.mp3",
      "RingTimestamp": 1729756306,
      "Sequence": 0,
      "SkillGroupId": 2522,
      "SkillGroupName": "test-tel",
      "StartTimestamp": 1729756306,
      "TransferFrom": "",
      "TransferFromType": "",
      "TransferTo": "",
      "TransferToType": "",
      "Type": "staffSeat"
    }
  ],
  "SessionId": "0955de87-c86d-4457-aac2-bee3d0a0b67a",
  "SkillGroup": "test-tel",
  "SkillGroupId": 2522,
  "StartTimestamp": 1729756291,
  "Time": 1729756291,
  "UUI": "",
  "Uui": ""
}
```



```
"VoicemailAsrURL": [],
"VoicemailRecordURL": []
},
],
"RequestId": "abc"
}
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

获取主被叫受保护的电话服务记录与录音

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取主被叫受保护的电话服务记录与录音

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：DescribeProtectedTelCdr。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
StartTimeStamp	是	Integer	起始时间戳，Unix 秒级时间戳 示例值：1611631881
EndTimeStamp	是	Integer	结束时间戳，Unix 秒级时间戳 示例值：1611631881
SdkAppId	是	Integer	应用 ID，可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
PageSize	是	Integer	分页尺寸，上限 100 示例值：100
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	话单记录总数 示例值：2
TelCdrList	Array of TelCdrInfo	话单记录
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 获取主被叫受保护的电话话单示例

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribeProtectedTelCdr
&SdkAppId=1400000000
```

```
&StartTimeStamp=1590547606
&EndTimeStamp=1590147606
&PageSize=10
&PageNumber=0
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "TelCdrList": [
      {
        "AcceptTimeStamp": 1729756307,
        "AsrStatus": "Complete",
        "AsrUrl": "https://xxxxxx",
        "Callee": "",
        "Caller": "",
        "CallerLocation": "广东深圳固话",
        "CustomRecordURL": "https://xxxxxxx.mp3",
        "Direction": 0,
        "Duration": 8,
        "EndStatus": 1,
        "EndStatusString": "ok",
        "EndedTimeStamp": 1729756315,
        "HungUpSide": "seat",
        "IVRDuration": 15,
        "IVRKeyPressed": [
          "香蕉"
        ],
        "IVRKeyPressedEx": [
          {
            "Key": "香蕉",
            "Label": "香蕉",
            "NodeLabel": "语音识别导航",
            "OriginalContent": "香蕉。",
            "TTSPrompt": "输入您喜欢吃的水果",
            "Timestamp": 1729756303211
          }
        ],
        "PostIVRKeyPressed": [
          {
            "Key": "1",
            "Label": "不满意",
            "NodeLabel": "",
            "OriginalContent": "",
            "TTSPrompt": "",
            "Timestamp": 0
          }
        ],
        "ProtectedCallee": "ABABABABA",
        "ProtectedCaller": "CDCDCDCDC",
        "QueuedSkillGroupId": 2522,
        "QueuedSkillGroupName": "test-tel",
        "QueuedTimeStamp": 1729756306,
        "RecordId": "e9fdbff5-2e35-4480-83f7-81917e723fa5",
```

```
"RecordURL": "https://xxxxxxx.mp3",
"Remark": "skillGroupId--34588",
"RingTimestamp": 1729756306,
"SeatUser": {
  "Mail": "xxxxxxx@tencent.com",
  "Name": "xiaoming",
  "Nick": "guanjia",
  "Phone": "00861580000000",
  "SkillGroupNameList": [
    "1539",
    "2522",
    "3551"
  ],
  "StaffNumber": "20012",
  "UserId": "xxxxxxx@tencent.com"
},
"ServeParticipants": [
  {
    "AcceptTimestamp": 1729756306,
    "CustomRecordURL": "https://xxxxx.mp3",
    "EndStatusString": "ok",
    "EndedTimestamp": 1729756312,
    "Mail": "xxxxxxx@tencent.com",
    "Phone": "",
    "RecordId": "96eaa575-be4c-48f1-99b8-9722f2f816ec",
    "RecordURL": "https://xxxxxxx.mp3",
    "RingTimestamp": 1729756306,
    "Sequence": 0,
    "SkillGroupId": 2522,
    "SkillGroupName": "test-tel",
    "StartTimestamp": 1729756306,
    "TransferFrom": "",
    "TransferFromType": "",
    "TransferTo": "",
    "TransferToType": "",
    "Type": "staffSeat"
  }
],
"SessionId": "0955de87-c86d-4457-aac2-bee3d0a0b67a",
"SkillGroup": "test-tel",
"SkillGroupId": 2522,
"StartTimestamp": 1729756291,
"Time": 1729756291,
"UI": "",
"Ui": "",
"VoicemailAsrURL": [],
"VoicemailRecordURL": []
},
"RequestId": "abc"
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

获取 PSTN 活动会话列表

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取当前正在通话的会话列表

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：DescribePSTNActiveSessionList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Offset	是	Integer	数据偏移 示例值：0
Limit	是	Integer	返回的数据条数，最大 25 示例值：25

3. 输出参数

参数名称	类型	描述
Total	Integer	列表总条数 示例值：1
Sessions	Array of PSTNSessionInfo	列表内容
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 PSTN 活动会话列表示例

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribePSTNActiveSessionList
&SdkAppId=1400000000
&Offset=0
&Limit=25
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Total": 1,
    "RequestId": "53bccf04-0870-4520-8614-f4bdddFd68df",
    "Sessions": [
      {
        "Direction": 1,
        "AcceptTimestamp": "1607702329",
        "ProtectedCaller": "",
        "RoomID": "32929373",
        "StartTimestamp": "1607702199",
        "Caller": "00864009282737",
        "ProtectedCallee": "",
        "StaffNumber": "1007",
        "SessionID": "0cf5be1b-de75-4445-a0c4-8dff3fa4b68b",
        "StaffEmail": "foo@tencent.com",
        "RingTimestamp": 1607702299,
        "Callee": "00864001783747",
        "SessionStatus": "InProgress"
      }
    ]
  }
}
```

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.PermissionDenied	权限不足。
InternalServerError	内部错误。
LimitExceeded.OutOfCountLimit	超出数量限制。

获取呼入实时数据统计指标

最近更新时间：2025-04-25 01:12:28

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取呼入实时数据统计指标

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：DescribeCallInMetrics。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
EnabledSkillGroup	否	Boolean	是否返回技能组维度信息，默认“是” 示例值：true
EnabledNumber	否	Boolean	是否返回线路维度信息，默认“否” 示例值：false
GroupIdList.N	否	Array of Integer	筛选技能组列表 示例值：[1,2,3,4,5]

3. 输出参数

参数名称	类型	描述
Timestamp	Integer	时间戳 示例值：1
TotalMetrics	CallInMetrics	总体指标 示例值： { "IvrCount":0, "QueueCount":0, "RingCount":0, "AcceptCount":0, "TransferOuterCount":0, "Ma:
NumberMetrics	Array of CallInNumberMetrics	线路维度指标
SkillGroupMetrics	Array of CallInSkillGroupMetrics	技能组维度指标 示例值：[{"SkillGroupId":0, "Name": "客服一组"}]
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 Re

4. 示例

示例1 获取呼入实时数据统计指标实例

获取呼入实时数据统计指标实例。

输入示例

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

{
  "EnabledNumber": true,
  "EnabledSkillGroup": true,
```

```
"SdkAppId": 0
}
```

输出示例

```
{
  "Response": {
    "Timestamp": 0,
    "TotalMetrics": {
      "IvrCount": 0,
      "QueueCount": 0,
      "RingCount": 0,
      "AcceptCount": 0,
      "TransferOuterCount": 0,
      "MaxQueueDuration": 0,
      "AvgQueueDuration": 0,
      "MaxRingDuration": 0,
      "AvgRingDuration": 0,
      "MaxAcceptDuration": 0,
      "AvgAcceptDuration": 0
    },
    "NumberMetrics": [
      {
        "Number": "0218888888",
        "Metrics": {
          "IvrCount": 0,
          "QueueCount": 0,
          "RingCount": 0,
          "AcceptCount": 0,
          "TransferOuterCount": 0,
          "MaxQueueDuration": 0,
          "AvgQueueDuration": 0,
          "MaxRingDuration": 0,
          "AvgRingDuration": 0,
          "MaxAcceptDuration": 0,
          "AvgAcceptDuration": 0
        },
        "SkillGroupMetrics": [
          {
            "SkillGroupId": 0,
            "Metrics": {
              "IvrCount": 0,
              "QueueCount": 0,
              "RingCount": 0,
              "AcceptCount": 0,
              "TransferOuterCount": 0,
              "MaxQueueDuration": 0,
              "AvgQueueDuration": 0,
              "MaxRingDuration": 0,
              "AvgRingDuration": 0,
              "MaxAcceptDuration": 0,
              "AvgAcceptDuration": 0
            },
            "Name": "客服一组"
          }
        ]
      }
    ]
  }
}
```

```
}
],
"SkillGroupMetrics": [
{
"SkillGroupId": 0,
"Name": "abc"
}
],
"RequestId": "abc"
}
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

获取 PSTN 会话信息

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取 PSTN 会话信息

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：DescribeTelSession。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话 ID 示例值：6bb56a09-2787-40bc-80c5-dc6dab783eff

3. 输出参数

参数名称	类型	描述
Session	PSTNSession	会话信息
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 PSTN 会话信息

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribeTelSession
&SdkAppId=1400000000
&SessionId=0cf5be1b-de75-4445-a0c4-8dff3fa4b68b
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "53bccf04-0870-4520-8614-f4bdddFd68df",
    "Session": {
      "SessionID": "0cf5be1b-de75-4445-a0c4-8dff3fa4b68b",
```

```
"RoomID": "32929373",
"Caller": "00864009282737",
"Callee": "00864001783747",
"StartTimestamp": 1607702199,
"RingTimestamp": 1607702299,
"AcceptTimestamp": 1607702329,
"StaffEmail": "foo@tencent.com",
"StaffNumber": "1007",
"SessionStatus": "InProgress",
"Direction": 1,
"OutBoundCaller": "",
"OutBoundCallee": "",
"ProtectedCallee": "",
"ProtectedCaller": ""
}
}
}
```

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.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

按实例获取电话消耗统计

最近更新时间：2025-04-25 01:12:27

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

按实例获取电话消耗统计

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：DescribeTelCallInfo。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
StartTimeStamp	是	Integer	起始时间戳，Unix 时间戳(查询维度仅支持天，例如查询5月1日应该传startTime:"2023-05-01 00:00:00","endTime":"2023-05-01 23:59:59"的时间戳,查5月1日和5月2日的应该传startTime:"2023-05-01 00:00:00","endTime":"2023-05-02 23:59:59"的时间戳) 示例值：0
EndTimeStamp	是	Integer	结束时间戳，Unix 时间戳，查询时间范围最大为90天(查询维度仅支持天，例如查询5月1日应该传startTime:"2023-05-01 00:00:00","endTime":"2023-05-01 23:59:59"的时间戳,查5月1日和5月2日的应该传startTime:"2023-05-01 00:00:00","endTime":"2023-05-02 23:59:59"的时间戳) 示例值：0
SdkAppldList.N	是	Array of Integer	应用ID列表，多个ID时，返回值为多个ID使用总和 示例值：["0\n"]

3. 输出参数

参数名称	类型	描述
TelCallOutCount	Integer	呼出套餐包消耗分钟数 示例值：10
TelCallInCount	Integer	呼入套餐包消耗分钟数 示例值：10
SeatUsedCount	Integer	坐席使用统计个数 示例值：50
VOIPCallInCount	Integer	音频套餐包消耗分钟数 示例值：10
AsrOfflineCount	Integer	离线语音转文字套餐包消耗分钟数 示例值：1
AsrRealtimeCount	Integer	实时语音转文字套餐包消耗分钟数 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 获取电话呼叫统计信息示例

获取电话呼叫统计信息

输入示例

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

```
{
  "StartTimeStamp": "0",
  "EndTimeStamp": "0",
  "SdkAppIdList": [
    "0"
  ]
}
```

输出示例

```
{
  "Response": {
    "TelCallOutCount": 0,
    "TelCallInCount": 0,
    "SeatUsedCount": 0,
    "VOIPCallInCount": 0,
    "AsrOfflineCount": 0,
    "AsrRealtimeCount": 0,
    "RequestId": "abc"
  }
}
```

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.DuplicatedAccount	重复账号。
FailedOperation.PermissionDenied	权限不足。
InternalError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

购买相关接口

获取用户购买信息列表

最近更新时间：2025-04-25 01:12:23

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

获取用户购买信息列表

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：DescribeCCCBuyInfoList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppIds.N	否	Array of Integer	应用ID列表，不传时查询所有应用 示例值：[1400000,14000001]

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	应用总数 示例值：10
SdkAppIdBuyList	Array of SdkAppIdBuyInfo	应用购买信息列表 示例值：[]
PackageBuyList	Array of PackageBuyInfo	套餐包购买信息列表 示例值：[]
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 获取用户购买信息列表示例

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribeCCCBuyInfoList
&SdkAppIds.0=1400000000
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "TotalCount": 0,
    "SdkAppIdBuyList": [
      {
        "SdkAppId": 1400000000,
        "Name": "客服中心",
        "StaffBuyNum": 0,
        "StaffBuyList": [
          {
            "Num": 0,
            "EndTime": 1623988825,
            "BuyTime": 1613988825,
            "SipNum": 0
          }
        ],
        "PhoneNumBuyList": [
          {
            "PhoneNum": "1809268888",
            "Type": 0,
            "CallType": 0,
            "EndTime": 1623988825,
            "BuyTime": 1613988825,
            "State": 0
          }
        ],
        "SipBuyNum": 0
      }
    ],
    "PackageBuyList": [
      {
        "PackageId": "d6253538",
        "Type": 0,
        "CapacitySize": 0,
        "CapacityRemain": 0,
        "EndTime": 1623988825,
        "BuyTime": 1613988825
      }
    ],
    "RequestId": "abc"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.AccountNotExist	账号不存在。
UnknownParameter	未知参数错误。

外呼相关接口

停止座席巡航式外呼任务

最近更新时间：2025-04-25 01:12:30

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

停止座席巡航式外呼任务

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：AbortAgentCruiseDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
CampaignId	是	Integer	任务 ID 示例值：2569

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 停止座席巡航式外呼任务示例

停止座席巡航式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "CampaignId": 2569
}
```

输出示例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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
InternalError	内部错误。
InvalidParameterValue	参数取值错误。

创建座席巡航式外呼任务

最近更新时间：2025-04-25 01:12:30

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

座席巡航式外呼。

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：CreateAgentCruiseDialingCampaign。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Name	是	String	任务名称 示例值：新能源SUV推荐
Agent	是	String	座席账号 示例值：foo@tencent.com
ConcurrencyNumber	是	Integer	单轮并发呼叫量 1-20 示例值：3
StartTime	是	Integer	任务启动时间，Unix 时间戳，到此时间后会自动启动任务 示例值：1708483433
EndTime	是	Integer	任务结束时间，Unix 时间戳，到此时间后会自动终止任务 示例值：1708485433
Callees.N	否	Array of String	被叫列表，支持 E.164 或不带国家码形式的号码 示例值：+8613012345678
Callers.N	否	Array of String	主叫列表，使用管理端展示的号码格式 示例值：0086075586013388
CallOrder	否	Integer	被叫呼叫顺序 0 随机 1 顺序 示例值：0
UII	否	String	调用方自定义数据，最大长度 1024 示例值：foobar

3. 输出参数

参数名称	类型	描述
CampaignId	Integer	生成的任务 ID 示例值：2569
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 创建座席巡航式外呼任务实例

创建座席巡航式外呼任务

输入示例

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

{
  "SdkAppId": 1400000000,
  "Name": "新能源SUV推荐",
  "Callees": [
    "+8613012345678",
    "+8613012345679",
    "+8613012345670"
  ],
  "Callers": [
    "0086075586013388"
  ],
  "Agent": "foo@tencent.com",
  "ConcurrencyNumber": 3,
  "CallOrder": 0,
  "StartTime": 1708483433,
  "EndTime": 1708485433
}
```

输出示例

```
{
  "Response": {
    "CampaignId": 2569,
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
LimitExceeded.OutOfCountLimit	超出数量限制。

CreateIVRSession

最近更新时间：2025-04-25 01:12:30

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建关联 IVR 的会话，仅高级版支持，目前支持呼入和自动外呼两种 IVR 类型。收到请求后 TCCC 会先尝试呼通被叫，然后进入 IVR 流程。

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：CreateIVRSession。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Callee	是	String	被叫 示例值：+8613012345678
IVRId	是	Integer	指定的 IVR Id，目前支持呼入和自动外呼两种类型 示例值：4600
Callers.N	否	Array of String	主叫号码列表 示例值：[00861,00862]
Variables.N	否	Array of Variable	自定义变量 示例值：{"name":"foo-bar"}
UUI	否	String	用户数据 示例值：foobar

3. 输出参数

参数名称	类型	描述
SessionId	String	新创建的会话 ID 示例值：adcf61b8-dbb9-4c20-87bf-c0744c04bade
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 创建关联 IVR 的会话示例

输入示例

```
POST / HTTP/1.1
Host: ccc.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: CreateIVRSession
<公共请求参数>
```

```
{
  "Callee": "008612300000000",
  "SdkAppId": 1400000000,
  "IVRId": 4500
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "SessionId": "6bb56a09278740bc80c5dc6dab783eff"
  }
}
```

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.AIConcurrentNoPackage	无可用的AI并发许可，请去控制台进行购买
FailedOperation.AIConcurrentOverFrequency	AI并发超频
FailedOperation.CallOutFailed	外呼失败。
FailedOperation.CalleelsBlackUser	高风险用户，请勿呼叫
FailedOperation.CallerOverFrequency	主叫号码外呼超频。
FailedOperation.CalloutRuleBlindArea	触发默认被叫规则，呼叫盲区

错误码	描述
FailedOperation.CalloutRuleMaxCallCountCalleeIntervalTime	触发默认外呼规则，被叫一段时间内呼叫量
FailedOperation.CalloutRuleMaxCallCountCalleePerDayAppID	触发默认外呼规则，被叫每日最大呼叫量
FailedOperation.CalloutRuleNotWorkTime	触发默认外呼规则，不在可外呼时间
FailedOperation.NoCallOutNumber	无可用的外呼号码。
FailedOperation.PermissionDenied	权限不足。
InternalError	内部错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。
LimitExceeded.BasePackageExpired	套餐包耗尽

企业资质相关接口

创建公司资质申请

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建公司资质申请（1、首次使用接口，建议先在云联络中心控制台查看各个资料模板:https://console.cloud.tencent.com/ccc/enterprise/update。2、参数中图片Url建议使用腾讯云Cos存储的临时链接）

默认接口请求频率限制：1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：CreateCompanyApply。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
CompanyInfo	是	CompanyApplyInfo	企业资质信息

3. 输出参数

参数名称	类型	描述
Id	Integer	申请单ID 示例值：1145
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 创建公司资质申请示例

输入示例

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

{
  "CompanyInfo": {
    "ApplicantType": 1,
    "CompanyName": "北京XXXXX有限责任公司",
    "BusinessId": "977111111111111",
    "BusinessIdPicUrl": "https://xxxx.com/xxxxx.jpg",
```

```
"IsEqualTencentCloud": 1,
"CorporationName": "王某某",
"CorporationId": "6108888888888888",
"CorporationIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"NetworkCommitmentPicUrl": "https://xxxx.com/xxxxx.jpg",
"CorporationHoldingOnIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"CorporationMobile": "18092688910",
"CorporationMobilePicUrl": "https://www.example.com/corporation_mobile_pic.gif",
"OperatorName": "李某某",
"OperatorId": "6100000000000000",
"OperatorIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"OperatorHoldingOnIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"OperatorMobile": "18192226857",
"OperatorEmail": "222@qq.com",
"OperatorMobilePicUrl": "https://xxxx.com/xxxxx.jpg",
"CommissionPicUrl": "https://www.example.com/commission_pic.JPG",
"CompanyAuthLetterPicUrl": "https://www.example.com/company_auth_pic.JPG",
"UseDescribeFileUrl": "https://www.example.com/use_describe_file.doc",
"AcceptPicUrl": "https://xxxxxxxxxx.com/file.jpg"
}
}
```

输出示例

```
{
  "Response": {
    "Id": 1145,
    "RequestId": "foo-bar"
  }
}
```

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
InternalError	内部错误。
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.CompanyExist	公司信息已经存在。
InvalidParameterValue.WaitingApprovalOverflow	待审核单已经超限，暂时不能再提交

查询公司资质申请列表

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询公司资质申请列表

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：DescribeCompanyList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
PageSize	是	Integer	分页尺寸，上限 100 示例值：10
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0
CompanyName.N	否	Array of String	公司名称 示例值：foobar科技有限公司
State.N	否	Array of Integer	审核状态，1-待审核，2-审核通过，3-驳回 示例值：1
ApplyID.N	否	Array of Integer	申请ID 示例值：1

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	总数 示例值：12
CompanyInfo	Array of CompanyStateInfo	企业资质审核信息
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 查询公司资质申请列表示例

输入示例

```
POST / HTTP/1.1
Host: ccc.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeCompanyList
<公共请求参数>
```

```
{
  "PageSize": 12,
  "PageNumber": 0,
  "CompanyName": [],
  "State": [],
  "ApplyID": []
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 0,
    "CompanyInfo": [
      {
        "Id": 1,
        "CompanyName": "北京XXX公司",
        "CreateTime": 164434242,
        "CheckTime": 154346666,
        "CheckMsg": "通过了",
        "State": 1,
        "BusinessId": "4324234323",
        "ModifyTime": 1602321412
      }
    ],
    "RequestId": "abc"
  }
}
```

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
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

修改公司资质申请

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

修改公司资质申请，只能修改状态为驳回或待审核的申请单。（1、首次使用接口，建议先在云联络中心控制台查看各个资料模板:https://console.cloud.tencent.com/ccc/enterprise/update。2、参数中图片Url建议使用腾讯云Cos存储的临时链接）

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：ModifyCompanyApply。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
ApplyId	是	Integer	申请单ID(只能修改状态为“驳回”或者“待审核”的申请单) 示例值：1
CompanyInfo	是	CompanyApplyInfo	企业资质信息

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 修改公司资质申请示例

输入示例

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

{
  "ApplyId": 1,
  "CompanyInfo": {
    "ApplicantType": 1,
    "CompanyName": "北京xxxxx有限责任公司",
    "BusinessId": "977111111111111",
    "BusinessIdPicUrl": "https://xxxx.com/xxxxx.jpg",
    "IsEqualTencentCloud": 1,
```

```
"CorporationName": "王某某",
"CorporationId": "6108888888888888",
"CorporationIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"NetworkCommitmentPicUrl": "https://xxxx.com/xxxxx.jpg",
"CorporationHoldingOnIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"CorporationMobile": "18092688910",
"CorporationMobilePicUrl": "https://www.example.com/corporation_mobile_pic.gif",
"OperatorName": "李某某",
"OperatorId": "6100000000000000",
"OperatorIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"OperatorHoldingOnIdPicUrl": "https://xxxx.com/xxxxx.jpg",
"OperatorMobile": "18192226857",
"OperatorEmail": "222@qq.com",
"OperatorMobilePicUrl": "https://xxxx.com/xxxxx.jpg",
"CommissionPicUrl": "https://www.example.com/commission_pic.JPG",
"CompanyAuthLetterPicUrl": "https://www.example.com/company_auth_pic.JPG",
"UseDescribeFileUrl": "https://www.example.com/use_describe_file.doc",
"AcceptPicUrl": "https://xxxxxxxxxx.com/file.jpg"
}
}
```

输出示例

```
{
  "Response": {
    "RequestId": "abc"
  }
}
```

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
InternalError	内部错误。
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.ApplyIDNotExist	审批单号不存在。
InvalidParameterValue.CompanyExist	公司信息已经存在。

语音智能体相关接口

获取 AI 耗时

最近更新时间：2025-04-25 01:12:24

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

获取 AI 时延信息

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：DescribeAllLatency。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话 ID 示例值：6bb56a09-2787-40bc-80c5-dc6dab783eff
StartTime	是	Integer	查找起始时间 示例值：1737350008
EndTime	是	Integer	1737350008 示例值：1737350008

3. 输出参数

参数名称	类型	描述
AllatencyDetail	Array of AllatencyDetail	时延明细数据 -1表示无对应数据
AllatencyStatistics	AllatencyStatistics	时延统计数据 -1表示无对应数据
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 AI 呼叫时延信息

获取 AI 呼叫时延信息

输入示例

```
POST / HTTP/1.1
Host: ccc.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: DescribeAILatency
<公共请求参数>
```

```
{
  "SdkAppId": 1400000000,
  "SessionId": "adcf61b8-dbb9-4c20-87bf-c0744c04bade",
  "StartTime": 1737350008,
  "EndTime": 1737356008
}
```

输出示例

```
{
  "Response": {
    "AILatencyStatistics": {
      "ASRLatency": {
        "MiddleLatency": 308,
        "MinLatency": 267,
        "P90Latency": 338
      },
      "ETELatency": {
        "MiddleLatency": 1327,
        "MinLatency": 1209,
        "P90Latency": 1540
      },
      "LLMLatency": {
        "MiddleLatency": 754,
        "MinLatency": 642,
        "P90Latency": 912
      },
      "TTSLatency": {
        "MiddleLatency": 267,
        "MinLatency": 146,
        "P90Latency": 366
      }
    },
    "RequestId": "ca9c3062-202b-4401-a930-3728d1ef7886"
  }
}
```

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.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameterValue.InstanceNotExist	实例不存在。

创建智能体通话

最近更新時間：2025-04-25 01:12:24

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

用于调用AI模型发起外呼通话，仅限自有电话号码使用，目前开通高级版座席**限时**免费体验。

发起通话前，请先确认您的AI模型是否兼容 OpenAI、Azure 或 Minimax 协议，并前往模型服务商网站获取相关鉴权信息。具体功能说明请参考文档 [腾讯云联络中心AI通话平台](#)。

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，完整公共参数列参见 [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：CreateAIAgentCall。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
AIAgentId	是	Integer	AI智能体ID 示例值：1
Callee	是	String	被叫号码 示例值：008613012345678
Callers.N	否	Array of String	主叫号码列表 示例值：["008601012341234", "008601012341235"]
Variables.N	否	Array of Variable	通用变量： 提示词变量 欢迎语变量 欢迎语延迟播放(秒级)：welcome-message-delay dify变量 1. dify-inputs-xxx 为dify的inputs变量 2. dify-inputs-user 为dify的用户值 3. dify-inputs-conversation_id 为dify的conversation_id值 示例值：[{"Key":"foo","Value":"bar"}]

3. 输出参数

参数名称	类型	描述
SessionId	String	新创建的会话 ID 示例值：adcf61b8-dbb9-4c20-87bf-c0744c04bade
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 使用 AI Agent 发起外呼

使用 AI Agent 发起外呼

输入示例

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

{
  "Callee": "008612300000000",
  "SdkAppId": 1400000000,
  "AIAgentId": 15,
  "Callers": [
    "008601012345678",
    "008601012345679"
  ],
  "PromptVariables": [
    {
      "Key": "foo",
      "Value": "bar"
    }
  ]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "SessionId": "6bb56a09278740bc80c5dc6dab783eff"
  }
}
```

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.CallOutFailed	外呼失败。
FailedOperation.CalleelsBlackUser	高风险用户，请勿呼叫
FailedOperation.CallerOverFrequency	主叫号码外呼超频。
FailedOperation.CalloutRuleBlindArea	触发默认被叫规则，呼叫盲区
FailedOperation.CalloutRuleMaxCallCountCalleeIntervalTime	触发默认外呼规则，被叫一段时间内呼叫量
FailedOperation.CalloutRuleMaxCallCountCalleePerDayAppID	触发默认外呼规则，被叫每日最大呼叫量
FailedOperation.CalloutRuleNotWorkTime	触发默认外呼规则，不在可外呼时间
FailedOperation.NoCallOutNumber	无可用的外呼号码。
FailedOperation.PermissionDenied	权限不足。
InternalError	内部错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。
LimitExceeded.BasePackageExpired	套餐包耗尽

创建AI通话

最近更新时间：2025-06-26 01:06:58

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

用于调用AI模型发起外呼通话，仅限自有电话号码使用，目前开通高级版座席**限时**免费体验。

发起通话前，请先确认您的AI模型是否兼容 OpenAI、Azure 或 Minimax 协议，并前往模型服务商网站获取相关鉴权信息。具体功能说明请参考文档 [腾讯云联络中心AI通话平台](#)。

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：CreateAICall。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Callee	是	String	被叫号码 示例值：008613012345678
LLMType	是	String	模型接口协议类型，目前兼容四种协议类型： – OpenAI协议(包括GPT、混元、DeepSeek等)："openai" – Azure协议："azure" – Minimax协议："minimax" – Dify协议："dify" 示例值：openai
APIKey	是	String	模型API密钥，获取鉴权信息方式请参见各模型官网 – OpenAI协议： GPT ， 混元 ， DeepSeek ； – Azure协议： Azure GPT ； – Minimax： Minimax 示例值：sk-proj-xxxx
APIUrl	是	String	模型接口地址 – OpenAI协议 GPT："https://api.openai.com/v1/" 混元："https://api.hunyuan.cloud.tencent.com/v1" Deepseek："https://api.deepseek.com/v1" – Azure协议 "https://{your-resource-name}.openai.azure.com?api-version={api-version}" – Minimax协议

参数名称	必选	类型	描述
			"https://api.minimax.chat/v1" 示例值: https://api.openai.com/v1/
SystemPrompt	否	String	用于设定AI人设、说话规则、任务等的全局提示词。示例: ## 人设您是人民医院友善、和蔼的随访医生李医生, 正在给患者小明的家长打电话, 原因是医院要求小明2024-08-08回院复查手术恢复情况, 但小明没有来。您需要按照任务流程对小明家长进行电话随访调查。## 要求简洁回复: 使用简练语言, 每次最多询问一个问题, 不要在一个回复中询问多个问题。富有变化: 尽量使表达富有变化, 表达机械重复。自然亲切: 使用日常语言, 尽量显得专业并亲切。提到时间时使用口语表述, 如下周三、6月18日。积极主动: 尝试引导对话, 每个回复通常以问题或下一步建议来结尾。询问清楚: 如果对方部分回答了您的问题, 或者回答很模糊, 请通过追问来确保回答的完整明确。遵循任务: 当对方的回答偏离了您的任务时, 及时引导对方回到任务中。不要从头开始重复, 从偏离的地方继续询问。诚实可靠: 对于客户的提问, 如果不确定请务必不要编造, 礼貌告知对方不清楚。不要捏造患者未提及的症状史、用药史、治疗史。其他注意点: 避免提到病情恶化、恢复不理想或疾病名称等使用会使患者感到紧张的表述。不要问患者已经直接或间接回答过的问题, 例如患者已经说没有不适症状, 那就不要再问手术部位是否有红肿疼痛症状的问题。##任务: 1.自我介绍您是人民医院负责随访的李医生, 并说明致电的目的。2.询问被叫方是否是小明家长。- 如果不是小明家长, 请礼貌表达歉意, 并使用 call_end 挂断电话。- 如果小明家长没空, 请礼貌告诉对方稍后会重新致电, 并使用 end_call 挂断电话。3.询问小明出院后水肿情况如何, 较出院时是否有变化。- 如果水肿变严重, 直接跳转步骤7。4.询问出院后是否给小朋友量过体温, 是否出现过发烧情况。- 如果没有量过体温, 请礼貌告诉家长出院后三个月内需要每天观察体温。- 如果出现过发烧, 请直接跳转步骤7。5.询问出院后是否给小朋友按时服药。- 如果没有按时服药, 请友善提醒家长严格按医嘱服用药物, 避免影响手术效果。6.询问小朋友在饮食上是否做到低盐低脂, 适量吃优质蛋白如鸡蛋、牛奶、瘦肉等。- 如果没有做到, 请友善提醒家长低盐低脂和优质蛋白有助小朋友尽快恢复。7.告知家长医生要求6月18日回院复查, 但没看到有相关复诊记录。提醒家长尽快前往医院体检复查血常规、尿常规。8.询问家长是否有问题需要咨询, 如果没有请礼貌道别并用call_end挂断电话。 示例值: 示例值请看上方描述
Model	否	String	模型名称, 如 - OpenAI协议 "gpt-4o-mini", "gpt-4o", "hunyuan-standard", "hunyuan-turbo", "deepseek-chat"; - Azure协议 "gpt-4o-mini", "gpt-4o"; - Minimax协议 "deepseek-chat". 示例值: gpt-4o-mini
VoiceType	否	String	默认提供以下音色参数值可选择, 如需自定义音色VoiceType请留空并在参数CustomTTSTConfig中配置 汉语: ZhiMei: 智美, 客服女声 ZhiXi: 智希 通用女声 ZhiQi: 智琪 客服女声 ZhiTian: 智甜 女童声 AiXiaoJing: 爱小静 对话女声 英语: WeRose: 英文女声 Monika: 英文女声 日语: Nanami 韩语: SunHi 印度尼西亚语(印度尼西亚): Gadis

参数名称	必选	类型	描述
			马来语（马来西亚）： Yasmin 泰米尔语（马来西亚）： Kani 泰语（泰国）： Achara 越南语(越南): HoaiMy 示例值: ZhiQi
Callers.N	否	Array of String	主叫号码列表 示例值: [008613912341234, 008613912341235]
WelcomeMessage	否	String	用于设定AI座席欢迎语。 示例值: 您好
WelcomeType	否	Integer	0: 使用welcomeMessage(为空时, 被叫先说话; 不为空时, 机器人先说话) 1: 使用ai根据prompt自动生成welcomeMessage并先说话 示例值: 0
WelcomeMessagePriority	否	Integer	0: 默认可打断, 2: 高优先不可打断 示例值: 0
MaxDuration	否	Integer	最大等待时长(毫秒), 默认60秒, 超过这个时间用户没说话, 自动挂断 示例值: 60000
Languages.N	否	Array of String	语音识别支持的语言, 默认是"zh" 中文, 填写数组,最长4个语言, 第一个语言为主要识别语言, 后面为可选语言, 注意:主要语言为中国方言时, 可选语言无效 目前全量支持的语言如下, 等号左面是语言英文名, 右面是Language字段需要填写的值, 该值遵循ISO639: 1. Chinese = "zh" # 中文 2. Chinese_TW = "zh-TW" # 中国台湾 3. Chinese_DIALECT = "zh-dialect" # 中国方言 4. English = "en" # 英语 5. Vietnamese = "vi" # 越南语 6. Japanese = "ja" # 日语 7. Korean = "ko" # 韩语 8. Indonesia = "id" # 印度尼西亚语 9. Thai = "th" # 泰语 10. Portuguese = "pt" # 葡萄牙语 11. Turkish = "tr" # 土耳其语 12. Arabic = "ar" # 阿拉伯语 13. Spanish = "es" # 西班牙语 14. Hindi = "hi" # 印地语 15. French = "fr" # 法语 16. Malay = "ms" # 马来语 17. Filipino = "fil" # 菲律宾语 18. German = "de" # 德语 19. Italian = "it" # 意大利语 20. Russian = "ru" # 俄语 示例值: ["zh", "en"]
InterruptMode	否	Integer	打断AI说话模式, 默认为0, 0表示自动打断, 1表示不打断。 示例值: 1
InterruptSpeechDuration	否	Integer	InterruptMode为0时使用, 单位为毫秒, 默认为500ms。表示服务端检测到持续InterruptSpeechDuration毫秒的人声则进行打断。 示例值: 500
EndFunctionEnable	否	Boolean	模型是否支持(或者开启)call_end function calling 示例值: true

参数名称	必选	类型	描述
EndFunctionDesc	否	String	EndFunctionEnable为true时生效；call_end function calling的desc，默认为"End the call when user has to leave (like says bye) or you are instructed to do so." 示例值：在用户不想继续通话或者展现出不开心的情绪时挂断通话
TransferFunctionEnable	否	Boolean	模型是否支持(或者开启)transfer_to_human function calling 示例值：true
TransferItems.N	否	Array of AITransferItem	TransferFunctionEnable为true的时候生效：转人工配置
NotifyDuration	否	Integer	用户多久没说话提示时长,最小10秒,默认10秒 示例值：10000
NotifyMessage	否	String	用户NotifyDuration没说话，AI提示的语句，默认是"抱歉，我没听清。您可以重复下吗？" 示例值：抱歉，我没听清。您可以重复下吗？
NotifyMaxCount	否	Integer	最大触发AI提示音次数，默认为 unlimited 示例值：1
CustomTTSTConfig	否	String	和VoiceType字段需要选填一个，这里是使用自己自定义的TTS，VoiceType是系统内置的一些音色 • Tencent TTS 配置请参考腾讯云TTS文档链接 <pre>{ "TTSType": "tencent", // String TTS类型，目前支持"tencent" 和 "minimax"，其他的厂商支持中 "AppId": "您的应用ID", // String 必填 "SecretId": "您的密钥ID", // String 必填 "SecretKey": "您的密钥Key", // String 必填 "VoiceType": 101001, // Integer 必填，音色 ID，包括标准音色 与精品音色，精品音色拟真度更高，价格不同于标准音色，请参见语音合成计 费概述。完整的音色 ID 列表请参见语音合成音色列表。 "Speed": 1.25, // Integer 非必填，语速，范围：[-2, 6]，分别 对应不同语速： -2：代表0.6倍 -1：代表0.8倍 0：代表1.0倍（默认） 1：代表1.2倍 2：代表1.5倍 6：代表2.5倍 如果需要更细化的语速，可以 保留小数点后 2 位，例如0.5/1.25/2.81等。 参数值与实际语速转换，可 参考 语速转换 "Volume": 5, // Integer 非必填，音量大小，范围：[0, 10]，分别 对应11个等级的音量，默认值为0，代表正常音量。 "PrimaryLanguage": 1, // Integer 可选 主要语言 1-中文（默 认） 2-英文 3-日文 "FastVoiceType": "xxxx" // 可选参数， 快速声音复刻的参数 }</pre> • Minimax TTS 配置请参考Minimax TTS文档链接。注意Minimax TTS存在频率限制，超频可能会导致回答卡顿，Minimax TTS频率限制相关文档链接。

参数名称	必选	类型	描述
			<pre>{ "TTSType": "minimax", // String TTS类型, "Model": "speech-01-turbo", "APIUrl": "https://api.minimax.chat/v1/t2a_v2", "APIKey": "eyxxxx", "GroupId": "181000000000000", "VoiceType": "female-tianmei-jingpin", "Speed": 1.2 }</pre> - 火山 TTS 配置音色类型参考火山TTS文档链接 语音合成音色列表-语音技术-火山引擎 大模型语音合成音色列表-语音技术-火山引擎 <pre>{ "TTSType": "volcengine", // 必填: String TTS类型 "AppId" : "xxxxxxx", // 必填: String 火山引擎分配的Appid "Token" : "TY9d4sQXHxxxxxxx", // 必填: String类型 火山引擎的访问token "Speed" : 1.0, // 可选参数 语速, 默认为1.0 "Volume": 1.0, // 可选参数, 音量大小, 默认为1.0 "Cluster" : "volcano_tts", // 可选参数, 业务集群, 默认是 volcano_tts "VoiceType" : "zh_male_aojiaobazong_moon_big tts" // 音色类型, 默认为大模型语音合成的音色。 如果使用普通语音合成, 则需要填写对应的音色类型。 音色类型填写错误会导致没有声音。 }</pre> - Azure TTS 配置请参考AzureTTS文档链接 <pre>{ "TTSType": "azure", // 必填: String TTS类型 "SubscriptionKey": "xxxxxxx", // 必填: String 订阅的Key "Region": "chinanorth3", // 必填: String 订阅的地区 "VoiceName": "zh-CN-XiaoxiaoNeural", // 必填: String 音色名 必填 "Language": "zh-CN", // 必填: String 合成的语言 "Rate": 1 // 选填: float 语速 0.5~2 默认为 1 }</pre>

参数名称	必选	类型	描述
			自定义 TTS 具体协议规范请参考腾讯文档 <pre>{ "TTSType": "custom", // String 必填 "APIKey": "ApiKey", // String 必填 用来鉴权 "APIUrl": "http://0.0.0.0:8080/stream-audio" // String, 必填, TTS API URL "AudioFormat": "wav", // String, 非必填, 期望输出的音频格式, 如mp3, ogg_opus, pcm, wav, 默认为 wav, 目前只支持pcm和wav, "SampleRate": 16000, // Integer, 非必填, 音频采样率, 默认为16000 (16k), 推荐值为16000 "AudioChannel": 1, // Integer, 非必填, 音频通道数, 取值: 1 或 2 默认为1 }</pre> 示例值: { "TTSType": "azure", "SubscriptionKey": "xx", "Region": "chinanorth3", "VoiceName": "zh-CN-", "Language": "zh-CN", "Rate": 1 }
VadSilenceTime	否	Integer	语音识别vad的时间, 范围为240-2000, 默认为1000, 单位为ms。更小的值会让语音识别分句更快。 示例值: 500
ExtractConfig.N	否	Array of AICallExtractConfigElement	通话内容提取配置 示例值: [{"InfoType": "Boolean", "InfoName": "兴趣", "InfoContent": "用户是否对产品感兴趣"}]
Temperature	否	Float	模型温度控制 示例值: 0
Variables.N	否	Array of Variable	通用变量: 提示词变量 欢迎语变量 欢迎语延迟播放(秒级): welcome-message-delay dify变量 1. dify-inputs-xxx 为dify的inputs变量 2. dify-inputs-user 为dify的用户值 3. dify-inputs-conversation_id 为dify的conversation_id值 示例值: [{"Key": "foo", "Value": "bar"}]

3. 输出参数

参数名称	类型	描述
SessionId	String	新创建的会话 ID 示例值: adcf61b8-dbb9-4c20-87bf-c0744c04bade

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 创建AI通话示例（患者随访场景）

本示例模拟某医院调用大模型对手术出院患者家长进行电话随访调查。

输入示例

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

{
  "Callee": "008612300000000",
  "SdkAppId": 1400000000,
  "SystemPrompt": "你是人民医院友善、和蔼的随访医生李医生，正在给患者小明的家长打电话。",
  "WelcomeMessage": "您好",
  "LLMType": "openai",
  "Model": "gpt-4o-mini",
  "APIKey": "sk-proj-xxxxx",
  "APIUrl": "https://api.openai.com/v1/",
  "VoiceType": "ZhiXi",
  "EndFunctionEnable": true
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "SessionId": "6bb56a09278740bc80c5dc6dab783eff"
  }
}
```

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.CallOutFailed	外呼失败。
FailedOperation.CalleelsBlackUser	高风险用户，请勿呼叫
FailedOperation.CallerOverFrequency	主叫号码外呼超频。
FailedOperation.CalloutRuleBlindArea	触发默认被叫规则，呼叫盲区
FailedOperation.CalloutRuleMaxCallCountCalleeIntervalTime	触发默认外呼规则，被叫一段时间内呼叫量
FailedOperation.CalloutRuleMaxCallCountCalleePerDayAppID	触发默认外呼规则，被叫每日最大呼叫量
FailedOperation.CalloutRuleNotWorkTime	触发默认外呼规则，不在可外呼时间
FailedOperation.NoCallOutNumber	无可用的外呼号码。
FailedOperation.PermissionDenied	权限不足。
InternalError	内部错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。
LimitExceeded.BasePackageExpired	套餐包耗尽

查询 AI 通话内容提取结果

最近更新时间：2025-04-25 01:12:24

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

获取 AI 通话内容提取结果。

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：DescribeAICallExtractResult。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话 ID 示例值：adcf61b8-dbb9-4c20-87bf-c0744c04bade
StartTime	是	Integer	查找起始时间 示例值：1737350008
EndTime	是	Integer	查找结束时间 示例值：1737356008

3. 输出参数

参数名称	类型	描述
ResultList	Array of AICallExtractResultElement	结果列表 示例值：[[{"InfoType":"Boolean","InfoName":"兴趣","InfoContent":"用户是否对产品感兴趣","Result":{"Boolean":true}}]]
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 AI 通话内容提取结果示例

查询 AI 通话内容提取结果示例

输入示例

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

```
{
  "SdkAppId": 1400000000,
  "SessionId": "adcf61b8-dbb9-4c20-87bf-c0744c04bade",
  "StartTime": 1737350008,
  "EndTime": 1737356008
}
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "ResultList": [
      {
        "InfoType": "Boolean",
        "InfoName": "兴趣",
        "InfoContent": "用户是否对产品感兴趣",
        "Result": {
          "Text": "",
          "Chosen": [],
          "Boolean": true,
          "Number": 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
InternalServerError	内部错误。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

AI服务相关接口

控制AI对话

最近更新时间：2025-07-04 01:12:40

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

提供服务端控制机器人的功能

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：ControlAIConversation。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SessionId	是	String	会话 ID
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Command	否	String	控制命令，目前支持命令如下： - ServerPushText，服务端发送文本给AI机器人，AI机器人会播报该文本
ServerPushText	否	ServerPushText	服务端发送播报文本命令，当Command为ServerPushText时必须填

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 发送播报文本

当您想让机器人主动播报文本的时候，可以使用该接口

输入示例

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

{
  "SdkAppId": "1400000000",
  "SessionId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
```

```
"Command": "ServerPushText",
"ServerPushText": {
  "Text": "你好很高兴为您服务",
  "Interrupt": true
}
```

输出示例

```
{
  "Response": {
    "RequestId": "xxx-xxx"
  }
}
```

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.ControlAIConversation	ControlAIConversation错误信息
FailedOperation.SessionNotExists	会话不存在
FailedOperation.SessionNotInControlState	会话不在可控制状态
InvalidParameterValue.InstanceNotExist	实例不存在。

其他接口

创建用户数据签名

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：`ccc.tencentcloudapi.com`。

创建用户数据签名

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：CreateUserSig。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
Uid	是	String	用户 ID，该值必须与 ClientData 字段中 Uid 的值一致 示例值：foo
ExpiredTime	是	Integer	有效期，单位秒，不超过 1 小时 示例值：3600
ClientData	否	String	用户签名数据，必填字段，为标准 JSON 格式 示例值：参考下文示例

3. 输出参数

参数名称	类型	描述
UserSig	String	签名结果 示例值：eJw0jkFPwjAYQP-Ld9Wwfm2HtokHswHk9lBMBtw22uEHbpa1JRjzfZ5vW95OX9
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 生成用户数据签名

输入示例

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



```
{
  "SdkAppId": 1400000000,
  "Uid": "staff1@xxx.com",
  "ClientData": "{\\"peerSource\\":\\"腾讯科技\\",\\"nickName\\":\\"企鹅一号\\",\\"avatarUrl\\":\\"https://foo.tencent.com/qieyihao.jpeg\\",\\"uid\\":1846526}",
  "ExpiredTime": 3600
}
```

输出示例

```
{
  "Response": {
    "RequestId": "48edd236-7ef1-45af-9e12-fc376ba355bf",
    "UserSig": "eJw0jkFPwjAYQP-Ld9Wwfm2HtokHswHk9IBMBtw22uEHbpa1JRrjfzfZ5vW95OX9QP68mtgvr50FrYRSQonrHpKxbaCabAcaEFGmMAhvTqVzZECjZIxPJUc5GjqAho1fJm1pnlfb9iYJx2WZZ*YqnLO1m9Wrh-uXed48ZR9vkYr13ZgM1FjQOOXyNkXJ2UCjt10Va9CwLRbH3Xzd7Pi7K5vgqyK97B9n33vxiqY10m4*-10X-pdPGPz*BQAA--9b7kId"
  }
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter	参数错误。
InvalidParameterValue.InstanceNotExist	实例不存在。

挂断电话

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

挂断电话

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：HangUpCall。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话ID 示例值：1001

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 挂断电话

输入示例

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

{
  "SdkAppId": 1400000000,
  "SessionId": "9dc79b2c-9902-419c-b963-16bee762ae3a"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127"
  }
}
```

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	内部数据库访问失败。

拉取会话录音转文本信息

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

拉取会话录音转文本信息

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：DescribeTelRecordAsr。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话 ID 示例值：6bb56a09-2787-40bc-80c5-dc6dab783eff

3. 输出参数

参数名称	类型	描述
AsrDataList	Array of AsrData	录音转文本信息
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 拉取录音信息

输入示例

```
https://ccc.tencentcloudapi.com/?Action=DescribeTelRecordAsr
&SdkAppId=1400000000
&SessionId=6bb56a09-2787-40bc-80c5-dc6dab783eff
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "6bb56a09-2787-40bc-80c5-dc6dab783eff",
    "AsrDataList": [
```

```
{
  "User": "foo@tencent.com",
  "Message": "测试会话",
  "Start": 1729697242236,
  "End": 1729697245426
}
```

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。
InternalServerError.DBError	内部数据库访问失败。
InvalidParameter.InstanceNotExist	实例不存在。
MissingParameter	缺少参数错误。
ResourceNotFound	资源不存在。

创建管理端访问链接

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

创建管理端访问链接

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：CreateAdminURL。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SeatUserId	是	String	管理员账号 示例值：foo@tencent.com

3. 输出参数

参数名称	类型	描述
URL	String	登录链接 示例值： https://tccc.qcloud.com/saas-manage/#/1400000000/home?token=6bb56a09-2787-40bc-80c5-dc6dab783eff
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 创建管理端登录链接示例

创建管理端登录链接示例

输入示例

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

{
  "SdkAppId": 1400000000,
```

```
"SeatUserId": "foo@tencent.com"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "48edd236-7ef1-45af-9e12-fc376ba355bf",
    "URL": "https://tccc.qcloud.com/saas-manage/#/1400000000/home?token=6bb56a09-2787-40bc-80c5-dc6dab783eff"
  }
}
```

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](#)。

查询IVR音频文件列表

最近更新时间：2025-04-25 01:12:33

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

查询IVR音频文件列表信息

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：DescribeIvrAudioList。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1600000001
PageSize	是	Integer	分页尺寸，上限 50 示例值：10
PageNumber	是	Integer	分页页码，从 0 开始 示例值：0
CustomFileName.N	否	Array of String	文件别名 示例值：欢迎
AudioFileName.N	否	Array of String	文件名 示例值：aaa.mp3
FileId.N	否	Array of Integer	文件ID 示例值：12

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	总数 示例值：0
FileInfo	Array of AudioFileInfo	文件信息
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 查询IVR音频文件列表示例

输入示例


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

{
  "SdkAppId": 1600000000,
  "CustomFileName": [
    "测试文件名"
  ],
  "AudioFileName": [],
  "FileId": [],
  "PageSize": 1,
  "PageNumber": 0
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "FileInfo": [
      {
        "FileId": 1,
        "CustomFileName": "测试文件名",
        "AudioFileName": "a.mp3",
        "Status": 0
      }
    ],
    "RequestId": "abc"
  }
}
```

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
InternalError	内部错误。
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

上传IVR音频文件

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

上传IVR中使用的音频文件，每日上传文件限制50个。（参数中音频文件Url建议使用腾讯云Cos存储的临时链接）

默认接口请求频率限制：1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：UploadIvrAudio。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1600000001
AudioList.N	是	Array of UploadAudioInfo	音频文件列表

3. 输出参数

参数名称	类型	描述
FailedFileList	Array of UploadIvrAudioFailedInfo	上传失败的文件列表
SuccessFileList	Array of AudioFileInfo	上传成功文件列表
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 上传IVR音频文件

输入示例

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

{
  "SdkAppId": 140000000,
  "AudioList": [
    {
      "CustomFileName": "测试文件名",
```

```
"AudioUrl": "http://xxxx.com/a.mp3"
}
]
}
```

输出示例

```
{
  "Response": {
    "RequestId": "a",
    "FailedFileList": [],
    "SuccessFileList": [
      {
        "AudioFileName": "business_id_pic5.mp3",
        "CustomFileName": "1231",
        "FileId": 50,
        "Status": 0
      },
      {
        "AudioFileName": "business_id_pic6.mp3",
        "CustomFileName": "4561",
        "FileId": 51,
        "Status": 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
FailedOperation.UploadFileOverflow	上传文件个数超过限制
InternalError.DBError	内部数据库访问失败。
InvalidParameterValue	参数取值错误。

转接人工

最近更新时间：2025-04-25 01:12:32

1. 接口描述

接口请求域名：ccc.tencentcloudapi.com。

特定场景下讲会话转接到人工坐席

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：TransferToManual。
Version	是	String	公共参数 ，本接口取值：2020-02-10。
Region	否	String	公共参数 ，此参数为可选参数。
SdkAppId	是	Integer	应用 ID（必填），可以查看 https://console.cloud.tencent.com/ccc 示例值：1400000000
SessionId	是	String	会话ID 示例值：d6253538-7c28-477b-998a-919059801899
SkillGroupId	是	Integer	技能组Id 示例值：5002

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 转人工

输入示例

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

{
  "SdkAppId": 1400000000,
  "SessionId": "adcf61b8-dbb9-4c20-87bf-c0744c04bade",
  "SkillGroupId": 5002
}
```

输出示例

```
{
  "Response": {
    "RequestId": "86a17f0e-bcb3-46bf-ac66-8f165fe52127"
  }
}
```

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.SessionNotExists	会话不存在
InternalServerError	内部错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InstanceNotExist	实例不存在。
InvalidParameterValue.RingAllGroupMemberOverflow	同振技能组人数超上限2，请确认并修改。
InvalidParameterValue.SkillGroupError	技能组错误。

数据结构

最近更新时间：2025-07-07 01:11:49

AICallExtractConfigElement

AI 通话提取配置项

被如下接口引用：CreateAICall。

名称	类型	必选	描述
InfoType	String	是	配置项类型，包括 Text 文本 Selector 选项 Boolean 布尔值 Number 数字 示例值：Number
InfoName	String	是	配置项名称，不可重复 示例值：姓名
InfoContent	String	否	配置项具体内容 示例值：客户姓名
Examples	Array of String	否	配置项提取内容示例 示例值：["张三"]
Choices	Array of String	否	InfoType 为 Selector，需要配置此字段 示例值：["动力","内饰","价格"]

AICallExtractResultElement

AI 通话提取结果。

被如下接口引用：DescribeAICallExtractResult。

名称	类型	描述
InfoType	String	提取信息的类型 Text 文本 Selector 选项 Boolean 布尔值 Number 数字 示例值：Number
InfoName	String	提取信息的名称 示例值：驾龄
InfoContent	String	提取信息的具体描述 示例值：客户具体驾龄是多少年
Result	AICallExtractResultInfo	提取信息的具体结果 示例值：{"Boolean":true}

AICallExtractResultInfo

AI 通话结果具体信息

被如下接口引用：DescribeAICallExtractResult。

名称	类型	描述
Text	String	提取的类型是文本 示例值：xx省xx市xx区xx街道
Chosen	Array of String	提取的内型是选项 示例值：["动力","价格"]

名称	类型	描述
Boolean	Boolean	提取类型是布尔值 示例值：true
Number	Float	提取类型是数字 示例值：7.0

AILatencyDetail

AI时延明细

被如下接口引用：DescribeAILatency。

名称	类型	描述
RoundId	String	对话ID 示例值：0c489e76-41a7-4390-aaaf-4eb26a7c9987
ASRLatency	Integer	asr时延（毫秒） 示例值：100
TTSLatency	Integer	tts时延（毫秒） 示例值：100
LLMLatency	Integer	llm时延（毫秒） 示例值：100
LLMFirstTokenLatency	Integer	llm首token时延(毫秒) 示例值：100
ETELatency	Integer	端到端时延（毫秒） 示例值：100

AILatencyStatistics

AI时延统计

被如下接口引用：DescribeAILatency。

名称	类型	描述
ASRLatency	AILatencyStatisticsInfo	asr时延统计
TTSLatency	AILatencyStatisticsInfo	tts时延统计
LLMLatency	AILatencyStatisticsInfo	llm时延统计
ETELatency	AILatencyStatisticsInfo	端到端时延统计

AILatencyStatisticsInfo

AI时延统计

被如下接口引用：DescribeAILatency。

名称	类型	描述
MinLatency	Integer	最小值 示例值：100
MiddleLatency	Integer	中位数 示例值：100
P90Latency	Integer	p90 示例值：100

AITransferItem

AI转人工配置项

被如下接口引用：CreateAICall。

名称	类型	必选	描述
TransferFunctionName	String	是	转人工的function calling 名称 示例值：transfer_to_human
TransferFunctionDesc	String	是	TransferFunctionEnable为true时生效；transfer_to_human function calling的desc，默认为 "Transfer to human when the user has to transfer to human (like says transfer to human) or you are instructed to do so." 示例值：当客户表达出需要人工服务，或者机器人无法处理问题时，转人工
TransferSkillGroupId	Integer	是	转人工的技能组ID 示例值：1

ActiveCarrierPrivilegeNumber

生效运营商白名单号码

被如下接口引用：DescribeActiveCarrierPrivilegeNumber。

名称	类型	描述
SdkAppId	Integer	实例Id 示例值：1400000000
Caller	String	主叫号码 示例值：008601012345678
Callee	String	被叫号码 示例值：008601012345678
CreateTime	Integer	生效unix时间戳(秒) 示例值：1668590448

AsrData

语音转文本信息

被如下接口引用：DescribeTelRecordAsr。

名称	类型	描述
User	String	用户方 示例值：xxx@xxx.com
Message	String	消息内容 示例值：hello world
Start	Integer	句子开始时间，Unix 毫秒时间戳 示例值：1729697242236
End	Integer	句子结束时间，Unix 毫秒时间戳 示例值：1729697245426

AudioFileInfo

音频文件审核信息

被如下接口引用：DescribeIvrAudioList, UploadIvrAudio。

名称	类型	描述
FileId	Integer	文件ID 示例值：1

名称	类型	描述
CustomFileName	String	文件别名 示例值：欢迎
AudioFileName	String	文件名 示例值：xxx.mp3
Status	Integer	审核状态，0-未审核，1-审核通过，2-审核拒绝 示例值：1

AutoCalloutTaskCalleeInfo

外呼任务被叫信息

被如下接口引用：DescribeAutoCalloutTask。

名称	类型	描述
Callee	String	被叫号码 示例值："008613912345678"
State	Integer	呼叫状态 0初始 1已接听 2未接听 3呼叫中 4待重试 示例值：1
Sessions	Array of String	会话ID列表 示例值：["aaaa-xxxx"]

AutoCalloutTaskInfo

自动外呼任务列表项

被如下接口引用：DescribeAutoCalloutTasks。

名称	类型	描述
Name	String	任务名 示例值：任务1
CalleeCount	Integer	被叫数量 示例值：5
Callers	Array of String	主叫号码列表 示例值：["008601012345678"]
NotBefore	Integer	起始时间戳 示例值：1145
NotAfter	Integer	结束时间戳 注意：此字段可能返回 null，表示取不到有效值。 示例值：1146
IvrlId	Integer	任务使用的IvrlId 示例值：5
State	Integer	任务状态： 0初始：任务创建，呼叫未开始 1运行中 2 已完成：任务中所有呼叫完成 3结束中：任务到期，但仍有部分呼叫未结束 4已结束：任务到期终止 示例值：1
TaskId	Integer	任务Id 示例值：341431

CallInMetrics

呼入实时指标

被如下接口引用：DescribeCallInMetrics。

名称	类型	描述
IvrCount	Integer	IVR驻留数量 示例值：0
QueueCount	Integer	排队中数量 示例值：0
RingCount	Integer	振铃中数量 示例值：0
AcceptCount	Integer	接通中数量 示例值：0
TransferOuterCount	Integer	客服转接外线中数量 示例值：0
MaxQueueDuration	Integer	最大排队时长 示例值：0
AvgQueueDuration	Integer	平均排队时长 示例值：0
MaxRingDuration	Integer	最大振铃时长 示例值：0
AvgRingDuration	Integer	平均振铃时长 示例值：0
MaxAcceptDuration	Integer	最大接通时长 示例值：0
AvgAcceptDuration	Integer	平均接通时长 示例值：0

CallInNumberMetrics

呼入线路维度相关指标

被如下接口引用：DescribeCallInMetrics。

名称	类型	描述
Number	String	线路号码 示例值：02188888888
Metrics	CallInMetrics	线路相关指标 示例值：{"IvrCount":0,"QueueCount":0,"RingCount":0,"AcceptCount":0,"TransferOuterCount":0,"MaxQueueDuration":0,"AvgQueueDuration":0,"MaxRingDuration":0,"AvgRingDuration":0,"MaxAcceptDuration":0,"AvgAcceptDuration":0}
SkillGroupMetrics	Array of CallInSkillGroupMetrics	所属技能组相关指标 示例值：[{"SkillGroupId":0,"Metrics":{"IvrCount":0,"QueueCount":0,"RingCount":0,"AcceptCount":0,"TransferOuterCount":0,"MaxQueueDuration":0,"AvgQueueDuration":0,"MaxRingDuration":0,"AvgRingDuration":0,"MaxAcceptDuration":0,"AvgAcceptDuration":0}}

CallInSkillGroupMetrics

呼入技能组相关指标

被如下接口引用：DescribeCallInMetrics。

名称	类型	描述
SkillGroupId	Integer	技能组ID 示例值：1
Metrics	CallInMetrics	数据指标

名称	类型	描述
Name	String	技能组名称 示例值：技能组名称

CalleeAttribute

被叫属性

被如下接口引用：CreateAutoCalloutTask, CreatePredictiveDialingCampaign, UpdatePredictiveDialingCampaign。

名称	类型	必选	描述
Callee	String	是	被叫号码 示例值：008613012345678
UUI	String	否	随路数据 示例值：foobar
Variables	Array of Variable	否	参数 示例值：{"foo":"bar"}

CarrierPrivilegeNumberApplicant

运营商白名单号码申请单

被如下接口引用：DescribeCarrierPrivilegeNumberApplicants。

名称	类型	描述
SdkAppId	Integer	实例Id 示例值：1400000000
ApplicantId	Integer	申请单Id 示例值：1145
Callers	Array of String	主叫号码列表 示例值：["008601012345678"]
Callees	Array of String	被叫号码列表 示例值：["008601012345678"]
Description	String	描述 示例值：申请描述
State	Integer	审批状态:1 待审核、2 通过、3 拒绝 示例值：1
CreateTime	Integer	创建时间，unix时间戳(秒) 示例值：1668573686
UpdateTime	Integer	更新时间，unix时间戳(秒) 示例值：1668573686

ClientInfo

座席登录的终端信息

被如下接口引用：DescribeStaffStatusMetrics。

名称	类型	描述
ClientType	String	登录的端类型，"Web"表示web工作台，"WeChatMiniProgram"表示微信小程序 示例值：Web
IsConnected	Boolean	当前登录的端是否在前台。若登录的端是Web，则该值为true；若登录的端是WeChatMiniProgram，true表示打开着微信小程序，false表示微信小程序退到后台 示例值：true

CompanyApplyInfo

企业资质申请信息

被如下接口引用：CreateCompanyApply, ModifyCompanyApply。

名称	类型	必选	描述
ApplicantType	Integer	是	申请人身份，0-公司法定代表人，1-经办人（受法定代表人委托） 示例值：1
CompanyName	String	是	企业名称 示例值：北京XXCC有限责任公司
BusinessId	String	是	统一社会信用代码 示例值：914111111111111111
BusinessIdPicUrl	String	是	营业执照扫描件(加盖公章)。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
CorporationName	String	是	法定代表人名称 示例值：王某某
CorporationId	String	是	法定代表人身份证号码 示例值：6100000000000008
CorporationIdPicUrl	String	是	法定代表人身份证正反面扫描件。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
NetworkCommitmentPicUrl	String	是	安全合规使用承诺书。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
IsEqualTencentCloud	Integer	是	是否与腾讯云账号的资质一致,0-不一致,1-一致 示例值：1
CorporationMobile	String	是	法定代表人手机号 示例值：18011111111
CorporationMobilePicUrl	String	是	法定代表人手机号码实名认证。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
UseDescribeFileUrl	String	是	通话话术。(支持doc、docx格式的文档不超过50MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
CompanyAuthLetterPicUrl	String	是	公司授权函。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
AcceptPicUrl	String	是	电话受理单。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
CorporationHoldingOnIdPicUrl	String	否	法定代表人手持身份证照，申请人类型为法定代表人时必须填。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg
OperatorName	String	否	经办人名称，申请人类型为经办人时必须填。 示例值：李某某
OperatorId	String	否	经办人证件号码，申请人类型为经办人时必须填。 示例值：6101111111111111
OperatorIdPicUrl	String	否	经办人身份证正反面扫描件，申请人类型为经办人时必须填。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值：https://xxxx.com/xxxxx.jpg

名称	类型	必选	描述
OperatorHoldingOnIdPicUrl	String	否	经办人手持身份证照，申请人类型为经办人时必填。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值: https://xxx.com/xxxxx.jpg
CommissionPicUrl	String	否	委托授权书，申请人类型为经办人时必填。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值: https://xxx.com/xxxxx.jpg
OperatorMobile	String	否	经办人手机号，申请人类型为经办人时必填。 示例值: 18022222222
OperatorEmail	String	否	经办人邮箱，申请人类型为经办人时必填。 示例值: 111@qq.com
OperatorMobilePicUrl	String	否	经办人手机号码实名认证，申请人类型为经办人时必填。(支持jpg、png、gif、jpeg格式的图片，每张图片应大于50K，不超过5MB，模板参见控制台:https://console.cloud.tencent.com/ccs/enterprise/update) 示例值: https://xxx.com/xxxxx.jpg

CompanyStateInfo

公司资质审核状态信息

被如下接口引用：DescribeCompanyList。

名称	类型	必选	描述
Id	Integer	否	申请单ID 示例值: 1
CompanyName	String	否	公司名称 示例值: 北京XX公司
CreateTime	Integer	否	创建时间 示例值: 1653534444
CheckTime	Integer	否	审核时间 示例值: 1653534444
CheckMsg	String	否	审核备注 示例值: 通过了
State	Integer	否	审核状态，1-待审核，2-审核通过，3-驳回 示例值: 1
BusinessId	String	否	公司统一社会信用代码 示例值: 546346435646
ModifyTime	Integer	否	修改时间 示例值: 1653534444
ContractNo	String	否	合同编号 示例值: 12324xxxxx

DescribePredictiveDialingCampaignsElement

查询预测式外呼任务列表元素

被如下接口引用：DescribePredictiveDialingCampaigns。

名称	类型	描述
CampaignId	Integer	任务 ID 示例值: 2569

名称	类型	描述
Name	String	任务名称 示例值：新能源SUV推荐
Status	Integer	任务状态 0 待开始 1 进行中 2 已暂停 3 已终止 4 已完成 示例值：0
StatusReason	Integer	任务状态原因 0 正常 1 手动结束 2 超时结束 示例值：0
CalleeCount	Integer	被叫号码个数 示例值：1000
FinishedCalleeCount	Integer	已完成的被叫个数 示例值：0
Priority	Integer	相同应用内多个任务运行优先级，从高到底 1 - 5 示例值：3
SkillGroupId	Integer	使用的座席技能组 ID 示例值：255

ErrStaffItem

批量添加客服时，返回出错客服的信息

被如下接口引用：CreateStaff。

名称	类型	描述
StaffEmail	String	座席邮箱地址 示例值：xxx@xxx.com
Code	String	错误码 示例值：err
Message	String	错误描述 示例值：some error

ExtensionInfo

话机信息

被如下接口引用：DescribeExtensions。

名称	类型	描述
SdkAppId	Integer	实例ID 示例值：1400000000
FullExtensionId	String	分机全名 示例值：xxx@xxx.com
ExtensionId	String	分机号 示例值：1001
SkillGroupId	String	所属技能组列表 示例值：1000,1001
ExtensionName	String	分机名称 示例值：分机1
CreateTime	Integer	创建时间 示例值：1617184839
ModifyTime	Integer	最后修改时间 示例值：1617184839

名称	类型	描述
Status	Integer	话机状态(0 离线、100 空闲、200忙碌) 示例值：100
Register	Boolean	是否注册 示例值：true
Relation	String	绑定座席邮箱 示例值：xxx@xxx.com
RelationName	String	绑定座席名称 示例值：座席1

Filter

筛选条件

被如下接口引用：DescribeActiveCarrierPrivilegeNumber, DescribeCarrierPrivilegeNumberApplicants。

名称	类型	必选	描述
Name	String	是	筛选字段名 示例值：Key
Values	Array of String	是	筛选条件值 示例值：["Value1","Value2"]

ForwardingConfig

呼转配置

被如下接口引用：DescribeStaffInfoList, ModifyStaff。

名称	类型	必选	描述
Enabled	Boolean	否	是否启用 示例值：true
Condition	Integer	否	1 无条件呼转 2 有条件呼转 示例值：2
Target	ForwardingTarget	否	呼转目标 示例值：{"Type":1,"StaffUserId":"foo@tencent.com"}

ForwardingTarget

呼转目标

被如下接口引用：ModifyStaff。

名称	类型	必选	描述
Type	Integer	否	呼转目标类型 1 座席 2 技能组 3 分机 示例值：1
StaffUserId	String	否	呼转目标为座席的账号 Type 为 1 时填写 示例值：foo@tencent.com
SkillGroupId	Integer	否	呼转目标为技能组的 ID，Type 为 2 时填写 示例值：1254
Extension	String	否	呼转目标为分机的账号，Type 为 3 时填写 示例值：1003@1400000000.sip.tccc.qcloud.com

IMCdrInfo

文本会话服务记录信息

被如下接口引用：DescribeIMCdrList, DescribeIMCdrs。

名称	类型	描述
Id	String	服务记录ID 示例值：12933
Duration	Integer	服务时长秒数 示例值：199
EndStatus	Integer	结束状态 0 异常结束 1 正常结束 3 无座席在线 17 座席放弃接听 100 黑名单 101 座席手动转接 102 IVR阶段放弃 108 用户超时自动结束 109 用户主动结束 示例值：1
Nickname	String	用户昵称 示例值：访客_e8d6
Type	Integer	服务类型 1为全媒体，2为文本客服 示例值：2
StaffId	String	客服ID 示例值：foo@tencent.com
Timestamp	Integer	服务时间戳 示例值：1632814604
SessionId	String	会话ID 示例值：87fa1d7a-4194-4f2a-83dc-d7c293411720
SkillGroupId	String	技能组ID 示例值：1154
SkillGroupName	String	技能组名称 示例值：foo
Satisfaction	IMSatisfaction	满意度 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"Id":1,"Label":"满意"}
ClientUserId	String	用户ID 示例值：user123

IMSatisfaction

IM满意度

被如下接口引用：DescribeIMCdrList, DescribeIMCdrs。

名称	类型	描述
Id	Integer	满意度值 示例值：2
Label	String	满意度标签 示例值：非常满意

IVRKeyPressedElement

ivr 按键信息

被如下接口引用：DescribeProtectedTelCdr, DescribeTelCdr。

名称	类型	描述
Key	String	命中的关键字或者按键 示例值：1
Label	String	按键关联的标签 示例值：非常满意
Timestamp	Integer	Unix 毫秒时间戳 示例值：1729697241440
NodeLabel	String	节点标签 示例值：分支节点
OriginalContent	String	用户原始输入 示例值：1
TTSPrompt	String	TTS 提示音内容 示例值：您好

Interface

回调接口

被如下接口引用：BindNumberCallInInterface。

名称	类型	必选	描述
URL	String	是	接口地址 示例值：https://cloud.tencent.com/

Message

单条消息

被如下接口引用：DescribeChatMessages。

名称	类型	描述
Type	String	消息类型 示例值：Text
Content	String	消息内容 示例值：您好，我是您的小助手

MessageBody

聊天消息

被如下接口引用：DescribeChatMessages。

名称	类型	描述
Timestamp	Integer	消息时间戳 示例值：1600007226
From	String	发消息的用户ID 示例值：Alan
Messages	Array of Message	消息列表 示例值：[{"Type":"Text","Content":"您好呀！"}]

NumberInfo

号码信息

被如下接口引用：DescribeNumbers。

名称	类型	描述
Number	String	号码 示例值：0086075512345678
CallOutSkillGroupIds	Array of Integer	绑定的外呼技能组 示例值：[1,2,3]
State	Integer	号码状态，1-正常，2-欠费停用，4-管理员停用，5-违规停用 示例值：1

OwnNumberApplyDetailItem

用户自带号码审批明细数据类型

被如下接口引用：CreateOwnNumberApply, ModifyOwnNumberApply。

名称	类型	必选	描述
CallType	Integer	是	号码类型：0-呼入 1-呼出 2-呼入呼出 示例值：2
PhoneNumber	String	是	线路号码 示例值：02066660000
MaxCallCount	Integer	是	最大并发呼叫数 示例值：1
MaxCallPSec	Integer	是	每秒最大并发数 示例值：1
OutboundCalleeFormat	String	否	呼出被叫格式，使用 {+E.164} 或 {E.164}， 示例值：{+E.164}

PSTNSession

PSTN 会话类型。

被如下接口引用：DescribeTelSession。

名称	类型	描述
SessionID	String	会话 ID 示例值：d62a280a-b7dc-4a05-8295-eba7b61c2490
RoomID	String	会话临时房间 ID 示例值：0
Caller	String	主叫 示例值：00861010101033
Callee	String	被叫 示例值：00861010101031
StartTimestamp	Integer	开始时间，Unix 时间戳 示例值：1617184839
RingTimestamp	Integer	振铃时间，Unix 时间戳 示例值：1617184839
AcceptTimestamp	Integer	接听时间，Unix 时间戳 示例值：1617184839
StaffEmail	String	座席邮箱 示例值：foo@tencent.com
StaffNumber	String	座席工号 示例值：007

名称	类型	描述
SessionStatus	String	会话状态 ringing 振铃中 seatJoining 等待座席接听 inProgress 进行中 finished 已完成 示例值: finished
Direction	Integer	会话呼叫方向, 0 呼入 1 - 呼出 示例值: 0
OutBoundCaller	String	转外线使用的号码 (转外线主叫) 示例值: 00861010101031
OutBoundCallee	String	转外线被叫 示例值: 00861010101032
ProtectedCaller	String	主叫号码保护ID, 开启号码保护映射功能时有效, 且Caller字段置空 示例值: 008613XXXX5678
ProtectedCallee	String	被叫号码保护ID, 开启号码保护映射功能时有效, 且Callee字段置空 示例值: 008613XXXX5678

PSTNSessionInfo

PSTN 会话信息

被如下接口引用: DescribePSTNActiveSessionList。

名称	类型	描述
SessionID	String	会话 ID 示例值: 1
RoomID	String	会话临时房间 ID 示例值: 1
Caller	String	主叫 示例值: 13712345678
Callee	String	被叫 示例值: 13712345678
StartTimestamp	String	开始时间, Unix 时间戳 示例值: 1617184839
AcceptTimestamp	String	接听时间, Unix 时间戳 示例值: 1617184839
StaffEmail	String	座席邮箱 示例值: xxx@xxx.com
StaffNumber	String	座席工号 示例值: 1
SessionStatus	String	座席状态 inProgress 进行中 示例值: inProgress
Direction	Integer	会话呼叫方向, 0 呼入 1 - 呼出 示例值: 0
RingTimestamp	Integer	振铃时间, Unix 时间戳 示例值: 1617184839
ProtectedCaller	String	主叫号码保护ID, 开启号码保护映射功能时有效, 且Caller字段置空 示例值: 008613XXXX5678
ProtectedCallee	String	被叫号码保护ID, 开启号码保护映射功能时有效, 且Callee字段置空 示例值: 008613XXXX5678

PackageBuyInfo

套餐包购买信息

被如下接口引用：DescribeCCCBuyInfoList。

名称	类型	描述
PackageId	String	套餐包Id 示例值：1001
Type	Integer	套餐包类型，0-外呼套餐包 1-400呼入套餐包 示例值：1
CapacitySize	Integer	套餐包总量 示例值：10000
CapacityRemain	Integer	套餐包剩余量 示例值：143
BuyTime	Integer	购买时间戳 示例值：1630000000
EndTime	Integer	截止时间戳 示例值：1640000000

PhoneNumBuyInfo

号码购买信息

被如下接口引用：DescribeCCCBuyInfoList。

名称	类型	描述
PhoneNum	String	电话号码 示例值：008602112345678
Type	Integer	号码类型，0-固话 1-虚商号码 2-运营商号码 3-400号码 示例值：1
CallType	Integer	号码呼叫类型，1-呼入 2-呼出 3-呼入呼出 示例值：2
BuyTime	Integer	购买时间戳 示例值：1630000000
EndTime	Integer	截止时间戳 示例值：1640000000
State	Integer	号码状态，1正常 2欠费停用 4管理员停用 5违规停用 示例值：1

SdkAppIdBuyInfo

应用购买信息

被如下接口引用：DescribeCCCBuyInfoList。

名称	类型	描述
SdkAppId	Integer	应用ID 示例值：1400000000
Name	String	应用名称 示例值：Test
StaffBuyNum	Integer	座席购买数（还在有效期内） 示例值：10
StaffBuyList	Array of StaffBuyInfo	座席购买列表（还在有效期内）

名称	类型	描述
		示例值：[]
PhoneNumBuyList	Array of PhoneNumBuyInfo	号码购买列表 示例值：[]
SipBuyNum	Integer	办公电话购买数（还在有效期内） 示例值：10

SeatUserInfo

座席用户信息

被如下接口引用：CreateStaff, DescribeProtectedTelCdr, DescribeTelCdr。

名称	类型	必选	描述
Name	String	是	座席名称 示例值：张三
Mail	String	是	座席邮箱 示例值：foo@tencent.com
StaffNumber	String	是	工号 示例值：007
Phone	String	否	座席电话号码（带0086前缀） 示例值：008613012345678
Nick	String	否	座席昵称 示例值：三三
UserId	String	否	用户ID 示例值：foo@tencent.com
SkillGroupNameList	Array of String	否	座席关联的技能组列表 示例值：["测试技能组"]
Role	Integer	否	1:管理员 2:质检员 3:普通座席 else:自定义角色ID 示例值：1
ExtensionNumber	String	否	座席分机号（1到8打头，4-6位） 示例值：1003

ServeParticipant

参与者信息

被如下接口引用：DescribeProtectedTelCdr, DescribeTelCdr。

名称	类型	描述
Mail	String	座席邮箱 示例值：foo@tencent.com
Phone	String	座席电话 示例值：008613012345678
RingTimestamp	Integer	振铃时间戳，Unix 秒级时间戳 示例值：1611631881
AcceptTimestamp	Integer	接听时间戳，Unix 秒级时间戳 示例值：1611631881
EndedTimestamp	Integer	结束时间戳，Unix 秒级时间戳 示例值：1611631891

名称	类型	描述
RecordId	String	录音 ID，能够索引到座席侧的录音 示例值：1611631881
Type	String	参与者类型，"staffSeat", "outboundSeat", "staffPhoneSeat" 示例值：staffSeat
TransferFrom	String	转接来源座席信息 示例值：bar@tencent.com
TransferFromType	String	转接来源参与者类型，取值与 Type 一致 示例值：staffSeat
TransferTo	String	转接去向座席信息 示例值：san@tencent.com
TransferToType	String	转接去向参与者类型，取值与 Type 一致 示例值：staffSeat
SkillGroupId	Integer	技能组 ID 示例值：1101
EndStatusString	String	结束状态 示例值：ok unconnected seatGiveUp seatForward outboundForward ivrGiveUp
RecordURL	String	录音 URL 示例值：https://xxxxxxx
Sequence	Integer	参与者序号，从 0 开始 示例值：0
StartTimestamp	Integer	开始时间戳，Unix 秒级时间戳 示例值：1611631881
SkillGroupName	String	技能组名称 示例值：tel
CustomRecordURL	String	录音转存第三方COS地址 示例值：https://xxxxxxx

ServerPushText

服务端控制AI对话机器人播报指定文本

被如下接口引用：ControlAIConversation。

名称	类型	必选	描述
Text	String	否	服务端推送播报文本
Interrupt	Boolean	否	是否允许该文本打断机器人说话
StopAfterPlay	Boolean	否	播报完文本后，是否自动关闭对话任务
Audio	String	否	服务端推送播报音频 格式说明：音频必须为单声道，采样率必须跟对应TTS的采样率保持一致，编码为Base64字符串。 输入规则：当提供Audio字段时，将不接受Text字段的输入。系统将直接播放Audio字段中的音频内容。
DropMode	Integer	否	默认为0，仅在Interrupt为false时有效 - 0表示当前有交互发生时，会丢弃Interrupt为false的消息 - 1表示当前有交互发生时，不会丢弃Interrupt为false的消息，而是缓存下来，等待当前交互结束后，再去处理 注意：DropMode为1时，允许缓存多个消息，如果后续出现了打断，缓存的消息会被清空 示例值：0
Priority	Integer	否	ServerPushText消息的优先级，0表示可被打断，1表示不会被打断。目前仅支持传入0，如果需要传入1，请提工单联系我们添加权限。 注意：在接收到Priority=1的消息后，后续其他任何消息都会被忽略（包括Priority=1的消息），直到Priority=1的消息处理结束。该字段可与Interrupt、DropMode字段配合使用。 例子：

名称	类型	必选	描述
			- Priority=1、Interrupt=true，会打断现有交互，立刻播报，播报过程中不会被打断 - Priority=1、Interrupt=false、DropMode=1，会等待当前交互结束，再进行播报，播报过程中不会被打断 示例值：0
AddHistory	Boolean	否	是否将文本加入到IIm历史上下文中 示例值：true

SkillGroupInfoItem

技能组信息

被如下接口引用：DescribeSkillGroupInfoList。

名称	类型	描述
SkillGroupId	Integer	技能组ID 示例值：1400000000
SkillGroupName	String	技能组名称 示例值：Test
Type	String	（废弃）类型：IM、TEL、ALL（全媒体） 示例值：IM
RoutePolicy	String	会话分配策略 示例值：fifo
UsingLastSeat	Integer	会话分配是否优先上次服务座席 示例值：0
MaxConcurrency	Integer	单客服最大并发数（电话类型默认1） 示例值：1
LastModifyTimestamp	Integer	最后修改时间 示例值：1611631881
SkillGroupType	Integer	技能组类型0-电话，1-在线，3-音频，4-视频 示例值：1
Alias	String	技能组内线号码 示例值：6666

SkillGroupItem

技能组信息

被如下接口引用：DescribeStaffInfoList。

名称	类型	描述
SkillGroupId	Integer	技能组ID 示例值：101
SkillGroupName	String	技能组名称 示例值：Test
Priority	Integer	优先级 示例值：1
Type	String	类型：IM、TEL、ALL（全媒体） 示例值：IM

StaffBuyInfo

座席购买信息

被如下接口引用：DescribeCCCBuyInfoList。

名称	类型	描述
Num	Integer	购买座席数量 示例值：10
BuyTime	Integer	购买时间戳 示例值：163000000
EndTime	Integer	截止时间戳 示例值：164000000
SipNum	Integer	购买办公电话数量 示例值：12

StaffInfo

带有技能组优先级的座席信息

被如下接口引用：DescribeStaffInfoList。

名称	类型	描述
Name	String	座席名称 示例值：xiaowang
Mail	String	座席邮箱 示例值：10982291@qq.com
Phone	String	座席电话号码 示例值：18092226822
Nick	String	座席昵称 示例值：xiaohong
StaffNumber	String	座席工号 示例值：101
RoleList	Array of Integer	用户角色id列表 示例值：[1]
SkillGroupList	Array of SkillGroupItem	所属技能组列表 示例值：0
LastModifyTimestamp	Integer	最后修改时间 示例值：1614149408
ExtensionNumber	String	座席分机号（1 到 8 打头，4 - 6 位） 示例值：1003
ForwardingConfig	ForwardingConfig	呼叫转移配置 示例值：{"Enabled":true,"Condition":2,"Target":{"Type":1,"StaffUserId":"foo@tencent.com"}}

StaffSkillGroupList

座席绑定技能组列表

被如下接口引用：BindStaffSkillGroupList。

名称	类型	必选	描述
SkillGroupId	Integer	是	技能组ID 示例值：2134
Priority	Integer	否	座席在技能组中的优先级（1为最高，5最低，默认3） 示例值：1

StaffStatusExtra

座席状态补充信息

被如下接口引用：DescribeStaffStatusMetrics。

名称	类型	描述
Type	String	im – 文本 tel – 电话 all – 全媒体 示例值：tel
Direct	String	in – 呼入 out – 呼出 示例值：in

StaffStatusMetrics

座席状态相关信息

被如下接口引用：DescribeStaffStatusMetrics。

名称	类型	描述
Email	String	座席邮箱 示例值：xxx@xxx.com
Status	String	座席状态 free 示闲 busy 忙碌 rest 小休 notReady 示忙 afterCallWork 话后调整 offline 离线 示例值：free
StatusExtra	StaffStatusExtra	座席状态补充信息
OnlineDuration	Integer	当天在线总时长 示例值：0
FreeDuration	Integer	当天示闲总时长 示例值：0
BusyDuration	Integer	当天忙碌总时长 示例值：0
NotReadyDuration	Integer	当天示忙总时长 示例值：0
RestDuration	Integer	当天小休总时长 示例值：0
AfterCallWorkDuration	Integer	当天话后调整总时长 示例值：0
Reason	String	小休原因 示例值：会议
ReserveRest	Boolean	是否预约小休 示例值：false
ReserveNotReady	Boolean	是否预约示忙 示例值：false
UseMobileAccept	Integer	手机接听模式：0 – 关闭 1 – 仅离线 2– 始终 示例值：0
UseMobileCallOut	Boolean	手机外呼开关 示例值：false
LastOnlineTimestamp	Integer	最近一次上线时间戳 注意：此字段可能返回 null，表示取不到有效值。 示例值：1679300000
LastStatusTimestamp	Integer	最近一次状态时间戳 注意：此字段可能返回 null，表示取不到有效值。 示例值：1679300000

名称	类型	描述
ClientInfo	Array of ClientInfo	客服登录的端信息 示例值：[{"ClientType": "Web", "IsConnected": true}]

TelCdrInfo

电话话单信息

被如下接口引用：DescribeProtectedTelCdr, DescribeTelCdr。

名称	类型	描述
Caller	String	主叫号码 示例值：008613012345678
Callee	String	被叫号码 示例值：008613012345670
Time	Integer	呼叫发起时间戳，Unix 时间戳 示例值：1611631881
Direction	Integer	呼入呼出方向 0 呼入 1 呼出 示例值：0
CallType	Integer	通话类型 1 呼出 2 呼入 3 音频呼入 5 预测式外呼 6 内线呼叫 示例值：1
Duration	Integer	通话时长 示例值：120
RecordURL	String	录音信息 示例值：https://xxxxx
RecordId	String	录音 ID 示例值：1611631881
SeatUser	SeatUserInfo	座席信息 示例值：0
EndStatus	Integer	EndStatus与EndStatusString一一对应，具体枚举如下： 场景 EndStatus EndStatusString 状态说明 电话呼入&呼出 1 ok 正常通话 电话呼入&呼出 0 error 异常结束 电话呼入 102 ivrGiveUp IVR 期间用户放弃 电话呼入 103 waitingGiveUp 排队时用户放弃 电话呼入 104 ringingGiveUp 振铃时用户放弃 电话呼入 105 noSeatOnline 无座席在线 电话呼入 106 notWorkTime 非工作时间 电话呼入 107 ivrEnd IVR 后直接结束 电话呼入 100 blackList 黑名单 电话呼出 2 unconnected 未接通 电话呼出 108 restrictedCallee 被叫因高风险受限 电话呼出 109 tooManyRequest 外呼超频限制 电话呼出 110 restrictedArea 外呼区域限制

名称	类型	描述
		电话呼出 111 restrictedTime 外呼时间限制 电话呼出 201 unknown 未知状态 电话呼出 202 notAnswer 被叫未接听 电话呼出 203 userReject 被叫拒接挂断 电话呼出 204 powerOff 被叫关机 电话呼出 205 numberNotExist 被叫空号 电话呼出 206 busy 被叫忙 电话呼出 207 outOfCredit 被叫欠费 电话呼出 208 operatorError 运营商线路异常 电话呼出 209 callerCancel 主叫取消 电话呼出 210 notInService 被叫不在服务区 电话呼入&呼出 211 clientError 客户端错误 电话呼出 212 carrierBlocked 运营商拦截 示例值：1
SkillGroup	String	技能组名称 示例值：tel
CallerLocation	String	主叫归属地 示例值：广东深圳
IVRDuration	Integer	IVR 阶段耗时 示例值：5
RingTimestamp	Integer	振铃时间戳，UNIX 秒级时间戳 示例值：1611631881
AcceptTimestamp	Integer	接听时间戳，UNIX 秒级时间戳 示例值：1611631881
EndedTimestamp	Integer	结束时间戳，UNIX 秒级时间戳 示例值：1611631881
IVRKeyPressed	Array of String	IVR 按键信息，e.g. ["1","2","3"] 示例值：["1","2","3"]
HungUpSide	String	挂机方 seat 座席 user 用户 system 系统 示例值：user
ServeParticipants	Array of ServeParticipant	服务参与者列表 示例值：0
SkillGroupId	Integer	技能组ID 示例值：110
EndStatusString	String	EndStatus与EndStatusString一一对应，具体枚举如下： 场景 EndStatus EndStatusString 状态说明 电话呼入&呼出 1 ok 正常通话 电话呼入&呼出 0 error 异常结束 电话呼入 102 ivrGiveUp IVR 期间用户放弃 电话呼入 103 waitingGiveUp 排队时用户放弃

名称	类型	描述
		电话呼入 104 ringingGiveUp 振铃时用户放弃 电话呼入 105 noSeatOnline 无座席在线 电话呼入 106 notWorkTime 非工作时间 电话呼入 107 ivrEnd IVR 后直接结束 电话呼入 100 blackList 黑名单 电话呼出 2 unconnected 未接通 电话呼出 108 restrictedCallee 被叫因高风险受限 电话呼出 109 tooManyRequest 外呼超频限制 电话呼出 110 restrictedArea 外呼区域限制 电话呼出 111 restrictedTime 外呼时间限制 电话呼出 201 unknown 未知状态 电话呼出 202 notAnswer 被叫未接听 电话呼出 203 userReject 被叫拒接挂断 电话呼出 204 powerOff 被叫关机 电话呼出 205 numberNotExist 被叫空号 电话呼出 206 busy 被叫忙 电话呼出 207 outOfCredit 被叫欠费 电话呼出 208 operatorError 运营商线路异常 电话呼出 209 callerCancel 主叫取消 电话呼出 210 notInService 被叫不在服务区 电话呼入&呼出 211 clientError 客户端错误 电话呼出 212 carrierBlocked 运营商拦截 示例值: ok
StartTimestamp	Integer	会话开始时间戳，UNIX 秒级时间戳 示例值: 1611631891
QueuedTimestamp	Integer	进入排队时间，Unix 秒级时间戳 示例值: 1611631891
PostIVRKeyPressed	Array of IVRKeyPressedElement	后置IVR按键信息（e.g. [{"Key": "1", "Label": "非常满意"}]） 示例值: 0
QueuedSkillGroupId	Integer	排队技能组Id 示例值: 110
SessionId	String	会话 ID 示例值: c2b303a5-7814-42f4-ab7d-573b7f195d30
ProtectedCaller	String	主叫号码保护ID，开启号码保护映射功能时有效，且Caller字段置空 示例值: 008613XXXX5678
ProtectedCallee	String	被叫号码保护ID，开启号码保护映射功能时有效，且Callee字段置空 示例值: 008613XXXX5678
UUI	String	客户自定义数据（User-to-User Interface） 示例值: whatever

名称	类型	描述
IVRKeyPressedEx	Array of IVRKeyPressedElement	IVR按键信息（e.g. [{"Key": "1", "Label": "非常满意"}]） 示例值：0
AsrUrl	String	获取录音ASR文本信息地址 示例值：https://api.tccc.qcloud.com/tcccadmin/cdr/getAsrText?session_id=
AsrStatus	String	AsrUrl的状态：Complete 已完成； Processing 正在生成中； NotExists 无记录(未开启生成离线asr或者无套餐包) 示例值：Complete
CustomRecordURL	String	录音转存第三方COS地址 示例值：https://xxxxx
Remark	String	备注 示例值：备注
QueuedSkillGroupName	String	排队技能组名称 示例值：销售组
VoicemailRecordURL	Array of String	通话中语音留言录音URL 示例值：["https://xxxxx"]
VoicemailAsrURL	Array of String	通话中语音留言ASR文本信息地址 示例值：["https://xxxxx"]

TimeRange

时间范围，24 小时制，格式为 09:00:00

被如下接口引用：CreateAutoCalloutTask, CreatePredictiveDialingCampaign。

名称	类型	必选	描述
StartTime	String	否	开始时间 示例值：09:00:00
EndTime	String	否	结束时间 示例值：14:00:00

UploadAudioInfo

上传音频文件信息

被如下接口引用：UploadIvrAudio。

名称	类型	必选	描述
CustomFileName	String	是	文件别名（可重复） 示例值：呼入使用
AudioUrl	String	是	音频文件链接。（支持mp3和wav格式，文件不超过5MB） 示例值：http://xxx.com/aa.mp3

UploadIvrAudioFailedInfo

上传音频文件失败信息

被如下接口引用：UploadIvrAudio。

名称	类型	描述
FileName	String	文件名 示例值：xxx.mp3
FailedMsg	String	失败原因 示例值：文件内容重复

Variable

变量

被如下接口引用：CreateAIAgentCall, CreateAICall, CreateAutoCalloutTask, CreateIVRSession, CreatePredictiveDialingCampaign, UpdatePredictiveDialingCampaign。

名称	类型	必选	描述
Key	String	是	变量名 示例值：foo
Value	String	是	变量值 示例值：bar

错误码

最近更新时间：2025-07-04 01:12:19

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。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数。

错误码	说明
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
FailedOperation.AIConcurrentNoPackage	无可用的AI并发许可，请去控制台进行购买
FailedOperation.AIConcurrentOverFrequency	AI并发超频
FailedOperation.CallOutFailed	外呼失败。
FailedOperation.CalleelsBlackUser	高风险用户，请勿呼叫
FailedOperation.CalleelsLimited	被叫号码外呼受限。
FailedOperation.CallerOverFrequency	主叫号码外呼超频。
FailedOperation.CalloutRuleBlindArea	触发默认被叫规则，呼叫盲区
FailedOperation.CalloutRuleMaxCallCountCalleeIntervalTime	触发默认外呼规则，被叫一段时间内呼叫量
FailedOperation.CalloutRuleMaxCallCountCalleePerDayAppID	触发默认外呼规则，被叫每日最大呼叫量
FailedOperation.CalloutRuleNotWorkTime	触发默认外呼规则，不在可外呼时间
FailedOperation.ControlAIConversation	ControlAIConversation错误信息
FailedOperation.CurStateNotAllowedModify	当前号码状态不能被修改。
FailedOperation.DuplicatedAccount	重复账号。
FailedOperation.NoCallOutNumber	无可用的外呼号码。
FailedOperation.PermissionDenied	权限不足。
FailedOperation.SeatStatusBusy	坐席处于忙碌状态。

错误码	说明
FailedOperation.SessionNotExists	会话不存在
FailedOperation.SessionNotInControlState	会话不在可控制状态
FailedOperation.UploadFileOverflow	上传文件个数超过限制
InternalError.DBError	内部数据库访问失败。
InvalidParameter.DuplicateAddress	重复的地址
InvalidParameter.DuplicatePhoneNumber	重复的号码
InvalidParameter.DuplicateSipAccount	重复的SIP账号
InvalidParameter.IllegalAddress	非法的地址
InvalidParameter.IllegalPhoneNumber	非法的号码
InvalidParameter.InstanceNotExist	实例不存在。
InvalidParameter.InvalidAddress	无效的地址
InvalidParameter.InvalidIP	无效的IP信息
InvalidParameter.InvalidPhoneNumber	无效的号码
InvalidParameter.InvalidPort	无效的端口信息
InvalidParameter.SipAccountPasswordFormat	密码不合法(长度大于等于八位，必须包含大小写字母以及数字)
InvalidParameter.SipAccountUserFormat	用户名不合法(只能包含A-Z、a-z、以及数字)
InvalidParameter.SipTrunkInUsed	SIP通道仍在使用中
InvalidParameter.SipTrunkNotFound	未找到SIP通道信息
InvalidParameterValue.AccountNotExist	账号不存在。
InvalidParameterValue.ApplyIDNotExist	审批单号不存在。
InvalidParameterValue.CompanyExist	公司信息已经存在。
InvalidParameterValue.ErrorApplyStatus	审核单状态错误。
InvalidParameterValue.ExperienceAccount	使用体验账号的智能体不支持此类操作
InvalidParameterValue.InstanceNotExist	实例不存在。
InvalidParameterValue.PhoneNumInvalid	号码状态无效。
InvalidParameterValue.PhoneNumIsBoundOtherAccount	号码已经绑定别的账号。
InvalidParameterValue.RecordNotExist	查询记录不存在。
InvalidParameterValue.RingAllGroupMemberOverflow	同振技能组人数超上限2，请确认并修改。
InvalidParameterValue.SkillGroupError	技能组错误。
InvalidParameterValue.SkillGroupExist	技能组已存在。
InvalidParameterValue.WaitingApprovalOverflow	待审核单已经超限，暂时不能再提交
LimitExceeded.BasePackageExpired	套餐包耗尽
LimitExceeded.OutOfCountLimit	超出数量限制。
OperationDenied.NotInWhiteList	不在白名单中。
OperationDenied.UinDisabled	账号已被禁用。