API 开发手册

云 PBX

版本: 2.0.0
日期: 2024年08月01日

目录

更新记录.....1
API 2.0 新增功能与优化.....1
API 2.0 更新记录.....4
简介.....7
API 简介.....7
API 接口&事件概览.....7
快速入门.....13
API 使用前提.....13
API 快速入门.....13
API 调用方式.....18
API 调用概述.....18
请求结构.....19
响应消息.....21
API 事件推送.....21
PBX API 网页设置.....23
API 接口说明.....27
API 认证.....27
API 认证机制.....27
获取 API token.....27
刷新 API token.....30
功能配置接口.....31
系统.....31
通讯录.....34
分机.....42
分机组.....69
中继.....76
呼入路由.....97

呼出路由 104
会议室 111
IVR 130
队列 134
语音留言 153
通话记录 160
录音 162
酒店应用 165
Linkus 客户端 174
呼叫操控接口 178
查询通话 178
发起呼叫 184
接受来电 186
拒接来电 188
监听通话 189
通话保持 191
恢复通话 192
通话静音 193
取消静音 194
呼叫停泊 196
查询被停泊的通话 197
呼叫转移(盲转) 199
呼叫转移（咨询转） 200
建立多方通话 203
播放语音 205
挂断广播语音 207
挂断通话 208
uaCSTA 呼叫操控接口 209
uaCSTA 呼叫操控接口概览 209
uaCSTA - 接受来电 211
uaCSTA - 拒接来电 213
uaCSTA - 挂断通话 214
事件推送 216

系统事件.....216
来电呼入控制流程事件.....218
ExtensionStatus (分机状态变更事件).....221
CallStatus（通话状态变更事件）.....222
NewCdr（通话记录事件）.....225
Transfer（呼叫转移事件）.....226
Forward（呼叫前转事件）.....229
CallFailed（呼叫失败事件）.....230
DTMF（按键信息事件）.....232
PlayPromptEnd（语音播放结束事件）.....238
satisfaction（满意度评价事件）.....243
ConferenceStatus（与会成员变更事件）.....244
uacstacall（uaCSTA 通话事件）.....246
QueueAgentRingNoAnswerTimeout (队列坐席响铃超时).....247
QueueAutoPause (队列坐席自动暂停).....248
错误代码.....249
API 场景应用.....255
酒店客房管理应用方案.....255
指挥调度系统应用方案.....264
呼叫中心系统应用方案.....269
IVR 预警系统应用方案.....275
IVR 语音互动服务应用方案.....278

更新记录

API 2.0 新增功能与优化

本节介绍 API 2.0 在 1.0 基础上的更新内容。

新增接口

- 新增以下会议室相关的接口：
  - 查询固定会议室（conference/query）
  - 查询临时会议室（instant_conference/query）
  - 查询会话中的会议室（conference/query_in_session）
  - 编辑固定会议室（conference/update）
  - 剔除成员（conference/kick_member）
  - 邀请成员（conference/add_member）
  - 静音/取消静音成员（conference/mute & conference/unmute）
- 新增多方通话接口（call/add_member）

接口统一与优化

统一接口

统一下列接口的请求地址。

API 1.0	API 2.0
合并建立通话接口	
extension/dial_extension	call/dial
extension/dial_outbound	
ivr/dial_extension	
ivr/dial_outbound	
queue/dial_outbound	
ringgroup/dial_outbound	
outbound/dial_outbound	
合并播放语音接口	
extension/playprompt	call/playprompt

API 1.0	API 2.0
outbound/playprompt	
extension/dial_number	
合并挂断通话接口	
extension/hangup	call/hungup
inbound/hangup	
outbound/hangup	
合并查询通话接口	
inbound/query	call/query
outbound/query	
合并通话转接接口	
calltransfer	call/attended_transfer call/attended_transfer_operate
inbound/transfer_number	
outbound/transfer_number	
合并通话监听接口	
extension/listen	call/listen
extension/whisper	
extension/barge	

优化接口

- 优化拒接来电接口：请求地址 `inbound/refuse` 修改为 `call/refuse_inbound`。
- 优化接听来电接口：请求地址 `inbound/accept` 修改为 `call/accept_inbound`。
- 优化通话静音接口：
  - 通话静音：请求地址 `extension/mute` 修改为 `call/mute`。
  - 取消静音：请求地址 `extension/unmute` 修改为 `call/unmute`。
- 优化通话保持接口：
  - 通话保持：请求地址 `extension/hold` 修改为 `call/hold`。
  - 恢复通话：请求地址 `extension/unhold` 修改为 `call/unhold`。
- 优化下载通话记录接口：请求参数 `extid` 修改为 `number`。
- 优化酒店相关接口：请求参数 `extid` 修改为 `number`。
- 优化语音留言相关接口：请求参数 `extid` 修改为 `number`。

- 优化广播相关接口：
  - 请求参数 `extid` 修改为 `number`。
  - 请求参数和响应参数去除 `id`。
  - 查询广播列表：请求地址 `paginggrouplist/query` 修改为`paginggroup/list`。

- 优化队列相关接口：
  - 请求参数 `extid` 修改为 `extnumber`。
  - 请求参数 `queueid` 修改为 `queuenumber`。
  - 队列迁出：请求地址 `queue/del_dynamicagent` 修改为 `queue/delete_dynamicagent`。
  - 查询队列：请求地址 `queuestatus` 修改为 `queue/query_status`。
  - 编辑队列：请求参数去除 `queueid`。
  - 查询队列设置：返回参数去除 `queueid`。
- 优化 IVR 相关接口：
  - 查询 IVR：请求参数 `ivrid` 修改为 `number`；返回参数 `ivrnumber` 修改为 `number`。
  - 修改 IVR 设置：请求参数去除 `ivrid`，`ivrnumber` 修改为 `number`。
- 优化中继相关接口：
  - 查询中继列表：请求地址 `trunklist/query` 修改为 `trunk/list`；返回参数新增 `id`。
  - 查询 SIP 中继设置：请求地址 `siptrunk/query` 修改为 `trunk/query_siptrunk`；请求参数去除 `trunkname`。
  - 创建 SIP 中继：请求地址 `siptrunk/add` 修改为 `trunk/add_siptrunk`；返回参数新增 `id`。
  - 编辑 SIP 中继：请求地址 `siptrunk/update` 修改为 `trunk/update_siptrunk`。
  - 删除 SIP 中继：请求地址 `siptrunk/updat` 修改为 `trunk/delete_siptrunk`；请求参数去除 `trunkname`。
- 优化分机相关接口：
  - 查询分机列表：请求地址 `extensionlist` 修改为 `extension/list`；返回参数 `extnumber` 修改为 `number`。
  - 查询分机设置：请求参数 `extid` 修改为 `number`；返回参数 `extnumber` 修改为 `number`。
  - 修改分机设置：请求参数去除 `extid`，`extnumber` 修改为 `number`。
- 优化呼入路由和呼出路由相关接口：
  - 请求参数 `name` 修改为 `id`。
  - 创建呼入路由或呼出路由：返回参数新增 `id`。

事件推送

- 变更如下报告的格式：
  - 来电呼叫请求（Invite）
  - 来电呼入请求（Incoming）
  - 呼叫失败（CallFailed）
  - 语音播放结束事件（PlayPromptEnd）
  - 呼叫转移事件（Transfer）
  - 呼叫前转事件（Forward）
  - 与会成员变更事件（ConferenceStatus）

API 2.0 更新记录

本节记录 API 2.0 的更新日志。

更新日期	更新内容
2024-4-7	• 针对会议室接口 `conference/query` 和 `conference/update` 新增以下参数： ◦ `mute`：查询、配置是否在会议室普通成员进入会议室时静音成员。 ◦ `prompt`：查询、配置成员进入和退出会议室的提示音。 • 针对队列接口 `queue/query` 和 `queue/update` 新增以下参数： ◦ `disallow_to_join_when_empty`：查询、配置是否允许呼叫者在队列无空闲坐席时加入队列。 ◦ `empty_defined_for_disallow_to_join_when_empty`：查询、配置视为无空闲坐席的场景。在此场景下禁止呼叫者加入队列。 ◦ `empty_defined_for_leave_empty`：查询、配置视为无坐席的场景。在此场景下呼叫者将被强制退出队列。 • 新增 `QueueAgentRingNoAnswerTimeout` 和 `QueueAutoPause` 事件推送。 • 修复以下会议室问题： ◦ 通过 `conference/add_member` 接口邀请成员加入会议室(启用了录音功能)，会议结束后无通话记录和通话录音。 ◦ 通过 `conference/query_in_session` 接口查询多个正在进行的会议，查询的第一个会议室无成员导致提示 `10110` 错误 (会议室号不存在) 。 • 修复队列问题：查询队列状态时无法获取主叫号码。 • 修复语音留言问题：如果用户通过物理中继呼入 PBX 并留言，查询语音留言时无法获取主叫号码。 • 修复录音问题：下载会议室通话的录音文件提示 `20014` 错误 (录音文件不存在)。

更新日期	更新内容
	• 修复以下酒店应用问题： ◦ 调用 `conference/add_member` 接口查询所有分机的闹铃时，API 意外断开连接。 ◦ 为注册在多个终端的分机添加闹铃后，只有一个终端会收到叫醒电话。
2021-05-24	• 新增呼叫停泊相关的接口： ◦ 停泊状态：`callpark_status` ◦ 停泊通话：`call/callpark` **注：** 取回停泊的通话，调用 `call/dial` 接口。 • 新增 uaCSTA 呼叫操控接口： ◦ 挂断电话：`uacstacall/hangup` ◦ 拒接来电：`uacstacall/refuse_inbound` ◦ 接听来电：`uacstacall/accept_inbound` • 新增支持 uaCSTA 通话事件推送。 • 新增分机组相关接口： ◦ 查询分机组设置：`extensiongroup/query` ◦ 删除分机组：`extensiongroup/delete` ◦ 修改分机组：`extensiongroup/update` ◦ 添加分机组：`extensiongroup/add`
2021-01-28	• 针对以下接口，新增 musiconhold 参数： ◦ conference/update：用于启用或禁用等待音乐。 ◦ conference/query：查看等待音乐的启用情况。 • 优化分机状态变更事件（ExtensionStatus）报告：新增报告参数 registeredIP，用于显示注册 SIP 账号的设备 IP 地址。 • 优化队列状态查询接口（queue/query_status)：响应参数新增主叫通话通道 ID（channelid)。 • 修复 NewCdr 报告问题：队列来电未接听时，NewCdr 报告中显示的参数 agentringtime 为 0。 • 修复会议室问题： ◦ 会议室中没有与会者时，通过 conference/add_member 接口无法邀请成员加入会议。 ◦ 使用 conference/update 接口更新某个会议室信息时，所有会议室号码会变成同一个。 • 修复请求录音随机串时提示 20012 错误的问题。
2020-11-09	• 支持通讯录相关的接口： ◦ companycontacts/query ◦ companycontacts/add ◦ companycontacts/update ◦ companycontacts/delete • 支持查询分机通话状态。

更新日期	更新内容
	• 优化以下接口，降低配置重载频率，提升系统稳定性。 ◦ extension/update • 修复自动录音问题：通过 API 发起呼叫时，无法自动录音。
2020-07-01	• 支持查询 Linkus 客户端状态。 • 支持向分机发送 Linkus 邮件。 • 支持启用或禁用分机的 Linkus 功能。 • 优化以下四个接口： ◦ inroute/update ◦ outroute/update ◦ siptrunk/update ◦ pagingroup/update • 修复 CDR 下载问题：通过 API 下载 CDR 文件，CDR 部分显示错乱。
2020-05-07	首次发布。

简介

API 简介

本文介绍云 PBX API 及使用 API 能够实现的功能。

云 PBX API

云 PBX 提供二次开发接口（API）。开发者可以利用 API 开发或者对接呼叫中心、酒店管理系统、CRM、OA 办公系统、计费系统等应用服务器。

交互方式

应用服务器跟 云 PBX 对接后，可以实现双向通信，双方互为服务器和客户端。

详细的通信方式和通信流程说明，请参见 API 调用概述和API 事件推送。

- **应用服务器通过 API 接口对 PBX 进行操作**：

  参数查询、状态查询、参数配置、呼叫操控等。
- **PBX 实时向应用服务器推送事件报告**：

  系统状态变更、分机状态变更、中继状态变更、呼叫状态变更、DTMF 按键信息、通话记录等。

主要功能

使用 API 接口与应用服务器对接后，可以实现四种类型的功能：

- 查询参数和状态
- 功能参数配置
- 呼叫操控
- 事件监控

API 接口&事件概览

本文提供云 PBX API 接口列表和推送事件列表。

API 认证

API Endpoint	描述
login	获取 API 调用凭证 token。
token/refresh	刷新 API 调用凭证 token。
heartbeat	延长 API 调用凭证 token 时效。

系统

API Endpoint	描述
deviceinfo/query	查询 PBX 信息。
logout	断开 API 连接。

通讯录接口

API Endpoint	描述
companycontacts/query	查询单个企业联系人、多个企业联系人或所有企业联系人的信息。
companycontacts/add	添加企业联系人。
companycontacts/update	修改企业联系人信息。
companycontacts/delete	删除企业联系人。

分机接口

API Endpoint	描述
extension/list	查询分机列表。
extension/query	查询单个分机、多个分机或所有分机的详细配置。
extension/query_call	查询单个分机、多个分机或所有分机的通话状态。
extension/add	添加分机。
extension/update	修改分机设置。
extension/delete	删除分机。

分机组接口

API Endpoint	描述
extensiongroup/query	查询单个分机组、多个分机组或所有分机组的详细配置。

API Endpoint	描述
extensiongroup/add	添加分机组。
extensiongroup/update	修改分机组配置。
extensiongroup/delete	删除分机组。

中继接口

API Endpoint	描述
trunk/list	查询中继列表。
trunk/query_siptrunk	查询单个 SIP 中继、多个 SIP 中继或所有 SIP 中继的详细配置。
trunk/add_siptrunk	添加 SIP 中继。
trunk/update_siptrunk	修改 SIP 中继配置。
trunk/delete_siptrunk	删除 SIP 中继。

呼入路由接口

API Endpoint	描述
inroute/query	查询单个呼入路由、多个呼入路由或所有呼入路由的详细配置。
inroute/add	添加呼入路由。
inroute/update	修改呼入路由配置。
inroute/delete	删除呼入路由。

呼出路由接口

API Endpoint	描述
outroute/query	查询单个呼出路由、多个呼出路由或所有呼出路由的详细配置。
outroute/add	添加呼出路由。
outroute/update	修改呼出路由配置。
outroute/delete	删除呼出路由。

IVR 接口

API Endpoint	描述
ivr/query	查询单个 IVR、多个 IVR 或所有 IVR 的详细配置。
ivr/update	修改 IVR 配置。

队列接口

API Endpoint	描述
queue/query	查询单个队列、多个队列或所有队列的详细配置。
queue/update	修改队列配置。
queue/query_status	查询队列坐席状态。
queue/add_dynamicagent	动态坐席签入队列。
queue/delete_dynamicagent	动态坐席签出队列。
queue/pause_agent	暂停坐席服务。
queue/unpause_agent	恢复坐席服务。

会议室接口

API Endpoint	描述
conference/query	查询固定会议室详情。
instant_conference/query	查询临时会议室详情。
conference/query_in_session	查询会话中的会议室详情。
instant_conference/start	添加临时会议室。
conference/update	修改固定会议室配置。
conference/add_member	邀请会议室成员。
conference/kick_member	踢除与会人员。
conference/mute_member	对与会成员进行静音。
conference/unmute_member	对与会成员进行静音解除。

语音留言接口

API Endpoint	描述
voicemail/query	查询单个分机、多个分机或所有分机的语音留言。
voicemail/delete	删除分机的语音留言。
voicemail/get_random	获取语音留言下载随机串。

通话记录和录音接口

API Endpoint	描述
recording/get_random	获取录音下载随机串。

API Endpoint	描述
cdr/get_random	获取通话记录下载随机串。

酒店应用接口

API Endpoint	描述
wakeupcall/create	添加酒店叫醒闹铃。
wakeupcall/query	查询酒店叫醒闹铃。
wakeupcall/update	修改酒店叫醒闹铃。
wakeupcall/delete	删除酒店叫醒闹铃。

呼叫操控接口

API Endpoint	描述
call/query	查询指定通话或指定类型的通话。
call/dial	发起呼叫。
call/accept_inbound	在呼入来电控制阶段，接听来电。
call/refuse_inbound	在呼入来电控制阶段，拒接来电。
call/listen	监听通话。
call/hold	保持通话。
call/unhold	恢复通话。
call/mute	通话静音。
call/unmute	取消通话静音。
call/callpark	停泊通话。
call/callpark_status	查询通话停泊状态。
call/transfer	呼叫转移（盲转）。
call/attended_transfer	呼叫转移（咨询转）。
call/add_member	通话过程中，邀请新成员加入通话。
call/playprompt	播放语音文件。
paginggroup/hangup/music	挂断广播语音。

uaCSTA 呼叫操控接口

API Endpoint	描述
uacstacall/accept_inbound	通过 uaCSTA 接听来电。
uacstacall/refuse_inbound	通过 uaCSTA 拒接来电。
uacstacall/hangup	通过 uaCSTA 挂断通话。

事件报告类型

事件	描述
BootUp	系统启动事件。
ConfigChange	配置变更事件。
Invite	启用来电接听控制后，当有外部来电时，PBX 向应用服务器推送 Invite 事件，并预留 10 秒时间，让应用服务器选择接听或拒接来电。
Incoming	启用来电接听控制后，当 PBX 接受外部来电时，PBX 向应用服务器推送 Incoming 事件，并预留 10 秒时间，让应用服务器选择来电目的地。 **注：** 应用服务器使用 call/transfer 接口来控制来电目的地。
ExtensionStatus	分机状态变更事件。
CallStatus	通话状态变更事件。
NewCdr	新的通话记录事件。
CallTransfer	呼叫转移事件。
CallForward	呼叫前转事件。
CallFailed	呼叫失败事件。
DTMF	按键信息事件。
PlayPromptEnd	语音播放结束事件。
satisfaction	满意度评价事件。
sms-send	语音发送状态事件。
ConferenceStatus	与会成员变更事件。
uacstacall	uaCSTA 通话事件。
QueueAgentRingNoAnswerTimeout	队列坐席响铃超时事件。
QueueAutoPause	队列坐席自动暂停事件。

快速入门

API 使用前提

本文介绍使用云 PBX 二次开发接口之前，你要做的前提配置。

前提配置

使用 API 之前，你需要先做好以下配置：

PBX 版本

本文档的描述基于 API 2.0.0，你需要升级 PBX 固件到适配 API 2.0.0 接口的版本。

PBX 版本要求：固件版本 81.13.0.30 或更高版本

启用 API 功能

默认情况下，PBX 的 API 功能是禁用的。使用 API 之前，你需要先在 PBX API 网页设置 页面启用 API ，并做好相关的配置。

API 快速入门

本文以Yeastar API demo 为例，为你演示如何配置 API 、调用 API 接口以及如何查看事件报告。

前提条件

在使用 API 之前，确保你已升级 PBX 固件，并在 PBX 网页启用 API 功能和做好相应的配置。

PBX 固件要求，请参见 API 使用前提。

PBX API 设置说明，请参见PBX API 网页设置。

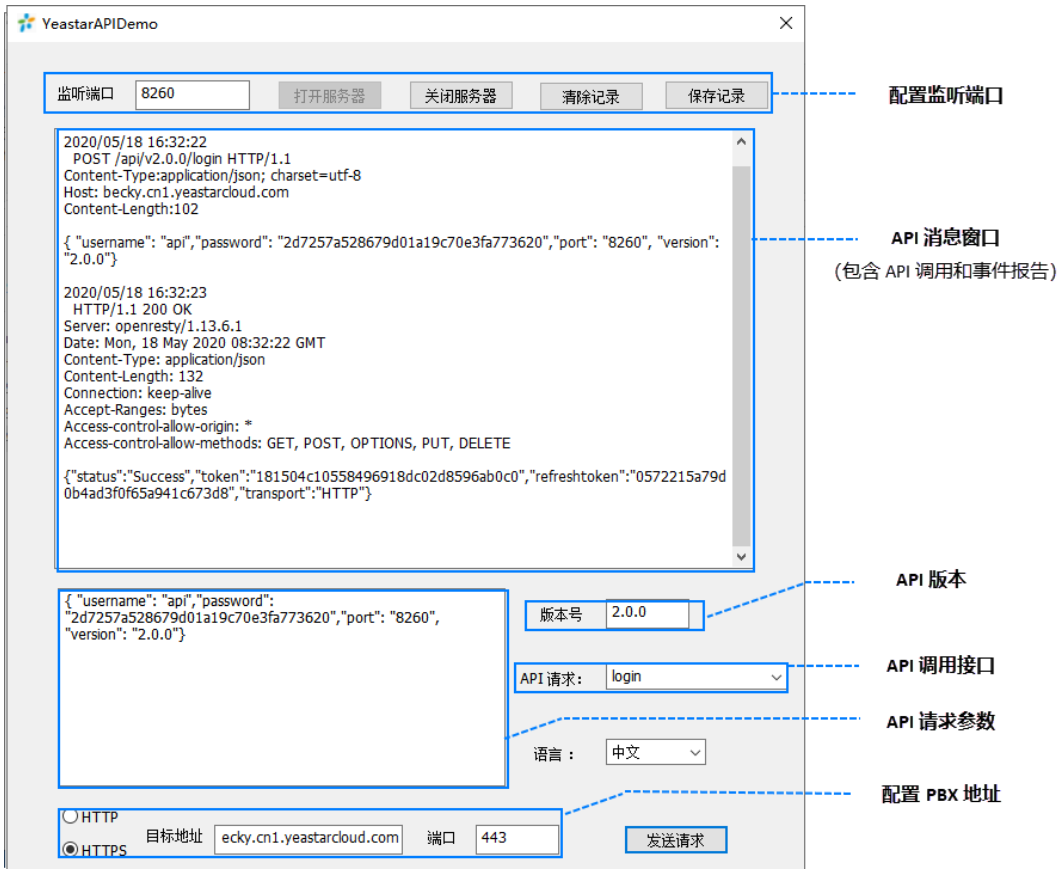
API demo

下载 API demo

点击下载 API demo 2.0 工具。

API demo 界面介绍

查看下面图示，了解 API demo 工具的界面布局。



API demo 测试示例

下面以Yeastar API demo 作为 API 调用工具，为你演示如何使用 API 接口 call/dial 完成分机拨打分机的任务，以及如何查看相关的事件报告。

> **注：**
> 调用 API 之前，请根据 API 接口参考文档了解使用说明，并查询必需的请求参数。如果 API 调用出错，你可以根据错误代码进行分析和排查错误。

API demo 设置

1. 在 API demo 中填写 API 版本、 PBX 的域名和通信端口。

- **版本号**：填写 API 版本号 2.0.0。

> **注：**
> 本文档中，所有接口和事件报告均采用的 API 版本为 2.0.0。

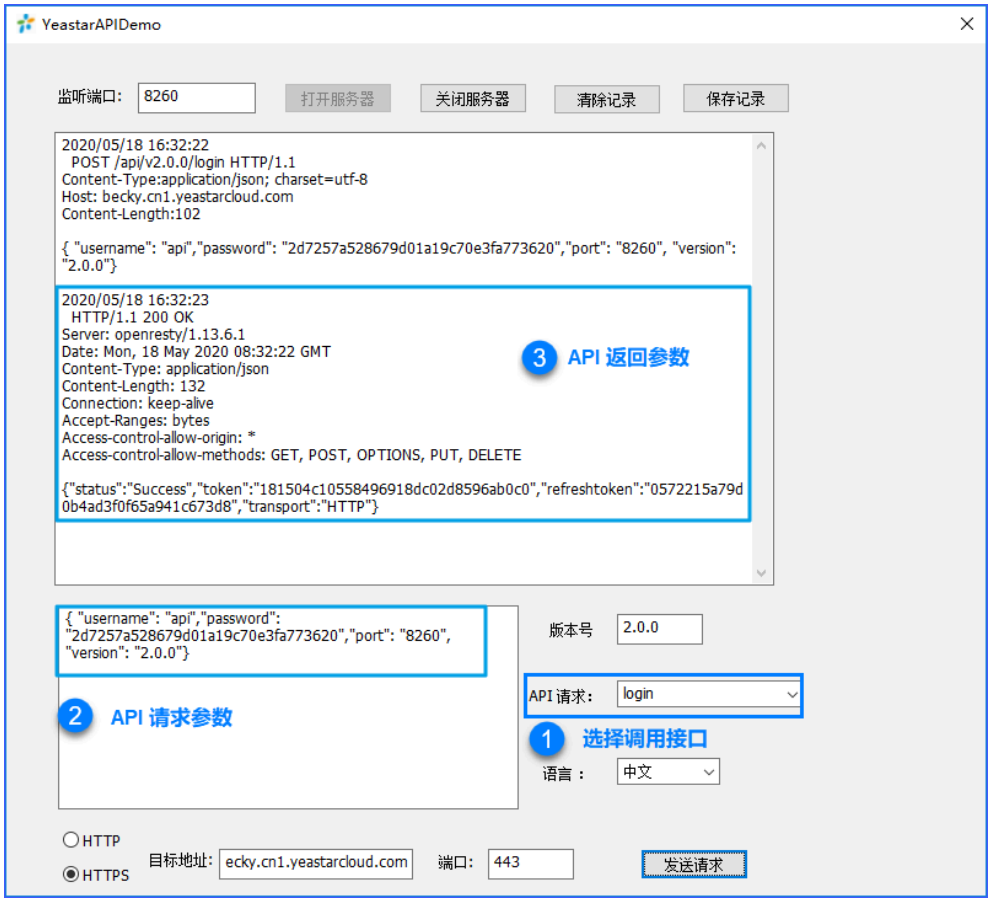
- **HTTP/HTTPS**： 选择 **HTTPS**。
- **目标地址**：填写 PBX 的域名。
- **端口**：填写 PBX 的网页端口。本示例填写 443。

2. 在 API demo 中配置监听端口，并点击 **打开服务器**，监听该端口。

   PBX 会通过该端口发送事件报告。

获取 API 授权

1. 调用 login 接口获取 API token。

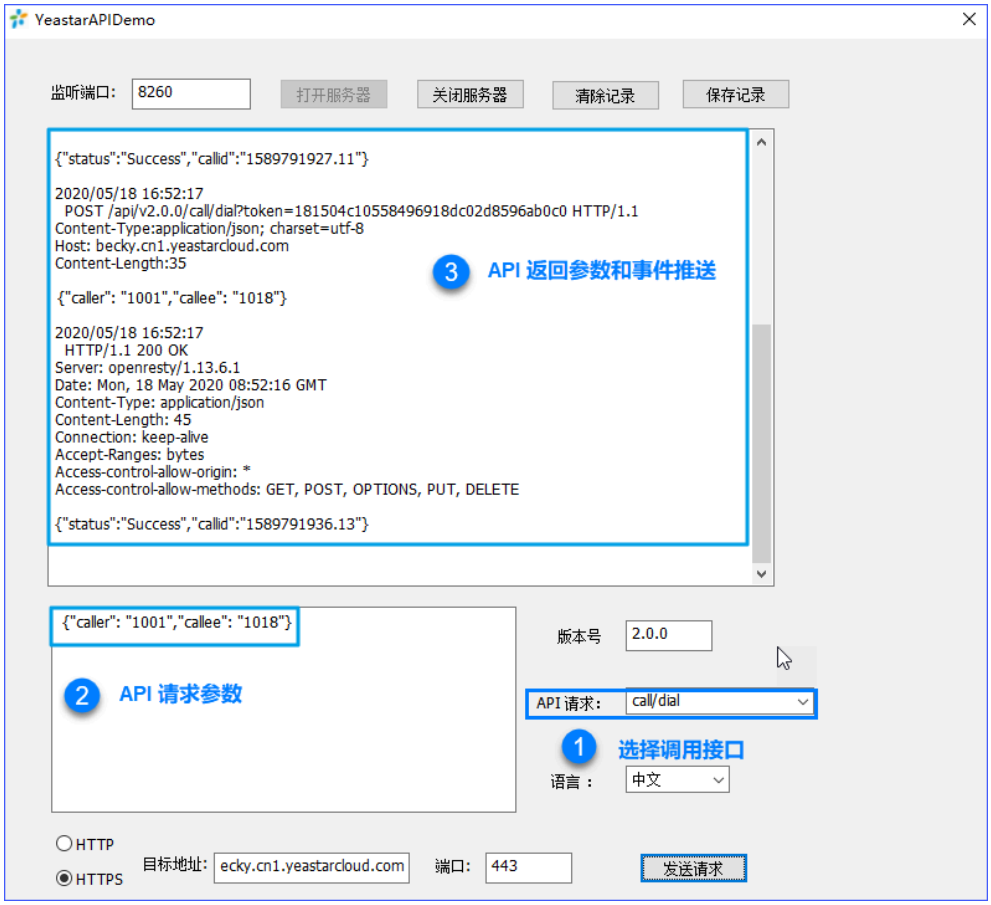


API 调用

1. 调用 call/dial 向分机发起呼叫。

   本例中，演示分机 1001 向分机 1018 发起呼叫，请求参数如下。

```
{"caller": "1001","callee": "1018"}
```



观察调用结果

如果该 API 调用成功后，后续流程为：

1. 主叫分机 1001 先振铃。
2. 分机 1001 摘机后，被叫分机 1018 开始振铃。
3. 分机 1018 接听来电后，双方建立通话。
4. 任意一方挂机后，通话结束。

查看事件报告

本次呼叫过程中，应用服务器会接收到通话状态变更事件（Call Status）和分机状态变更事件（ExtensionStatus）。

- 关于 CallStatus 事件的参数说明，请参见CallStatus（通话状态变更事件）。
- 关于 ExtensionStatus 事件的参数说明，请参见ExtensionStatus (分机状态变更事件)。

通话结束后，应用服务器还会接收到一条通话记录事件（NewCdr）。

- 关于 NewCdr 事件的参数说明，请参见NewCdr（通话记录事件）。

API 调用方式

API 调用概述

本文介绍 API 调用的通信协议、通信端口和交互流程。

通信协议

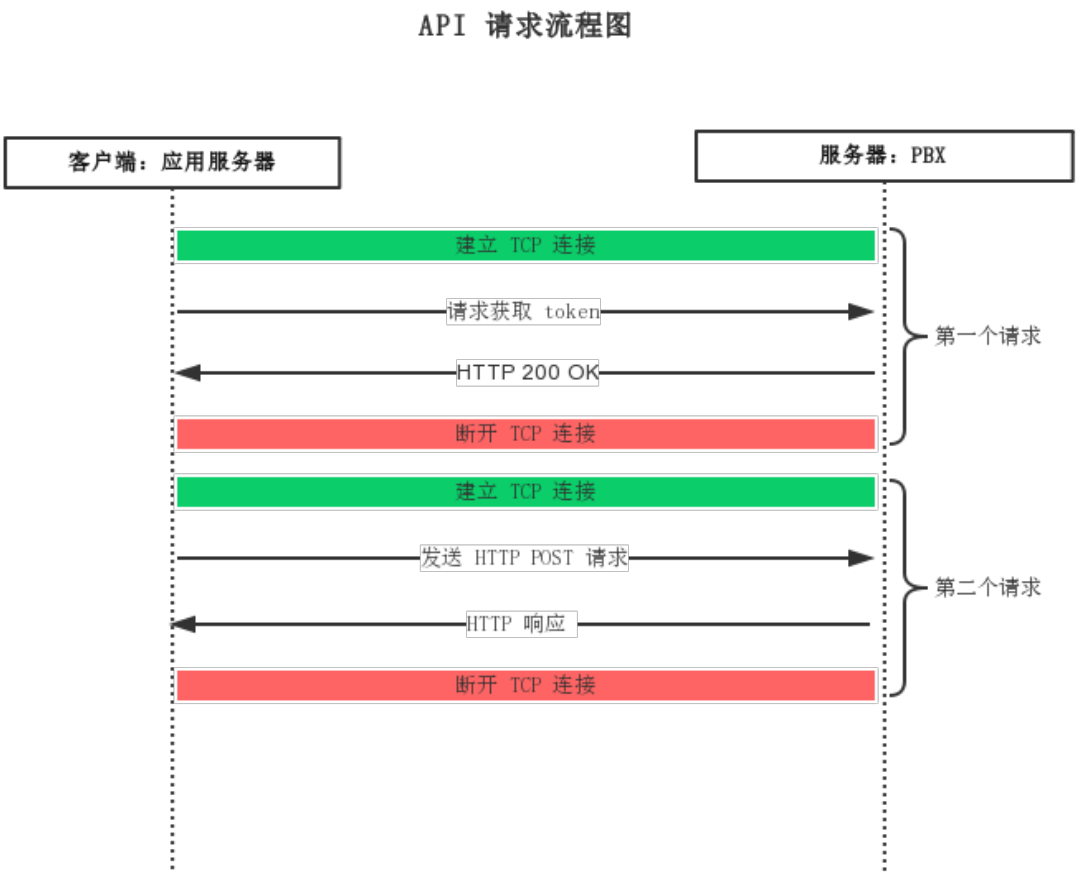
API 调用接口采用 HTTP/HTTPS POST 方法、TCP 短连接的方式，每个 API 请求都会重新建立一个 TCP 连接。

通信端口

API 调用的通信端口与 PBX 网页访问端口一致，使用 HTTPS 443。

交互流程

应用服务器发送 API 请求时，应用服务器作为客户端，PBX 作为服务器。下图为 API 请求流程示例图。



请求结构

发起 API 请求的 URL 由不同参数拼凑而成，有固定的请求结构。本文介绍 云 PBX的 API 请求结构，包括通信协议、请求方式和请求地址说明。

结构示例

下面为 签入队列的请求示例。

```
POST /api/v2.0.0/
queue/add_dynamicagent?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
```

```
 "queuenumber": "6700",
 "extnumber": "1000",
 "password": "6700123"
}
```

- `POST` 表示请求方式。
- `/api/v2.0.0` 即 API 路径，表示 API 版本号。
- `queue/add_dynamicagent` 表示要调用的 API 接口签入队列。
- `token=1e3b3ebb6a974cb42ed31de5413df52d` 表示用于验证请求合法性的 token。
- `Content-Type:application/json; charset=utf-8` 表示请求内容格式。
- `Host` 表示云 PBX 的域名。
- `queuenumber`、`extnumber`、`password` 表示具体请求参数。

通信协议

云 PBX API 使用 HTTPS 通信协议，所有 API 请求均使用 HTTPS POST 方式调用。

请求地址格式

API URL 由 Base URL、Endpoint 和 API token 组成，统一格式如下。

> **注：**
> 获取 API token 和 刷新 API token 请求地址无需带 token。

```
{base_url}/{api_path}/{endpoint}?token={token}
```

{base_url}

本手册中，所有 API 请求示例，都使用下面的的Base URL：

```
https://{pbx_domain}
```

{api_path}

API 的路径格式为 `api/{version}`，其中，`{version}` 为 API 版本号。

本文档中，使用的 API 版本为 v2.0.0，所有的 API 接口的 api_path 都是 `api/v2.0.0`

{endpoint}

Endpoint （终点），也称作 “路径”，表示 API 请求地址的具体地址。

关于 云 PBX 支持的所有 Endponit，请参见 API 接口&事件概览。

{token}

Token 为 API 接口的调用凭证。应用服务器请求调用 PBX 的 API 接口之前，必须先 获取 API token。之后的所有 API 请求都必须在 API URL 带上 token，否则 API 接口调用将会失败。

详细的 API 认证说明，请参见API 认证机制。

响应消息

云 PBX 返回的响应消息为 JSON 格式。

不管是正常响应消息或异常响应消息，HTTP 状态码为 200 OK。

正常响应示例

接口调用成功后会返回“Success” 状态，有时候还会返回相应的参数。我们称这样的响应为正常响应。HTTP 状态码为 200 OK。

```
HTTP/1.1 200 OK
{
    "status": "Success"
}
```

异常响应示例

接口调用出错后，会返回“Failed” 状态和错误码。我们称这样的响应为异常响应。HTTP 状态码为 200 OK。

你可以根据错误代码进行分析和排查错误。

```
HTTP/1.1 200 OK
{
    "status": "Failed",
    "errno": "10003"
}
```

API 事件推送

本文介绍 云 PBX 的事件推送方式及其通信协议、通信端口和交互流程。

事件推送方式

云 PBX 支持 TCP 短连接向应用服务器推送 HTTP 事件报告。

HTTP 事件推送

通信协议

PBX 采用 TCP 短连接的方式发送 HTTP 事件报告。

短连接时，PBX 每次推送事件都会新建 TCP 连接，应用服务器收到事件后应断开连接。

通信端口

应用服务器请求 获取 API token，登录 PBX 时，需要带 `port` 参数。`port` 表示事件报告监听端口。PBX 将事件报告通过该端口发送到应用服务器。

事件报告格式

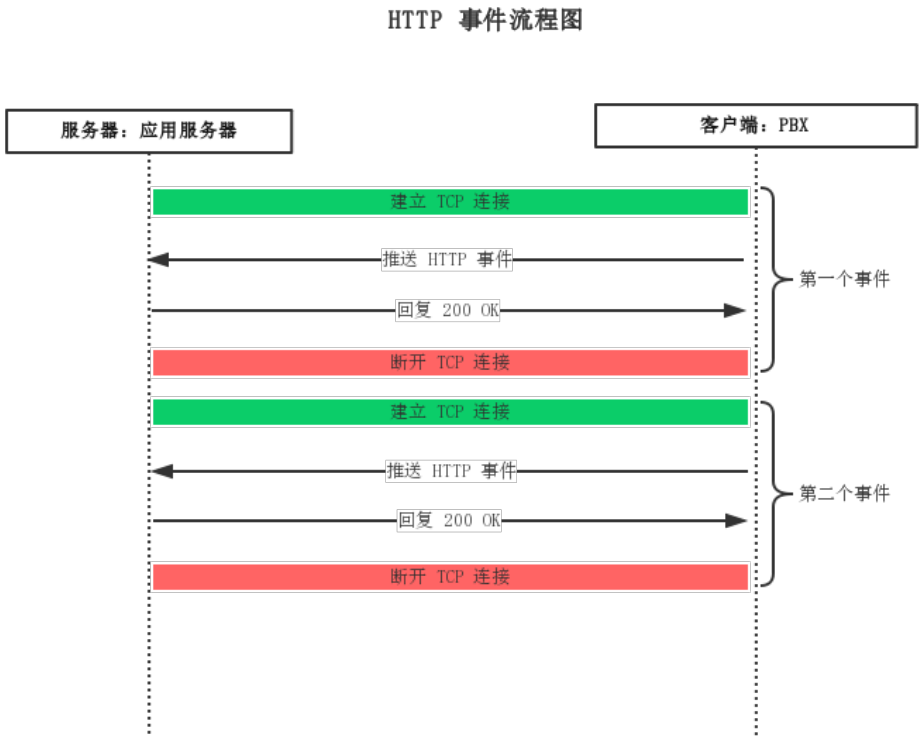
JSON 格式。

推送前提

选用此方式推送事件报告，应用服务器必须先获取 API token 连接到 PBX，并且保持 token 的有效性，否则应用服务器无法接收到事件报告。

交互流程

下图为 HTTP 事件推送流程示例图。



PBX API 网页设置

在连接应用服务器与 PBX 之前，你要先在 PBX 网页启用 API 并做好相关的配置。本文介绍 PBX API 的相关网页设置。

二次开发接口- 基本设置

使用 API 功能，你必须启用 API 并做好基本设置。

登录 PBX 网页，进入 **配置 > PBX > 常规设置 > 二次开发接口**，启用并更改 API 基本设置。

- **启用**：是否启用 API 功能。
- **用户名**：填写自定义的用户名。
- **密码**：填写自定义的密码。

> **注：**
> 此处的用户名和密码是应用服务器对接到 PBX 时要用到的用户名及密码。
> - 请求获取 API token 带的 password 参数为MD5 加密后的32位小写字符串。

二次开发接口- 高级设置

如果你需要在应用服务器监控分机和外线的状态或操控外线来电，你需要在 **高级设置** 里开启相应的开关。

登录 PBX 网页，进入 **配置 > PBX > 常规设置 > 二次开发接口**，更改 API 高级设置。

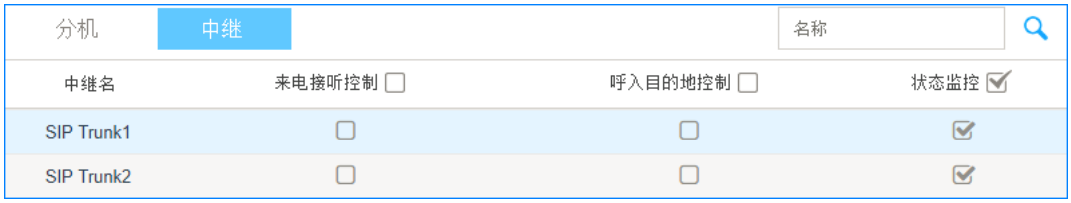
分机

选择是否监控分机的状态。

当该分机状态有变更时（如：分机状态从忙变为空闲），PBX 会主动向应用服务器发送ExtensionStatus (分机状态变更事件)。

中继

选择是否控制中继的来电、是否监控中继的通话状态。



- **来电接听控制**：勾选来电接听控制之后，当外线来电通过该中继呼入时，PBX 将发送Invite（来电呼叫请求）到应用服务器，并预留 10 秒时间让应用服务选择接受或拒绝来电。

> **注：**
> 如果应用服务器没有在 10 秒内作出操作，PBX 将接受该来电，并将该来电引导到呼入路由的目的地。

  - **接受**：应用服务器调用 call/accept_inbound 来接受来电。

    来电会进入对应呼入路由目的地。
  - **拒绝**：应用服务器调用 call/refuse_inbound 来拒绝来电。

    来电会被挂断，且不会到达 PBX。PBX 不会生成相关的通话记录。
- **呼入目的地控制**：勾选呼入目的地控制后，当外线来电通过该中继呼入且已被 PBX 接受时，PBX 将发送Incoming（来电呼入）到应用服务器，且预留 10 秒时间让应用服务器操控该来电的目的地。

> **注：**
> - 应用服务器调用 call/transfer 来操控来电的目的地。
> - 如果应用服务器没有在 10 秒内作出操作，PBX 会将来电引导到呼入路由的目的地。

- **状态监控**：勾选中继的状态监控后，PBX 会发送该中继的CallStatus（通话状态变更事件）到应用服务器。

删除被拉黑的 API IP

如果某个 IP 在短时间内，多次尝试登陆 API 接口失败，PBX 会将该 IP 拉入禁止名单，并将该 IP 类型标记为“API-IP”。如果该 IP 为受信任的 IP，你可以将这个 IP 从禁止名单列表中删除。

1. 进入 **配置 > 系统 > 安全中心 > 访问控制 > IP 禁止名单**，查看被拉黑的 IP。

访问控制
服务
IP许可名单
限制国家访问
IP禁止名单
删除
类型
被攻击时间
协议
被攻击端口
源IP地址
删除
API-IP
2018-10-19 14:37:44
TCP
443
192.168.7.24(api,IP is block...

2. 选择 IP 地址，点击 。

API 接口说明

API 认证

API 认证机制

本文介绍 API 调用和推送事件报告的认证机制。

认证机制

API 调用和事件报告采用 token 进行认证。

- **API 调用**

  应用服务器需要通过 API 用户名和密码（MD5 加密32位小写字母）向 PBX 获取 API token，每次调用 API 接口都要带上有效的 token。

  > **注：**
  > 通过刷新 API token， 保证 token 的有效性。

- **HTTP 事件报告**

  应用服务器需要先通过 API 用户名和密码（MD5 加密32位小写字母）向 PBX 获取 API token，并保证 token 的有效性，PBX 才会推送事件报告到应用服务器。

  > **注：**
  > 通过刷新 API token， 保证 token 的有效性。

Token 时效

token 的有效时长为 30 分钟。

为保证应用服务器能够与 PBX 正常通信，应用服务器必须在每 30 分钟之内刷新 API token。

获取 API token

云 PBX API 采用用户名和密码的方式验证，只有用户名和密码验证通过的应用服务器，API 才会处理其发送的请求。

> **注：**
> 如果密码连续错误五次，则应用服务器的 域名将会被锁 10 分钟。登录 PBX 网页，可以马上删除被拉黑 IP 地址。

请求地址

```
{base_url}/{api_path}/login
```

请求参数

参数	是否必选	类型	描述
username	是	String	在 PBX API 网页设置的用户名。
password	是	String	将PBX API 网页设置的密码经过 MD5 加密，填写加密后的32位小写字符串。
version	否	String	发送事件报告格式的版本。 **注：** • 本文档使用事件报告版本为 2.0.0。 • 如果不带此参数，默认发送的事件报告版本为 1.0.0，建议设置此参数，获取正确格式的事件报告。
port	否	String	应用服务器要监听此端口，接收 PBX 发送的事件报告。 **默认值**：8260。
url	否	String	应用服务器获取事件报告的 URL。 **注：** • 如果不带此参数，则 PBX 默认向应用服务器的 IP 地址发送事件报告。
urltag	否	String	指定 URL 的类型。 **取值范围**： • 0：定义`url`的值为相对路径。PBX 会将相对路径的`url`与应用服务器的 IP 地址结合起来，发送事件报告到具体的路径。

参数	是否必选	类型	描述
			例如：应用服务器 IP 地址为110.22.2.3；监听端口为8260；`url`设置为report；则 PBX 发送事件报告到110.22.2.3:8260/report 。 • 1：定义`url`的值为绝对路径。 例如：应用服务器 IP 地址为110.22.2.3；`url`设置为110.22.2.3:8260/report；则 PBX 发送事件报告到110.22.2.3:8260/report 。 **注：** 使用绝对路径，可以避免复杂网络环境下，事件报告无法发送成功的问题。 **默认值**：0。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
token	String	调用接口凭证，所有的 API 接口调用都需要带 token。
refreshtoken	String	使用`refreshtoken` 获取新的 `token` 和 `refreshtoken`。
transport	String	PBX 向应用服务器推送报告的协议。 • HTTP：PBX 采用 TCP 短连接的方式发送 HTTP 事件报告。 关于事件报告发送形式的说明，请参见API 事件推送。

示例

请求示例

```
POST /api/v2.0.0/login HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "username": "api",
```

```
    "password": "2d7257a528679d01a19c70e3fa773870",
    "port": "8260",
    "version": "2.0.0"
}
```

响应示例

第三方应用服务器登录成功。

```
HTTP/1.1 200 OK
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success",
    "token": "2bc5a7494ef88b0506da66c60ccf232d",
    "refreshtoken": "58f29c380365e5546cd3ac8c34edc88c",
    "transport": "TCP"
}
```

第三方应用服务器登录失败。

```
HTTP/1.1 200 OK
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
      "status": "Failed",
      "errno": "30003"
}
```

刷新 API token

刷新API token，确保应用服务器能跟 PBX 正常通信。

请求地址

```
{base_url}/{api_path}/token/refresh
```

请求参数

参数	是否必选	类型	描述
refreshtoken	是	String	最近一次返回的 refreshtoken 值。

响应参数

参数	类型	描述
token	String	新的 API token。
refreshtoken	String	新的 refreshtoken，用于下次刷新 token 的请求。

示例

请求示例

```
POST /api/v2.0.0/token/refresh HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "refreshtoken": "58f29c380365e5546cd3ac8c34edc88c"
}
```

响应示例

```
HTTP/1.1 200 OK
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success",
    "token": "1c0a3fd34ca5e5292042a77e4d9ff950",
    "refreshtoken": "4a1f20e901b45d4106e7aee076d22115"
}
```

功能配置接口

系统

查询 PBX 信息

查询 PBX 的基本信息，如产品型号、固件版本、启动时间等。

请求地址

```
{base_url}/{api_path}/deviceinfo/query?token={token}
```

请求参数

无参数，直接发送查询设备信息的请求即可。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
deviceinfo	Object<DeviceInfo>	设备信息。

DeviceInfo

参数	类型	描述
devicename	String	产品设备名称。
sn	String	产品序列号。
firmwarever	Integer	固件版本号。
systemtime	String	系统时间。
uptime	Integer	启动时间。
extensionstatus	String	已创建分机数/最大分机数。

示例

请求示例

```
POST /api/v2.0.0/deviceinfo/query?token=6fc544c66fd06a1a6eb4b4c4547f6d06
Content-Type:application/json; charset=utf-8
Host:api.yeastarcloud.com
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
```

```
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "deviceinfo": {
  "devicename": "Yeastar Cloud PBX",
  "sn": "36828053U0H2BTHG",
  "firmwarever": "81.9.0.23",
  "systemtime": "2019-10-15 20:07:32 UTC-8",
  "uptime": "00:57:59",
  "extensionstatus": "3/10"
 }
}
```

断开 API 连接

请求断开应用服务器与 PBX API 接口的连接。

请求地址

```
{base_url}/{api_path}/logout?token={token}
```

> **注：**
> 请求地址中的 token 为上次最新的 API token。

请求参数

无参数，直接发送登出的请求即可。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/logout?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

通讯录

查询企业联系人

查询单个企业联系人、多个企业联系人或所有企业联系人的信息。

请求地址

```
{base_url}/{api_path}/companycontacts/query?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	企业联系人的唯一 ID。 **取值范围**： • 单个企业联系人 id：查询指定企业联系人的信息。 例如：`"id":"11"` • 多个企业联系人 id：查询多个企业联系人的信息。多个企业联系人 id 用半角逗号（,）隔开。 例如：`"id":"11,12"` • all：查询所有企业联系人的信息。

参数	是否必选	类型	描述
			例如：`"id":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
companycontacts	Array <CompanycontactsInfo>	企业联系人的信息。

CompanycontactsInfo

参数	类型	描述
id	Integer	企业联系人的唯一 ID。
lastname	String	企业联系人的姓氏。
firstname	String	企业联系人的名字。
company	String	企业联系人就职的公司的名字。
email	String	企业联系人的邮箱地址。
businessnum	String	企业联系人的办公号码。
businessnum2	String	企业联系人的其它办公号码。
mobile	String	企业联系人的手机号码。
mobile2	String	企业联系人的其它手机号码。
homenum	String	企业联系人的家庭号码。
homenum2	String	企业联系人的其它家庭号码。
businessfax	String	企业联系人的公司传真号码。
homefax	String	企业联系人的家庭传真号码。
othernum	String	企业联系人的其它号码。
zipcode	String	企业联系人所在城市的邮政编码。
street	String	企业联系人居住的街道。
city	String	企业联系人居住的城市。

参数	类型	描述
state	String	企业联系人居住的州。
country	String	企业联系人居住的国家。

示例

请求示例

查询 id 为12 的企业联系人的信息。

```
POST /
api/v2.0.0/companycontacts/query?token=42e4fc1a0d093ec973a5bcfb28424fcb
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "12"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "companycontacts": [
  {
   "id": "12",
   "firstname": "Maggie",
   "lastname": "Chou",
   "email": "",
   "company": "",
   "businessnum": "15880123456",
   "businessnum2": "",
   "businessfax": "",
   "mobile": "",
   "mobile2": "",
   "homenum": "",
   "homenum2": "",
```

```
    "homefax": "",
    "othernum": "",
    "zipcode": "",
    "street": "",
    "city": "",
    "state": "",
    "country": ""
   }
  ]
}
```

添加企业联系人

添加并配置企业联系人信息。

请求地址

```
{base_url}/{api_path}/companycontacts/add?token={token}
```

请求参数

> **注：**
> 必填参数：`lastname`、`firstname`、至少一个联系号码。

参数	是否必选	类型	描述
lastname	是	String	企业联系人的姓氏。
firstname	是	String	企业联系人的名字。
company	否	String	企业联系人就职的公司的名字。
email	否	String	企业联系人的邮箱地址。
businessnum	否	String	企业联系人的办公号码。
businessnum2	否	String	企业联系人的其它办公号码。
mobile	否	String	企业联系人的手机号码。
mobile2	否	String	企业联系人的其它手机号码。
homenum	否	String	企业联系人的家庭号码。
homenum2	否	String	企业联系人的其它家庭号码。

参数	是否必选	类型	描述
businessfax	否	String	企业联系人的公司传真号码。
homefax	否	String	企业联系人的家庭传真号码。
othernum	否	String	企业联系人的其它号码。
zipcode	否	String	企业联系人所在城市的邮政编码。
street	否	String	企业联系人居住的街道。
city	否	String	企业联系人居住的城市。
state	否	String	企业联系人居住的州。
country	否	String	企业联系人居住的国家。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
id	Integer	企业联系人的唯一 ID。

示例

请求示例

请求添加企业联系人 Maggie Chou，办公号码为 15880123456。

```
POST /api/v2.0.0/companycontacts/add?token=02c893a131633e96771eb9af3b14f0e2
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "firstname": "Maggie",
 "lastname": "Chou",
 "businessnum": "15880123456"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
```

```
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "id": "12"
}
```

修改企业联系人

修改企业联系人的信息，如邮箱地址、手机号码等。

请求地址

```
{base_url}/{api_path}/companycontacts/update?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	企业联系人的唯一 ID。
lastname	否	String	企业联系人的姓氏。
firstname	否	String	企业联系人的名字。
company	否	String	企业联系人就职的公司的名字。
email	否	String	企业联系人的邮箱地址。
businessnum	否	String	企业联系人的办公号码。
businessnum2	否	String	企业联系人的其它办公号码。
mobile	否	String	企业联系人的手机号码。
mobile2	否	String	企业联系人的其它手机号码。
homenum	否	String	企业联系人的家庭号码。
homenum2	否	String	企业联系人的其它家庭号码。
businessfax	否	String	企业联系人的公司传真号码。
homefax	否	String	企业联系人的家庭传真号码。
othernum	否	String	企业联系人的其它号码。
zipcode	否	String	企业联系人所在城市的邮政编码。
street	否	String	企业联系人居住的街道。

参数	是否必选	类型	描述
city	否	String	企业联系人居住的城市。
state	否	String	企业联系人居住的州。
country	否	String	企业联系人居住的国家。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

请求将 id 为13 的企业联系人的邮箱设置为 maggie@yeastar.com。

```
POST /
api/v2.0.0/companycontacts/update?token=42e4fc1a0d093ec973a5bcfb28424fcb
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "13",
 "email": "maggie@yeastar.com"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success"
}
```

删除企业联系人

删除单个企业联系人、多个企业联系人或所有企业联系人。

请求地址

```
{base_url}/{api_path}/companycontacts/delete?token={token}
```

请求参数

参数	是否必选	Type	Description
id	是	Integer	企业联系人的唯一 ID。 **取值范围**： • 单个企业联系人 id：删除指定企业联系人。 例如：`"id":"9"` • 多个企业联系人 id：删除多个企业联系人。多个企业联系人 id 用半角逗号（,）隔开。 例如：`"id":"8,9"` • all：删除所有企业联系人。 例如：`"id":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除 id 为13 的企业联系人。

```
POST /
api/v2.0.0/companycontacts/delete?token=42e4fc1a0d093ec973a5bcfb28424fcb
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
```

```
 "id": "13"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

分机

查询分机列表

查询 PBX 上分机列表的基本信息，如：分机号码、分机状态、分机类型、分机名称等。

请求地址

```
{base_url}/{api_path}/extension/list?token={token}
```

请求参数

无参数，直接发送查询分机列表的请求即可。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
extlist	Array<ExtensionInfo>	分机信息列表。

ExtensionInfo

参数	类型	描述
number	String	分机号。
status	String	分机当前状态。 • Unavailable：不可用。 • Registered：已注册。 • Ringing：响铃。 • Busy：忙线。 • Hold：通话保持。 • Malfunction：故障。 • Idle：空闲。
type	String	分机类型。 • SIP
username	String	用户名。
agentid	String	用户呼入队列后，系统播报的坐席工号。此参数默认为空，表示播报分机号作为坐席的工号。 **提示：** 调用 extension/update 接口可以编辑分机的 `agentid`。

示例

请求示例

```
POST /api/v2.0.0/extension/list?token=7dff09fe45414a4f340e978e274b53ea
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "extlist": [
```

```
  {
   "number": "1022",
   "status": "Idle",
   "type": "SIP",
   "username": "1022"
  },
  {
   "number": "1023",
   "status": "Registered",
   "type": "SIP",
   "username": "1023"
  }
 ]
}
```

查询分机详情

查询单个分机、多个分机或所有分机的详细配置。

请求地址

```
{base_url}/{api_path}/extension/query?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：查询指定分机。 例如：`"number":"1000"` • 多个分机号码：查询多个分机。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1000,1001"` • all：查询所有分机。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
extinfos	Array<ExtensionInfo>	分机信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
username	String	用户名。
status	String	分机当前状态。 • Unavailable：不可用。 • Registered：已注册。 • Ringing：响铃。 • Busy：忙线。 • Hold：通话保持。 • Malfunction：故障。 • Idle：空闲。
type	String	分机类型。
callerid	String	来电显示号码。
registername	String	注册名称。
registerpassword	String	注册密码。
maxregistrations	Integer	同时注册数。
loginpassword	String	用户密码。
email	String	邮箱地址。
mobile	Integer	手机号码。
language	String	提示音语言。
hasvoicemail	String	是否启用 **启用语音邮箱** 功能。 • on：开启。 • off：关闭。
enablevmtoemail	String	是否启用 **发送语音留言到邮箱** 功能。

参数	类型	描述
		• off：关闭。 • on：开启。发送语音留言到分机用户邮箱。 • custom：发送语音留言到自定义邮箱。
vmsecret	Integer	语音信箱密码。
alwaysforward	String	是否启用 **总是呼叫转移** 功能。 • on：开启。 • off：关闭。
atransferto	String	`alwaysforward` 的目的地。 • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。
atransferext	Integer	`atransferto` 为 `Extension` 时，设置分机的号码。
atransferprefix	Integer	`atransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。
atransfernum	String	`atransferto` 为 `Custom Number` 时，设置呼出号码。
noanswerforward	String	是否启用 **无应答呼叫转移** 功能。 • on：开启。 • off：关闭。
ntransferto	String	`noanswerforward` 的目的地。 • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。
ntransferext	Integer	`ntransferto` 为 `Extension` 时，设置分机的号码。
ntransferprefix	String	`ntransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。
ntransfernum	String	`ntransferto` 为 `Custom Number` 时，设置呼出号码。
busyforward	String	是否启用 **忙时呼叫转移** 功能。

参数	类型	描述
		• on：开启。 • off：关闭。
btransferto	String	`busyforward` 的目的地。 • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。
btransferext	Integer	`btransferto` 为 `Extension` 时，设置分机的号码。
btransferprefix	String	`btransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。
btransfernum	String	`btransferto` 为 `Custom Number` 时，设置呼出号码。
allowbeingmonitored	String	是否启用 **允许被监听** 功能。 • on：开启。 • off：关闭。
monitormode	String	监听模式。 • Disabled：禁用。 • Extensive：通用模式。 • Listen：普通监听。 • Whisper：密语监听。 • Barge-in：强插监听。
ringtimeout	Integer	响铃超时。
maxduration	String	最大通话时长。
dnd	String	是否启用 **免打扰** 功能。 • on：开启。 • off：关闭。
callrestriction	String	是否启用 **外呼限制** 功能。 • on：开启。 • off：关闭。
agentid	String	用户呼入队列后，系统播报的坐席工号。此参数默认为空，表示播报分机号作为坐席的工号。 **提示：**

参数	类型	描述
		调用 extension/update 接口可以编辑分机的 `agentid`。
unselectoutroute	String	分机不可用的呼出路由。
selectoutroute	String	分机可用的呼出路由。

示例

请求示例

查询分机1001的详细配置。

```
POST /api/v2.0.0/extension/query?token=309bcf2d729fc93e20eafc803a23b76d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "number": "1001"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "extinfos": [
  {
   "number": "1001",
   "username": "1001",
   "status": "Registered",
   "type": "SIP",
   "callerid": "1001",
   "registername": "1001",

  "registerpassword": "f2844e41123368345574515f8eaf6595a0a05e50c9d7d1669062
2ed2ea6965f0",
   "maxregistrations": "1",
```

```
  "loginpassword": "81d3f187d07d54e2e48b217b62e8436a36d1eb75c5845d4509ac2d1
4bdeebd97107ab5bbd60c3d8cbfbd0f205b4b655a",
   "email": "ann@yeastar.com",
   "moblie": "18508937881",
   "language": "System Default",
   "hasvoicemail": "on",
   "enablevmtoemail": "off",
   "vmsecret": "1001",
   "alwaysforward": "off",
   "noanswerforward": "on",
   "ntransferto": "Voicemail",
   "ntransferprefix": "9",
   "busyforward": "on",
   "btransferto": "Voicemail",
   "btransferprefix": "9",
   "ringsimultaneous": "off",
   "mobileprefix": "9",
   "enablemobile": "off",
   "allowbeingmonitored": "off",
   "monitormode": "Disabled",
   "ringtimeout": "30",
   "maxduration": "Follow System",
   "dnd": "off",
   "callrestriction": "off",
   "agentid": "none",
   "unselectoutroute": "local_calls",
   "selectoutroute": "routeout"
  }
 ]
}
```


查询分机通话状态

查询单个分机、多个分机或所有分机的通话状态。

请求地址

```
{base_url}/{api_path}/extension/query_call?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：查询指定分机。 例如：`"number":"1000"` • 多个分机号码：查询多个分机。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1000,1001"` • all：查询所有分机。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
calllist	Array <ExtesnionInfo>	分机通话列表。

ExtensionInfo

参数	类型	描述
number	String	分机号。
numbercalls	Array <CallStatus>	分机通话信息。

CallStatus

参数	类型	描述
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	Integer	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过 IVR、队列、响铃组、广播组或会议室与分机建立通话时，才会携带 callpath。 例如：分机 1000 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。

参数	类型	描述
		• ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过中继、IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。 例如：外线 2203303 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼出。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过 IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。

参数	类型	描述
		例如：使用 IVR （6500）拨打外线号码 5503302，callpath 为 6500。

示例

响应示例

```
POST /
api/v2.0.0/extension/query_call?token=4dfa215a2a835361425878c43e25dc8d
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "number": "1000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success",
    "calllist": [
        {
            "number": "1084",
            "numbercalls": [
                {
                    "callid": "1593507412.76",
                    "members": [
                        {
                            "inbound": {
                                "from": "31622",
                                "to": "1084",
                                "trunkname": "sps-3.183",
                                "channelid":
 "PJSIP/trunk-sps-3.183-endpoint-00000034",
                                "callpath": "1000"
```

```
                            }
                        },
                        {
                            "ext": {
                                "number": "1084",
                                "channelid": "PJSIP/1084-00000035"
                            }
                        }
                    ]
                }
            ]
        }
    ]
}
```

添加分机

添加并设置分机。

请求地址

```
{base_url}/{api_path}/extension/add?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **注：** • 只允许输入数字。 • 最大长度为7位。
username	是	String	显示名称。 **注：** 不允许输入以下字符：`!` `%` `.` `@` `:` `;` `&` `"` `'` `\` `<` `>` `` ` `` `$`
registername	是	String	注册名称。 **注：**

参数	是否必选	类型	描述
			• 只允许输入字母、数字。 • 最大长度为31位。
registerpassword	是	String	注册密码。 **注：** 只允许输入字母、数字和以下字符：`~` `^` `*` `-` `_` `?` 默认使用强密码。 • 必须包含数字（非连续）、大小写字母等。 • 密码长度大于16位。 • 不得包含该分机的“分机号码”、“显示名称”。
callerid	否	String	显示号码。 **注：** • 只允许输入字母、数字。 • 最大长度31位。 **默认值**：同分机号码。
maxregistrations	否	Integer	同时注册数。 **取值范围**：1~4 **默认值**：1
loginpassword	否	String	用户密码。 **注：** 只允许输入字母、数字和以下字符：`~` `^` `*` `-` `_` `?` 默认使用强密码。 • 必须包含数字（非连续）、大小写字母等。 • 密码长度大于10位。

参数	是否必选	类型	描述
			• 不得包含该分机的“分机号码”、“显示名称”。
email	否	String	邮箱地址。 **注：** • 须符合邮箱地址格式。 • 不允许输入空格和以下字符：`#` `;` `,` `[` `]` `=` `&` `"` `'` `\` `<>` `` ` `` `$`
mobile	否	Integer	手机号码。 **注：** • 设置呼叫转移目的地为 **用户手机** 时，此项不为空。 • 只允许输入数字。 • 最大长度31位。
hasvoicemail	否	String	是否启用 **启用语音邮箱** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：on。
vmsecret	否	Integer	语音信箱密码。 **注：** • 只允许输入数字。 • 最大长度63位。 **默认值**：同分机号码。
enablevmtoemail	否	String	是否启用 **发送语音留言到邮箱** 功能。 **取值范围**：

参数	是否必选	类型	描述
			• off：关闭。 • on：开启。发送语音留言到分机用户邮箱。 • custom：发送语音留言到自定义邮箱。 **默认值**：off。
alwaysforward	否	String	是否启用 **总是呼叫转移** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：off。
atransferto	否	String	`alwaysforward` 的目的地。 **取值范围**： • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。
atransferext	否	Integer	`atransferto` 为 `Extension` 时，设置分机的号码。
atransferprefix	否	Integer	`atransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** • 只允许输入数字。 • 最大长度7位。
atransfernum	否	Integer	`atransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** • 只允许输入数字。 • 最大长度15位。
noanswerforward	否	String	是否启用 **无应答呼叫转移** 功能。 **取值范围**：

参数	是否必选	类型	描述
			• on：开启。 • off：关闭。 **默认值**：on。
ntransferto	否	String	`noanswerforward` 的目的地。 **取值范围**： • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。 **默认值**：Voicemail。
ntransferext	否	Integer	`ntransferto` 为 `Extension` 时，设置分机的号码。
ntransferprefix	否	Integer	`ntransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** • 只允许输入数字。 • 最大长度7位。
ntransfernum	否	Integer	`ntransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** • 只允许输入数字。 • 最大长度15位。
busyforward	否	String	是否启用 **忙时呼叫转移** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：on。
btransferto	否	String	`busyforward` 的目的地。 **取值范围**：

参数	是否必选	类型	描述
			• Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。 **默认值**：Voicemail。
btransferext	否	Integer	`btransferto` 为 `Extension` 时，设置分机的号码。
btransferprefix	否	Integer	`btransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** • 只允许输入数字。 • 最大长度7位。
btransfernum	否	Integer	`btransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** • 只允许输入数字。 • 最大长度15位。
allowbeingmonitored	否	String	是否启用 **允许被监听** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：off。
monitormode	否	String	监听模式。 **取值范围**： • Disabled：禁用。 • Extensive：通用模式。 • Listen：普通监听。 • Whisper：密语监听。 • Barge-in：强插监听。 **默认值**：Disabled。
ringtimeout	否	Integer	响铃超时时间。单位：秒。

参数	是否必选	类型	描述
			默认值：30。
maxduration	否	String	最大通话时长。单位：秒。 **默认值**：Follow System。
dnd	否	String	是否启用 **免打扰** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：off。
callrestriction	否	String	是否启用 **外呼限制** 功能。 **取值范围**： • on：开启。 • off：关闭。 **默认值**：off。
agentid	否	String	用户呼入队列后，系统播报的坐席工号。此参数默认为空，表示播报分机号作为坐席的工号。 **提示：** 调用 extension/update 接口可以编辑分机的 agentid。
selectoutroute	否	String	选择分机可用的呼出路由。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

添加分机5000。

```
POST /api/v2.0.0/extension/add?token=721eebe15cb3c27b60daeb1f120f0c05
```

```
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "5000",
 "username": "5000",
 "registername": "5000",
 "registerpassword": "v0S*ZrB6Y3g~1-oe"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

修改分机

修改分机的配置，如：分机名称、同时注册数、邮箱地址等。

请求地址

```
{base_url}/{api_path}/extension/update?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **注：** • 只允许输入数字。 • 最大长度为7位。
username	否	String	显示名称。

参数	是否必选	类型	描述
			注： 不允许输入以下字符：`!` `%` `.` `@` `:` `;` `&` `"` `'` `\` `<` `>` `` ` `` `$`
registername	否	String	注册名称。 **注：** • 只允许输入字母、数字。 • 最大长度为31位。
registerpassword	否	String	注册密码。 **注：** 只允许输入字母、数字和以下字符：`~` `^` `*` `-` `_` `?` 默认使用强密码。 • 必须包含数字（非连续）、大小写字母等。 • 密码长度大于16位。 • 不得包含该分机的“分机号码”、“显示名称”。
callerid	否	String	显示号码。 **注：** • 只允许输入字母、数字。 • 最大长度31位。
maxregistrations	否	Integer	同时注册数。 **取值范围**： • 1 • 2 • 3 • 4
loginpassword	否	String	用户密码。

参数	是否必选	类型	描述
			注： 只允许输入字母、数字和以下字符：`~` `^` `*` `-` `_` `?` 默认使用强密码。 • 必须包含数字（非连续）、大小写字母等。 • 密码长度大于10位。 • 不得包含该分机的“分机号码”、“显示名称”。
email	否	String	邮箱地址。 **注：** • 须符合邮箱地址格式。 • 不允许输入空格和以下字符：`#` `;` `,` `[` `]` `=` `&` `"` `'` `\` `<>` `` ` `` `$`
mobile	否	Integer	手机号码。 **注：** • 设置呼叫转移目的地为 **用户手机** 时，此项不为空。 • 只允许输入数字。 • 最大长度31位。
hasvoicemail	否	String	是否启用 **启用语音邮箱** 功能。 **取值范围**： • on：开启。 • off：关闭。
vmsecret	否	Integer	语音信箱密码。 **注：** • 只允许输入数字。

参数	是否必选	类型	描述
			・最大长度63位。
enablevmtoemail	否	String	是否启用 **发送语音留言到邮箱** 功能。 **取值范围**： ・off：关闭。 ・on：开启。发送语音留言到分机用户邮箱。 ・custom：发送语音留言到自定义邮箱。
alwaysforward	否	String	是否启用 **总是呼叫转移** 功能。 **取值范围**： ・on：开启。 ・off：关闭。
atransferto	否	String	`alwaysforward` 的目的地。 **取值范围**： ・Voicemail：语音信箱。 ・Extension：分机。 ・Mobile Number：用户手机。 ・Custom Number：自定义号码。
atransferext	否	Integer	`atransferto` 为 `Extension` 时，设置分机的号码。
atransferprefix	否	Integer	`atransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** ・只允许输入数字。 ・最大长度7位。
atransfernum	否	Integer	`atransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** ・只允许输入数字。 ・最大长度15位。
noanswerforward	否	String	是否启用 **无应答呼叫转移** 功能。

参数	是否必选	类型	描述
			取值范围： • on：开启。 • off：关闭。
ntransferto	否	String	`noanswerforward` 的目的地。 **取值范围**： • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。
ntransferext	否	Integer	`ntransferto` 为 `Extension` 时，设置分机的号码。
ntransferprefix	否	Integer	`ntransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** • 只允许输入数字。 • 最大长度7位。
ntransfernum	否	Integer	`ntransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** • 只允许输入数字。 • 最大长度15位。
busyforward	否	String	是否启用 **忙时呼叫转移** 功能。 **取值范围**： • on：开启。 • off：关闭。
btransferto	否	String	`busyforward` 的目的地。 **取值范围**： • Voicemail：语音信箱。 • Extension：分机。 • Mobile Number：用户手机。 • Custom Number：自定义号码。

参数	是否必选	类型	描述
btransferext	否	Integer	`btransferto` 为 `Extension` 时，设置分机的号码。
btransferprefix	否	Integer	`btransferto` 为 `Mobile Number` 或 `Custom Number` 时的呼出前缀。 **注：** • 只允许输入数字。 • 最大长度7位。
btransfernum	否	Integer	`btransferto` 为 `Custom Number` 时，设置呼出号码。 **注：** • 只允许输入数字。 • 最大长度15位。
allowbeingmonitored	否	String	是否启用 **允许被监听** 功能。 **取值范围**： • on：开启。 • off：关闭。
monitormode	否	String	监听模式。 **取值范围**： • Disabled：禁用。 • Extensive：通用模式。 • Listen：普通监听。 • Whisper：密语监听。 • Barge-in：强插监听。
ringtimeout	否	Integer	响铃超时时间。单位：秒。
maxduration	否	String	最大通话时长。单位：秒。
dnd	否	String	是否启用 **免打扰** 功能。 **取值范围**： • on：开启。 • off：关闭。
callrestriction	否	String	是否启用 **外呼限制** 功能。 **取值范围**：

参数	是否必选	类型	描述
			• on：开启。 • off：关闭。
agentid	否	String	用户呼入队列后，系统播报的坐席工号。此参数默认为空，表示播报分机号作为坐席的工号。 **提示：** 调用 extension/update 接口可以编辑分机的 `agentid`。
selectoutroute	否	String	选择分机可用的呼出路由。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

将分机1002的用户名修改为 Amy，其他信息不变。

```
POST /api/v2.0.0/extension/update?token=7afa377107feb9566654d5dd3a9a2a93
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "1002",
 "username": "Amy"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
```

```
{
    "status": "Success"
}
```

删除分机

删除单个分机、多个分机或所有分机。

请求地址

```
{base_url}/{api_path}/extension/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：删除指定分机。 例如：`"number":"4000"` • 多个分机号码：删除多个分机。多个分机号码用半角逗号（,）隔开。 例如：`"number":"4000,4001"` • all：删除所有分机。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除分机1000。

```
POST /api/v2.0.0/extension/delete?token=7afa377107feb9566654d5dd3a9a2a93
Content-Type:application/json; charset=utf-8
```

```
Content-Length:37
Host: api.yeastarcloud.com

{
 "number": "1000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

分机组

查询分机组详情

查询单个分机组、多个分机组或所有分机组的详细配置。

请求地址

```
{base_url}/{api_path}/extensiongroup/query?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	分机组的唯一 ID。 **取值范围**: • 单个分机组 id：查询指定分机组的信息。 例如：`"id":"13"`

参数	是否必选	类型	描述
			• 多个分机组 id：查询多个分机组信息。多个分机组用半角逗号（,）隔开。 例如：`"id":"13,14"` • `all`：查询所有分机组信息。 例如：`"id":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
extgroupinfos	Array <Extgroupinfos>	分机组信息列表。

Extgroupinfos

参数	类型	描述
id	Integer	分机组的唯一 ID。
name	String	分机组名称。
members	String	分机组成员。

示例

请求示例

查询 id 为 13 的分机组信息。

```
POST /api/v2.0.0/
extensiongroup/query?token=5b470331116bb15d16c24940083d9dfc HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "id": "13"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
```

```
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success",
  "extgroupinfos":
      [
        {
           "id":"13",
           "name":"training team",
           "members":"1008,1009,1010,1011,1012"
         }
       ]
}
```

添加分机组

添加并设置分机组。

请求地址

```
{base_url}/{api_path}/extensiongroup/add?token={token}
```

请求参数

参数	是否必选	类型	描述
name	是	String	分机组名称。
members	是	String	分机组成员 。 **注：** 多个成员用半角逗号（,）隔开。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

参数	类型	描述
id	Integer	分机组的唯一 ID。 分机组添加成功后，系统会为分机组自动分配一个 ID。
errornum	String	该分机号码不存在。

示例

请求示例 1

添加一个分机组，分机组名称为 “sale”。

```
POST /api/v2.0.0/extensiongroup/add?token=45e3ddb103930c8d006a768d46e2af7b
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "name": "sale",
  "members": "1000,1001,1002"
 }
```

响应示例 1

分机组 “sale” 创建成功。

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success",
  "id":"7"
 }
```

请求示例 2

添加一个分机组，分机组名称为 “test”。

```
POST /api/v2.0.0/extensiongroup/add?token=45e3ddb103930c8d006a768d46e2af7b
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "name": "test",
  "members": "1004,1005,1055"
```

```
 }
```

响应示例 2

分机号码 “1055” 不存在，应用服务器返回错误的分机号码。

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success",
  "id":"8",
  "errnum":"1055"
 }
```

修改分机组

修改分机组的配置，如分机组名称或分机组成员。

请求地址

```
{base_url}/{api_path}/extensiongroup/update?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	分机组的唯一 ID。
name	否	String	分机组名称。
members	否	String	分机组成员。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

修改 id 为 8 的分机组的名称和成员。

```
POST /
api/v2.0.0/extensiongroup/update?token=6b4b4bc214498e9a55af60411393a31f
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "id":"8",
  "name": "yeastar",
  "members": "1000,1001,1011,1017,1018"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success"
 }
```

删除分机组

删除单个分机组、多个分机组或所有分机组。

请求地址

```
{base_url}/{api_path}/extensiongroup/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	分机组的唯一 ID。 **取值范围**:

参数	是否必选	类型	描述
			• 单个分机组 id：删除指定的分机组。 例如：`"id":"19"` • 多个分机组 id：删除多个分机组。多个分机组 id 用半角逗号（,）隔开。 例如：`"id":"18,19"` • `all`：删除所有分机组。 例如：`"id":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除 id 为 11 的分机组。

```
POST /api/v2.0.0/
extensiongroup/delete?token=6b4b4bc214498e9a55af60411393a31f HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "id": "11"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success"
 }
```

中继

查询中继列表

查询 PBX 上所有中继的基本信息，如：中继名称、中继状态、中继类型等。

请求地址

```
{base_url}/{api_path}/trunk/list?token={token}
```

请求参数

无参数，直接发送查询中继列表的请求即可。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
trunklist	Array<TrunkInfo>	中继信息列表。

TrunkInfo

参数	类型	描述
trunkname	String	中继名称。
id	Integer	中继的唯一 ID。
type	String	中继类型。
status	String	中继当前状态。 • Registering：注册中。 • Failure：注册失败。 • Registered：已注册。 • Disable：禁用。 • Unknown：未知。

示例

请求示例

```
POST /api/v2.0.0/trunk/list?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "trunklist": [
  {
   "trunkname": "siptrunk",
   "id": "7",
   "status": "registered",
   "type": "SIP"
  }
 ]
}
```

查询 SIP 中继详情

查询单个 SIP 中继、多个 SIP 中继或所有 SIP 中继的详细配置。

请求地址

```
{base_url}/{api_path}/trunk/query_siptrunk?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	SIP 中继的唯一 ID。 **取值范围**: • 单个 SIP 中继 id：查询指定中继的配置。 例如：`"id":"15"`

参数	是否必选	类型	描述
			• 多个 SIP 中继 id：查询多个 SIP 中继的配置。多个 SIP 中继 id 用半角逗号（,）隔开。 例如：`"id":"15,16"` • all：查询所有 SIP 中继的配置。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
trunks	Array<TrunkInfo>	中继信息列表。

TrunkInfo

支持三种中继：注册中继信息、点对点中继信息和账号中继信息。

表 1. 注册中继信息

参数	类型	描述
id	Integer	中继的唯一 ID。
trunkname	String	中继名称。
trunktype	String	中继类型。 register：注册中继。
host	String	SIP 运营商的域名或 IP 地址。
port	String	中继的 SIP 端口。
domain	String	SIP 运营商的域名或 IP 地址。
username	String	SIP 运营商提供的 SIP 账号。
authname	String	SIP 运营商提供的 SIP 账号。
fromuser	String	From 头域。 **注：** 如果 SIP 运营商不支持，则放空。
password	String	SIP 账号的密码。
extensionsdod	String	分机绑定的 DOD 号码。

表 1. 注册中继信息 (续)

参数	类型	描述

		• **绑定单个 DOD 号码**:
		格式:
		`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
		示例:
		`"extensionsdod":"5503301-yeastar-1000"`
		• **绑定多个 DOD 号码**:
		格式:
		`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
		多个 DOD 号码用半角逗号（,）隔开。
		示例:
		`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	String	分机组绑定的 DOD 号码。
		• **绑定单个 DOD 号码**:
		格式:
		`"extensionsgroupdod": "{dod_number}-{dod_name}-{extension_group_name}"`
		示例:
		`"extensionsgroupdod":"5503301-yeastar-sales"`
		• **绑定多个 DOD 号码**:
		格式:
		`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`

表 1. 注册中继信息 (续)

参数	类型	描述
		多个 DOD 号码用半角逗号（,）隔开。 **示例**: `"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

表 2. 点对点中继信息

参数	类型	描述
id	Integer	中继的唯一 ID。
trunkname	String	中继名称。
trunktype	String	中继类型。 peer：点对点中继。
host	String	SIP 运营商的域名或 IP 地址。
port	String	中继的 SIP 端口。
domain	String	SIP 运营商的域名或 IP 地址。

extensionsdod	String	分机绑定的 DOD 号码。
		• **绑定单个 DOD 号码**:
		格式:
		`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
		示例:
		`"extensionsdod":"5503301-yeastar-1000"`

extensionsgroupdod	String	分机组绑定的 DOD 号码。 • **绑定单个 DOD 号码**: **格式**: `"extensionsgroupdod": "{dod_number}-{dod_name}-{extension_group_name}"`

表 2. 点对点中继信息 (续)

参数	类型	描述
		示例： `"extensionsgroupdod":"5503301-yeastar-sales"`

表 3. 账号中继信息

参数	类型	描述
id	Integer	中继的唯一 ID。
trunkname	String	中继名称。
trunktype	String	中继类型。 account：账号中继。
username	String	账号中继的用户名。
authname	String	账号中继的认证名。
password	String	账号中继的密码。

extensionsdod	String	分机绑定的 DOD 号码。
		• **绑定单个 DOD 号码**：
		格式：
		`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
		示例：
		`"extensionsdod":"5503301-yeastar-1000"`

extensionsgroupdod	String	分机组绑定的 DOD 号码。
		• **绑定单个 DOD 号码**：
		格式：
		`"extensionsgroupdod": "{dod_number}-{dod_name}-{extension_group_name}"`
		示例：
		`"extensionsgroupdod":"5503301-yeastar-sales"`

示例

请求示例

查询 id 为15的中继的详细信息。

```
POST /api/v2.0.0/
trunk/query_siptrunk?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "id": "15"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "trunks": [
  {
   "id": "15",
   "trunkname": "test",
   "trunktype": "peer",
   "host": "192.168.6.36",
   "port": "5060",
   "domain": "192.168.6.36",
   "extensionsdod": "",
   "extensionsgroupdod": "5503301-yeastar-sales,5503302-yealink-support,"
  }
 ]
}
```

添加 SIP 中继

添加并设置 SIP 中继。

请求地址

```
{base_url}/{api_path}/trunk/add_siptrunk?token={token}
```

请求参数

根据不同类型的中继，需要发送不同的请求参数。

- 添加注册中继
- 添加点对点中继
- 添加账号中继

添加注册中继

参数	是否必选	类型	描述
trunkname	是	String	中继名称。
trunktype	是	String	中继类型。 **取值**： register：注册中继。
host	是	String	SIP 运营商的域名或 IP 地址。
port	是	String	中继的 SIP 端口。 **默认值**：5060。
domain	是	String	SIP 运营商的域名或 IP 地址。
username	是	String	SIP 运营商提供的 SIP 账号。
authname	是	String	SIP 运营商提供的 SIP 账号。
fromuser	否	String	From 头域。 **注：** 如果 SIP 运营商不支持，则放空。
didsetting	是	String	与该中继关联的 DID 号码和 DNIS 名称。 **格式**： `"didsetting":"{dnis_name}-{did_number}"` 多个 DID 号码用半角逗号（,）隔开。 **示例**：

参数	是否必选	类型	描述
			`"didsetting":"1005name-1001005,1006name-1001006"`
password	是	String	SIP 账号的密码。

extensionsdod	否	String	分机绑定的 DOD 号码。
			取值范围:
			• 单个 DOD 号码:
			格式:
			`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
			示例:
			`"extensionsdod":"5503301-yeastar-1000"`
			• 多个 DOD 号码:
			格式:
			`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例:
			`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	否	String	分机组绑定的 DOD 号码。 **取值范围**: • 单个 DOD 号码: **格式**: `"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}` **示例**:

参数	是否必选	类型	描述

			`"extensionsgroupdod":"5503301-yeastar-sales"`
			• 多个 DOD 号码：
			格式：
			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

添加点对点中继

参数	类型	描述
trunkname	String	中继名称。
trunktype	String	中继类型。 **取值**： peer：点对点中继。
host	String	SIP 运营商的域名或 IP 地址。
port	String	中继的 SIP 端口。 **默认值**：5060。
domain	String	SIP 运营商的域名或 IP 地址。
extensionsdod	String	分机绑定的 DOD 号码。 **取值范围**： • 单个 DOD 号码： **格式**： `"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"` **示例**：

参数	类型	描述

		`"extensionsdod":"5503301-yeastar-1000"`
		• 多个 DOD 号码：
		格式：
		`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
		多个 DOD 号码用半角逗号（,）隔开。
		示例：
		`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	String	分机组绑定的 DOD 号码。
		取值范围：
		• 单个 DOD 号码：
		格式：
		`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
		示例：
		`"extensionsgroupdod":"5503301-yeastar-sales"`
		• 多个 DOD 号码：
		格式：
		`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
		多个 DOD 号码用半角逗号（,）隔开。
		示例：
		`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

添加账号中继

参数	类型	描述
trunkname	String	中继名称。

参数	类型	描述
trunktype	String	中继类型。 **取值**： account：账号中继。
username	String	账号中继的用户名。
authname	String	账号中继的认证名。
password	String	账号中继的密码。

extensionsdod	String	分机绑定的 DOD 号码。
		取值范围：
		• 单个 DOD 号码：
		格式：
		`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
		示例：
		`"extensionsdod":"5503301-yeastar-1000"`
		• 多个 DOD 号码：
		格式：
		`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
		多个 DOD 号码用半角逗号（,）隔开。
		示例：
		`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	String	分机组绑定的 DOD 号码。 **取值范围**： • 单个 DOD 号码： **格式**： `"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}` **示例**：

参数	类型	描述

		`"extensionsgroupdod":"5503301-yeastar-sales"`
		• 多个 DOD 号码：
		格式：
		`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
		多个 DOD 号码用半角逗号（,）隔开。
		示例：
		`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

响应参数

参数	类型	描述
id	Integer	中继的唯一 ID。
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

添加一个注册类型的中继，中继名称为“Trunk1”。

```
POST /api/v2.0.0/trunk/add_siptrunk?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "trunkname": "Trunk1",
    "trunktype": "register",
    "host": "192.168.12.161",
    "port": "5060",
    "domain": "192.168.12.161",
    "username": "1001",
    "authname": "1001",
    "didsetting":"1005name-10011005,1006name-10011006",
```

```
    "password": "Pajsjx9183",
}
```

响应示例

中继添加成功后，系统会为中继自动分配一个 id。

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "id": "10",
 "status": "Success"
}
```

修改 SIP 中继

修改 SIP 中继的基本设置和 DOD 号码设置。

请求地址

```
{base_url}/{api_path}/trunk/update_siptrunk?token={token}
```

请求参数

根据不同类型的中继，需要发送不同的请求参数。

- 修改注册中继
- 修改点对点中继
- 修改账号中继

修改注册中继

参数	是否必选	类型	描述
id	是	Integer	中继的唯一 ID。

参数	是否必选	类型	描述
trunkname	否	String	中继名称。
trunktype	是	String	**取值**： register：注册中继。
host	否	String	SIP 运营商的域名或 IP 地址。
port	否	String	中继的 SIP 端口。
domain	否	String	SIP 运营商的域名或 IP 地址。
username	否	String	SIP 运营商提供的 SIP 账号。
authname	否	String	SIP 运营商提供的 SIP 账号。
fromuser	否	String	From 头域。 **注：** 如果 SIP 运营商不支持，则放空。
password	否	String	SIP 账号的密码。

extensionsdod	否	String	分机绑定的 DOD 号码。
			取值范围：
			• 单个 DOD 号码：
			格式：
			`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
			示例：
			`"extensionsdod":"5503301-yeastar-1000"`
			• 多个 DOD 号码：
			格式：

参数	是否必选	类型	描述

			`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	否	String	分机组绑定的 DOD 号码。
			取值范围：
			• 单个 DOD 号码：
			格式：
			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales"`
			• 多个 DOD 号码：
			格式：
			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

修改点对点中继

参数	是否必选	类型	描述
id	是	Integer	中继的唯一 ID。
trunkname	否	String	中继名称。
trunktype	是	String	中继类型。

参数	是否必选	类型	描述
			取值： peer：点对点中继。
host	否	String	SIP 运营商的域名或 IP 地址。
port	否	String	中继的 SIP 端口。
domain	否	String	SIP 运营商的域名或 IP 地址。

extensionsdod	否	String	分机绑定的 DOD 号码。
			取值范围：
			• 单个 DOD 号码：
			格式：
			`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
			示例：
			`"extensionsdod":"5503301-yeastar-1000"`
			• 多个 DOD 号码：
			格式：
			`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	否	String	分机组绑定的 DOD 号码。 **取值范围**： • 单个 DOD 号码： **格式**：

参数	是否必选	类型	描述

			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}"`
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales"`
			• 多个 DOD 号码：
			格式：
			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

修改账号中继

参数	是否必选	类型	描述
id	是	Integer	中继的唯一 ID。
trunkname	否	String	中继名称。
trunktype	是	String	中继类型。 **取值**： account：账号中继。
username	否	String	账号中继的用户名。
authname	否	String	账号中继的认证名。
password	否	String	账号中继的密码。
extensionsdod	否	String	分机绑定的 DOD 号码。

参数	是否必选	类型	描述

			取值范围:
			• 单个 DOD 号码:
			格式:
			`"extensionsdod": "{dod_number}-{dod_name}-{extension_number}"`
			示例:
			`"extensionsdod":"5503301-yeastar-1000"`
			• 多个 DOD 号码:
			格式:
			`"extensionsdod":"{dod_number}-{dod_name}-{extension_number}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例:
			`"extensionsdod":"5503301-yeastar-1000,5503302-yeastar-1001"`

extensionsgroupdod	否	String	分机组绑定的 DOD 号码。
			取值范围:
			• 单个 DOD 号码:
			格式:
			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}`
			示例:
			`"extensionsgroupdod":"5503301-yeastar-sales"`
			• 多个 DOD 号码:
			格式:

参数	是否必选	类型	描述

			`"extensionsgroupdod":"{dod_number}-{dod_name}-{extension_group_name}"`
			多个 DOD 号码用半角逗号（,）隔开。
			示例：
			`"extensionsgroupdod":"5503301-yeastar-sales,5503302-yeastar-support"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/
trunk/update_siptrunk?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "id":"8",
    "trunkname": "siptrunk1",
    "trunktype": "register",
    "host": "192.168.12.161",
    "port": "5060",
    "domain": "192.168.12.161",
    "username": "1001",
    "authname": "1001",
    "password": "Pajsjx9183",
}
```

响应示例

```
HTTP/1.1 200 OK
```

```
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

删除 SIP 中继

删除单个 SIP 中继、多个 SIP 中继或所有 SIP 中继。

请求地址

```
{base_url}/{api_path}/trunk/delete_siptrunk?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	SIP 中继的唯一 ID。 **取值范围**： • 单个 SIP 中继 id：删除指定 SIP 中继。 例如：`"id":"16"` • 多个 SIP 中继 id：删除多个 SIP 中继。多个 SIP 中继 id 用半角逗号（,）隔开。 例如：`"id":"15,16"` • all：删除所有 SIP 中继。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除 id 为16 的 SIP 中继。

```
POST /api/v2.0.0/
trunk/delete_siptrunk?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "16"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success"
}
```

呼入路由

查询呼入路由

查询单个呼入路由、多个呼入路由或所有呼入路由的设置。

请求地址

```
{base_url}/{api_path}/inroute/query?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼入路由的唯一 ID。

参数	是否必选	类型	描述
			取值范围： • 单个呼入路由 id：查询指定呼入路由的设置。 例如：`"id":"5"` • 多个呼入路由 id：查询多个呼入路由的设置。多个呼入路由用半角逗号（,）隔开。 例如：`"id":"5,6"` • all：查询所有呼入路由的设置。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
inroutes	Array<InrouteInfo>	呼入路由信息。

InrouteInfo

参数	类型	描述
id	Integer	呼入路由的唯一 ID。
name	String	呼入路由的名称。
did	String	DID 匹配模式。
caller_id	String	来电匹配模式。
trunks	String	中继。
desttype	String	呼入目的地类型。 • hangup：挂机。 • extension：分机。 • extrange：分机范围。 • voicemail：语音邮箱。 • ivr：IVR。 • ringgroup：响铃组。 • queue：队列。 • conference：会议室。 • disa：DISA。 • callback：回拨。 • outroute：呼出路由。

参数	类型	描述
		• faxtoemail：传真到邮件。
dest	String	具体目的地。

示例

请求示例

查询 id 为5的呼入路由的详细信息。

```
POST /api/v2.0.0/inroute/query?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "5"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "inroutes": [
  {
   "id": "5",
   "name": "Route_in",
   "did": "4000",
   "caller_id": "3000",
   "trunks": "To_6.36",
   "desttype": "extension",
   "dest": "4000"
  }
 ]
}
```

添加呼入路由

添加并设置呼入路由。

请求地址

```
{base_url}/{api_path}/inroute/add?token={token}
```

请求参数

参数	是否必选	类型	描述
name	是	String	呼入路由的名称。
did	否	String	DID 匹配模式。
caller_id	否	String	来电匹配模式。
trunks	是	String	中继名称。多个中继名称用半角逗号（,）隔开。 例如：`"trunks":"trunk1,trunk2"`
desttype	是	String	目的地类型。 **取值范围**： • hangup：挂机。 • extension：分机。 • extrange：分机范围。 • voicemail：语音邮箱。 • ivr：IVR。 • ringgroup：响铃组。 • queue：队列。 • conference：会议室。 • disa：DISA。 • callback：回拨。 • outroute：呼出路由。 • faxtoemail：传真到邮件。
dest	是	String	具体目的地。

响应参数

参数	类型	描述
id	Integer	呼入路由的唯一 ID。
status	String	请求结果。 • Success：成功。

参数	类型	描述
		• Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/inroute/add?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "name": "trunk1",
 "did": "5503301",
 "caller_id": "3000",
 "trunks": "sipTrunk",
 "desttype": "hangup"
}
```

响应示例

呼入路由添加成功后，系统会为呼入路由自动分配一个 id。

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "id": "8",
 "status": "Success"
}
```

修改呼入路由

修改呼入路由的设置。

请求地址

```
{base_url}/{api_path}/inroute/update?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼入路由的唯一 ID。
name	否	String	呼入路由的名称。
did	否	String	DID 匹配模式。
caller_id	否	String	来电匹配模式。
trunks	否	String	中继名称。多个中继名称用半角逗号（,）隔开。 例如：`"trunks":"trunk1,trunk2"`
desttype	否	String	目的地类型。 **取值范围**： • hangup：挂机。 • extension：分机。 • extrange：分机范围。 • voicemail：语音邮箱。 • ivr：IVR。 • ringgroup：响铃组。 • queue：队列。 • conference：会议室。 • disa：DISA。 • callback：回拨。 • outroute：呼出路由。 • faxtoemail：传真到邮件。
dest	否	String	具体目的地。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/inroute/update?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

```
{
 "id": "8",
 "name": "trunk1",
 "trunks": "Siptrunk",
 "desttype": "hangup"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

删除呼入路由

删除单个呼入路由、多个呼入路由或所有呼入路由。

请求地址

```
{base_url}/{api_path}/inroute/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼入路由的唯一 ID。 **取值范围**： • 单个呼入路由 id：删除指定呼入路由。 例如：`"id":"15"` • 多个呼入路由 id：删除多个呼入路由。多个呼入路由 id 用半角逗号（,）隔开。 例如：`"id":"15,16"` • all：删除所有呼入路由。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除 id 为8的呼入路由。

```
POST /api/v2.0.0/inroute/delete?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "8"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

呼出路由

查询呼出路由

查询单个呼出路由、多个呼出路由或所有呼出路由的设置。

请求地址

```
{base_url}/{api_path}/outroute/query?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼出路由的唯一 ID。 **取值范围**： • 单个呼出路由 id：查询指定呼出路由的设置。 例如：`"id":"1"` • 多个呼出路由 id：查询多个呼出路由的设置。多个呼出路由 id 用半角逗号（,）隔开。 例如：`"id":"1,2"` • all：查询所有呼出路由的设置。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
outroutes	Array<OutrouteInfo>	呼出路由信息。

OutrouteInfo

参数	类型	描述
id	Integer	呼出路由的唯一 ID。
name	String	呼出路由的名称。
prefix	String	呼出模式。
strip	Integer	删除前缀位数。
prepend	String	前置号码。
trunks	String	中继名称。
extensions	Integer	分机号码。
extensionsgroup	String	分机组。

示例

请求示例

查询 id 为1的呼出路由的详细信息。

```
POST /api/v2.0.0/outroute/query?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

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

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "outroutes": [
  {
   "id": "1",
   "name": "Route_out",
   "prefix": "159XXXXXXXX",
   "strip": "",
   "prepend": "",
   "trunks": "siptrunk",
   "extensions": "1020,1001,1000",
   "extensionsgroup": ""
  }
 ]
}
```

添加呼出路由

添加并设置呼出路由。

请求地址

```
{base_url}/{api_path}/outroute/add?token={token}
```

请求参数

参数	是否必选	类型	描述
name	是	String	呼出路由的名称。
prefix	是	String	呼出模式。
strip	否	Integer	删除前缀位数。
prepend	否	String	前置号码。
trunks	是	String	中继名称。多个中继名称之间用半角逗号（,）隔开。 例如：`"trunks":"trunk1,trunk2"`
extensions	是	Integer	允许使用该呼出路由的分机。多个分机用半角逗号（,）隔开。 例如：`"extensions":"1001,1002"`
extensionsgroup	否	String	允许使用该呼出路由的分机组。多个分机组用半角逗号（,）隔开。 例如：`"extensionsgroup":"sales,support"`

响应参数

参数	类型	描述
id	Integer	呼出路由的唯一 ID。
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/outroute/add?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "name": "outroute1",
```

```
    "prefix": "0",
    "strip": "1",
    "trunks": "siptrunk",
    "extensions": "1001",
    "extensionsgroup": "sales,support"
}
```

响应示例

呼出路由添加成功后，系统会为呼出路由自动分配一个 id。

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "id": "8",
 "status": "Success"
}
```

修改呼出路由

修改呼出路由的设置。

请求地址

```
{base_url}/{api_path}/outroute/update?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼出路由的唯一 ID。
name	否	String	呼出路由的名称。
prefix	否	String	呼出模式。
strip	否	Integer	删除前缀位数。
prepend	否	String	前置号码。
trunks	否	String	中继名称。多个中继名称用半角逗号（,）隔开。

参数	是否必选	类型	描述
			例如：`"trunks":"trunk1,trunk2"`
extensions	否	Integer	允许使用该呼出路由的分机。多个分机用半角逗号（,）隔开。 例如：`"extensions":"1001,1002"`
extensionsgroup	否	String	允许使用该呼出路由的分机组。多个分机组用半角逗号（,）隔开。 例如：`"extensionsgroup":"sales,support"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/outroute/update?token=277ac400357b509b4a587ff2157f7ad5
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "id": "8",
    "name": "outroute2",
    "prefix": "0",
    "strip": "1",
    "trunks": "siptrunk",
    "extensions": "1003",
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
```

```
{
    "status": "Success"
}
```

删除呼出路由

删除单个呼出路由、多个呼出路由或所有呼出路由。

请求地址

```
{base_url}/{api_path}/outroute/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
id	是	Integer	呼出路由的唯一 ID。 **取值范围**： • 单个呼出路由 id：删除指定呼出路由。 例如：`"id":"9"` • 多个呼出路由 id：删除多个呼出路由。多个呼出路由 id 用半角逗号（,）隔开。 例如：`"id":"8,9"` • all：删除所有呼出路由。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除 id 为9的呼出路由。

```
POST /api/v2.0.0/outroute/delete?token=1e3b3ebb6a974cb42ed31de5413df52d
```

```
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "id": "9"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

会议室

查询固定会议室

查询 PBX 网页已创建的会议室配置详情。支持查询单个、多个或所有会议室。

请求地址

```
{base_url}/{api_path}/conference/query?token={token}
```

请求参数

参数	是否必选	类型	说明
number	是	String	固定会议室号码。 **取值范围**： • 单个会议室号码：查询指定会议室。 例如：`"number":"6400"` • 多个会议室号码：查询多个固定会议室。多个会议室号码用半角逗号（,）隔开。 例如：`"number":"6400,6401"`

参数	是否必选	类型	说明
			• all：查询所有固定会议室。

响应参数

参数	类型	说明
status	String	请求结果。 • Success：成功。 • Failed：失败。
conferences	Array <ConferenceInfo>	会议室信息列表。

ConferenceInfo

参数	类型	说明
number	String	会议室号码。
name	String	会议室名称。
participantepassword	String	与会者密码。
moderatorpassword	String	主持人密码。
waitformoderator	String	是否等候主持人进入后才开始会议。 • yes：是。 • no：否。
allowinvite	String	是否允许与会者邀请其他成员。 • yes：是。 • no：否。
moderators	Integer	会议室主持人。
musiconhold	String	是否启用等待音乐。 • yes：是。 • no：否。
mute	String	是否在会议室普通成员进入会议室时静音成员。 • yes：是。 • no：否。
prompt	String	成员进入、退出会议室的提示音。

参数	类型	说明
		• default：默认。成员进入或退出会议室时，所有与会者会听到一声短“嘟”提示音。 • exten：分机号码。成员进入或退出会议室时，所有与会者会听到主叫号码。 • none：禁用。成员进入或退出会议时，所有与会者都不会听到任何提示音。

示例

请求示例

查询固定会议室6401的配置详情。

```
POST /api/v2.0.0/conference/query?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6401"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success",
    "conferences": [
        {
            "number": "6401",
            "name": "PM",
            "participantepassword": "3201",
            "moderatorpassword": "",
            "waitformoderator": "no",
            "allowinvite": "yes",
            "moderators": "",
            "musiconhold": "yes",
```

```
            "mute": "no",
            "prompt": "default"
        }
    ]
}
```

查询临时会议室

查询通过 API 接口 `instant_conference/create` 调用的临时会议室。支持查询单个、多个或所有会议室。

请求地址

```
{base_url}/{api_path}/instant_conference/query?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	临时会议室号码。 **取值范围**： • 单个会议室号码：查询指定临时会议室。 例如：`"number":"6450"` • 多个会议室号码：查询多个临时会议室。多个会议室号码用半角逗号（,）隔开。 例如：`"number":"6450,6451"` • all：查询所有临时会议室。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
conferences	Array <ConferenceInfo>	会议室信息列表。

ConferenceInfo

参数	类型	描述
number	String	会议室号码。
name	String	会议室名称。
moderators	Integer	会议室主持人。
member	String	与会成员信息。

示例

请求示例

查询临时会议室6466的信息。

```
POST /api/v2.0.0/
instant_conference/query?token=3ddc40cf73c8ed477e4e466b044e9c89
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6466"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "conferences": [
  {
   "number": "6466",
   "name": " 6466",
   "moderators": "4000",
   "members": "4001"
  }
 ]
}
```

查询正在进行的会议

查询单个、多个或所有正在进行的会议室信息。支持查询固定会议室和临时会议室。

请求地址

```
{base_url}/{api_path}/conference/query_in_session?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	会话中的会议室号码。 **取值范围**： • 单个会议室号码：查询指定的会话中的会议室。 例如：`"number":"6400"` • 多个会议室号码：查询多个会话中的会议室。多个会议室号码用半角逗号（,）隔开。 例如：`"number":"6400,6401"` • all：查询所有会话中的会议室。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
conference	Array <ConferenceInfo>	会议室信息列表。

ConferenceInfo

参数	类型	描述
callid	String	每个通话的唯一 ID。
conferencenumber	Integer	会议室号码。
conferencename	String	会议室名称。
members	Array <MemberType>	与会成员信息。

MemberType

参数	类型	描述
outbound	Array <OutboundInfo>	外线呼出的信息。
inbound	Array <InboundInfo>	外线号码呼入会议室。
ext	Array <ExtensionInfo>	分机的信息。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	String	被叫号码。
trunkname	String	通话使用的中继名称。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	会议室成员的状态。 • ANSWER：该成员接受邀请，进入会议室。 • ANSWERED：该成员拨打会议室号码，主动进入会议室。 • HOLD：该成员已进入会议室，等待主持人进入会议室。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	String	被叫号码。
trunkname	String	通话使用的中继名称。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	会议室成员的状态。 • ANSWER：该成员接受邀请，进入会议室。 • ANSWERED：该成员拨打会议室号码，主动进入会议室。 • HOLD：该成员已进入会议室，等待主持人进入会议室。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	会议室成员的状态。 • ANSWER：该成员接受邀请，进入会议室。 • ANSWERED：该成员拨打会议室号码，主动进入会议室。 • HOLD：该成员已进入会议室，等待主持人进入会议室。

示例

请求示例

查询会话中的会议室6410的详细信息。

```
POST /api/v2.0.0/
conference/query_in_session?token=3ddc40cf73c8ed477e4e466b044e9c89
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6410"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "conference": [
  {
   "callid": "1573465840.181",
   "conferencenumber": "6400",
   "conferencename": "6400",
   "members": [
    {
```

```
      "ext": {
       "number": "4003",
       "channelid": "PJSIP/4003-00000036",
       "memberstatus": "ANSWERED"
      }
     },
     {
      "ext": {
       "number": "4001",
       "channelid": "PJSIP/4001-00000035",
       "memberstatus": "ANSWERED"
      }
     },
     {
      "inbound": {
       "from": "3000",
       "to": "6400",
       "trunkname": "To6.36",
       "channelid": "PJSIP/trunk-To6.36-endpoint-00000037",
       "memberstatus": "ANSWERED"
      }
     }
    ]
   }
  ]
}
```


修改固定会议室

修改 PBX 网页已创建的会议室的配置。

请求地址

```
{base_url}/{api_path}/conference/update?token={token}
```

请求参数

参数	是否必选	类型	说明
number	是	String	固定会议室号码。
name	否	String	固定会议室名称。 **注：**

参数	是否必选	类型	说明
			不允许输入以下字符：`&'<>`|$]\"`。
participantepassword	否	String	与会者密码。 **注：** 只允许输入数字和 `*`。
moderatorpassword	否	String	主持人密码。 **注：** 只允许输入数字和 `*`。
waitformoderator	否	String	是否等候主持人进入后才开始会议。 **取值范围**： • yes：是。 • no：否。
allowinvite	否	String	是否允许与会者邀请其他成员。 **取值范围**： • yes：是。 • no：否。
moderators	否	Integer	会议室主持人。 **注：** 主持人只能是内部分机号码。
musiconhold	否	String	是否启用等待音乐。 **取值范围**： • yes：是。 • no：否。
mute	否	String	是否在会议室普通成员进入会议室时静音成员。 **取值范围**： • yes：是。 • no：否。
prompt	否	String	成员进入、退出会议室的提示音。 **取值范围**：

参数	是否必选	类型	说明
			• default：默认。成员进入或退出会议室时，所有与会者会听到一声短“嘟”提示音。 • exten：分机号码。成员进入或退出会议室时，所有与会者会听到主叫号码。 • none：禁用。成员进入或退出会议时，所有与会者都不会听到任何提示音。

响应参数

参数	类型	说明
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

修改会议室6400的与会者密码为“5678”。

```
POST /api/v2.0.0/conference/update?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6400",
 "participantepassword": "5678"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

发起临时会议

通过此接口发起临时会议，即时邀请联系人加入会议。

请求地址

```
{base_url}/{api_path}/instant_conference/start?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	临时会议室号码。 **注：** 该号码必须处于空闲状态，即其他固定会议室、临时会议室或其他功能未占用此号码。
name	是	String	临时会议室名称。
moderators	是	Integer	临时会议室主持人。 **取值范围：** • 单个分机号码：仅设置一个主持人。 • 多个分机号码：设置多个主持人。多个分机号码用半角逗号（,）隔开。 **注：** 要成功建立临时会议室，至少有一个主持人要进入会议室。
members	是	String	临时会议室成员。 **取值：** 支持填写内部分机号码和外部号码，多个号码用半角逗号（,）隔开。
dialpermission	是	String	使用哪个分机的呼叫权限，邀请会议室成员。 **注：** 如果邀请的成员为外部号码，必须填写具备外呼权限的分机号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/
instant_conference/start?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6466",
 "name": "6466",
 "members": "4000,4001",
 "moderators": "4000",
 "dialpermission": "4000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

邀请会议室成员

邀请内部成员或外部成员加入会议室。

请求地址

```
{base_url}/{api_path}/conference/add_member?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	会议室号码。表示邀请成员进入哪个会议室。
operator	是	Integer	操作者号码。表示谁来执行邀请成员的动作。 **注：** 只有主持人有权限邀请成员。
invitemember	是	Integer	要邀请的成员的号码。可邀请内部分机用户或外部用户。 **取值范围**： • 单个号码：邀请单个成员。 例如：`"invitemember":"2023"` • 多个号码：邀请多个成员。多个号码用半角逗号（,）隔开。 例如：`"invitemember":"2023,2024"`
dialpermission	是	Integer	使用哪个分机的呼叫权限，邀请会议室成员。 **注：** 如果 `invitemember` 包含外部号码，`dialpermission` 必须为具备外呼权限的分机号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

邀请内部分机4001加入会议室6400。

```
POST /api/v2.0.0/
conference/add_member?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6400",
 "operator": "4000",
 "invitemember": "4001",
 "dialpermission": "4000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

踢除会议室成员

踢除会议室的单个、多个或所有成员。

请求地址

```
{base_url}/{api_path}/conference/kick_member?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	会议室号码。表示要踢除哪个会议室的成员。
operator	是	Integer	操作者号码。表示谁来执行踢除成员的动作。 **注：** 只有主持人有权限踢除成员。

参数	是否必选	类型	描述
membernumber	是	Integer	成员号码。 **取值范围**： • 单个号码：踢除指定成员。 例如：`"membernumber":"4000"` • 多个号码：踢除多个成员。多个号码用半角逗号（,）隔开。 例如：`"membernumber":"4000,4001"` • all：踢除所有成员。 例如：`"membernumber":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

请求从会议室6400踢除分机4001。

```
POST /api/v2.0.0/
conference/kick_member?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6400",
 "operator": "4000",
 "membernumber": "4001"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
```

```
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

静音会议室成员

静音会议室的单个、多个或所有成员。

请求地址

```
{base_url}/{api_path}/conference/mute_member?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	会议室号码。
operator	是	Integer	操作者号码。表示谁来执行静音成员的动作。 **注：** 只有主持人有权限静音成员。
membernumber	是	Integer	成员号码。 **取值范围**： • 单个号码：静音指定成员。 例如：`"membernumber":"4000"` • 多个号码：静音多个成员。多个号码用半角逗号（,）隔开。 例如：`"membernumber":"4000,4001"` • all：静音所有成员。 例如：`"membernumber":"all"`

响应参数

参数	类型	描述
status	String	请求结果。

参数	类型	描述
		• Success：成功。 • Failed：失败。

示例

请求示例

静音会议室6400的成员-分机3000。

```
POST /api/v2.0.0/
conference/mute_member?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6400",
 "operator": "4000",
 "membernumber": "3000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

取消静音会议室成员

取消静音会议室的单个、多个或所有成员。

请求地址

```
{base_url}/{api_path}/conference/unmute_member?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	会议室号码。
operator	是	Integer	操作者号码。表示谁来执行取消静音成员的动作。 **注：** 只有主持人有权限取消静音成员。
membernumber	是	Integer	成员号码。 **取值范围**： • 单个号码：取消静音指定成员。 例如：`"membernumber":"4000"` • 多个号码：取消静音多个成员。多个号码用半角逗号（,）隔开。 例如：`"membernumber":"4000,4001"` • all：取消静音所有成员。 例如：`"membernumber":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

取消静音会议室6400的成员-分机2000。

```
POST /api/v2.0.0/
conference/unmute_member?token=721eebe15cb3c27b60daeb1f120f0c05
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6400",
 "operator": "4000",
 "membernumber": "2000"
```

```
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

IVR

查询 IVR

查询单个 IVR、多个 IVR 或所有 IVR 的详细信息。

请求地址

```
{base_url}/{api_path}/ivr/query?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	IVR 号码。 **取值范围**： • 单个 IVR 号码：查询指定 IVR 的信息。 例如：`"number":"6500"` • 多个 IVR 号码：查询多个 IVR 的信息。多个 IVR 号码用半角逗号（,）隔开。 例如：`"number":"6500,6501,6502"` • all：查询所有 IVR 的信息。 例如：`"number":"all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
ivr	Array <IVRInfo>	IVR 信息列表。

IVRInfo

参数	类型	描述
number	String	IVR 号码。
ivrname	String	IVR 名称。
prompt	String	IVR 提示音。
promptrepeat	Integer	提示音播放次数。
responsetimeout	Integer	响应超时时间。
digittimeout	Integer	按键超时时间。
dialext	String	允许拨打分机号码。 • on：允许。 • off：不允许。
dialoutboundroutes	String	允许从呼出路由拨出。 • on：允许。 • off：不允许。
dialtocheckvoicemail	String	允许查阅语音留言。 • on：允许。 • off：不允许。

示例

请求示例

查询 IVR 6500 的信息。

```
POST /api/v2.0.0/ivr/query?token=1e3b3ebb6a974cb42ed31de5413df52d
```

```
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6500"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "ivr": [
  {
   "number": "6500",
   "ivrname": "6500",
   "prompt": "[Default]",
   "promptrepeat": "3",
   "responsetimeout": "3",
   "digittimeout": "3",
   "dialext": "on",
   "dialoutboundroutes": "off",
   "dialtocheckvoicemail": "off"
  }
 ]
}
```

修改 IVR

修改 IVR 的配置，如：IVR 名称、提示音播放次数等。

请求地址

```
{base_url}/{api_path}/ivr/update?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	IVR 号码。
ivrname	否	String	IVR 名称。 **注：** • 不允许输入空格和以下字符：`! $ ( ) / # ; , [ ] " = < > & ' ` ^ % @ { } |`。 • 最大长度31位。
promptrepeat	否	String	提示音播放次数。 **取值范围**：1~5。
responsetimeout	否	Integer	响应超时时间。单位：秒。 **取值范围**：1~10。
digittimeout	否	Integer	按键超时时间。单位：秒。 **取值范围**：1~10。
dialext	否	String	允许拨打分机号码。 **取值范围**： • on：允许。 • off：不允许。
dialoutboundroutes	否	String	允许从呼出路由拨出。 **取值范围**： • on：允许。 • off：不允许。
dialtocheckvoicemail	否	String	允许查阅语音留言。 **取值范围**： • on：允许。 • off：不允许。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

修改 IVR 6500的名称为“Welcome”。

```
POST /api/v2.0.0/ivr/update?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6500",
 "ivrname": "Welcome"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

队列

查询队列

查询单个队列、多个队列或所有队列的详细信息。

请求地址

```
{base_url}/{api_path}/queue/query?token={token}
```

请求参数

参数	是否必选	类型	说明
number	是	String	队列号码。 **取值范围**： • 单个队列号码：查询指定队列。 例如：`"number":"6700"` • 多个队列号码：查询多个队列的信息。多个队列号码用半角逗号（,）隔开。 例如：`"number":"6700,6701,6702"` • all：查询所有队列。 例如：`"number":"all"`

响应参数

参数	类型	说明
status	String	请求结果。 • Success：成功。 • Failed：失败。
queues	Array <QueueInfo>	队列信息列表。

QueueInfo

参数	类型	说明
number	String	队列号码。
password	String	加入队列动态坐席的密码。
queuename	String	队列名称。
ringstrategy	String	响铃策略。 • Ring All：全部响铃。 • Least Recent：最近最少被叫响铃。 • Fewest Calls：最少接通响铃。 • Random：随机响铃。 • Rrmemory：顺序响铃。

参数	类型	说明
		• Linear：线性响铃。
failoveraction	String	呼入失败目的地。 • Hang up：挂机。 • Extension：分机。 • Voicemail：语音信箱。 • IVR：IVR（自动话务员）。 • Ring Group：响铃组。 • Queue：队列。 • Conference：会议室。 • Fax to Email：传真到邮件。 • Dial by name：通过名字呼叫。
failoverdest	String	呼入失败具体目的地。
agents	String	固定坐席。
dynamicagents	String	动态坐席。
agenttimeout	String	坐席响铃时间。
agentannounce	String	坐席应答提示音。
wrapuptime	Integer	坐席结束与客户的通话后，继续接听新来电的时间间隔。
ringinuse	String	使用中振铃。 • on：开启。 • off：关闭。
retry	Integer	重试间隔时间。
musiconhold	String	等待音乐。
maxwaittime	Integer	最大等待时间。
joinempty	String	无坐席时允许呼入。 • on：开启。 • off：关闭。
disallow_to_join_when_empty	String	无坐席时禁止呼入。 • on：无坐席时禁止呼入。 • off：无坐席时允许呼入。
empty_defined_for_disallow_to_join_when_empty	String	视为无坐席的场景。在此场景下不允许呼叫者呼入队列。

参数	类型	说明
		注： 当所有坐席都签出时，该队列默认处于无坐席状态。 • 1：所有坐席都未注册在线 • 2：所有坐席都处于暂停状态 • 3：所有坐席都在使用中
leavewhenempty	String	无坐席时结束等待。 • on：开启。 • off：关闭。
empty_defined_for_leave_empty	String	视为无坐席的场景。在此场景下强制呼叫者退出队列。 **注：** 当所有坐席都签出时，该队列默认处于无坐席状态。 • 1：所有坐席都未注册在线 • 2：所有坐席都处于暂停状态 • 3：所有坐席都在使用中
joinannounce	String	进入队列提示音。
announcepos	String	公告当前位置。 • on：开启。 • off：关闭。
announcefreq	String	播放用户公告的频率。
announceholdtime	String	公告等待时间。 • on：开启。 • off：关闭。
userannounce	String	系统公告提示音。
userannouncefreq	Integer	播放系统公告的频率。
breakoutkey	String	按键 DTMF。
breakoutaction	String	按键目标。 • Hang up：挂机。 • Extension：分机。

参数	类型	说明
		• Voicemail：语音信箱。 • IVR：自动话务员。 • Ring Group：响铃组。 • Queue：队列。 • Conference：会议室。 • Fax to Email：传真到邮件。 • Dial by Name：通过名字呼叫。
breakoutdest	String	按键目标的最终目的地。
satisfactionsurvey	String	满意度调查提示音。没有设置则默认不播报。此参数默认值为 None。
idannouncement	String	播放坐席工号的提示音。没有设置则默认不播报。此参数默认值为 None。

示例

请求示例

查询队列6700的详细信息。

```
POST /api/v2.0.0/queue/query?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6700"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success",
    "queues": [
        {
            "number": "6700",
            "password": "",
            "queuename": "Support",
```

```
            "ringstrategy": "Ring All",
            "failoveraction": "Extension",
            "failoverdest": "1000",
            "agents": "1000,1001,",
            "agenttimeout": "30",
            "agentannounce": "[None]",
            "wrapuptime": "30",
            "ringinuse": "off",
            "retry": "30",
            "musiconhold": "[None]",
            "maxwaittime": "1800",
            "joinempty": "on",
            "disallow_to_join_when_empty": "off",
            "empty_defined_for_disallow_to_join_when_empty": "1,2,3",
            "leavewhenempty": "off",
            "empty_defined_for_leave_empty": "1,2",
            "joinannounce": "[None]",
            "announcepos": "on",
            "announcefreq": "30",
            "announceholdtime": "on",
            "userannounce": "[None]",
            "userannouncefreq": "60",
            "breakoutkey": "None",
            "satisfactionsurvey": "None",
            "idannouncement": "None"
        }
    ]
}
```

查询队列状态

查询队列的状态。

请求地址

```
{base_url}/{api_path}/queue/query_status?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	队列号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
queues	Array <QueueInfo>	队列信息列表。

QueueInfo

参数	类型	描述
number	String	队列号码。
queuestatus	Array <QueuestatusInfo>	队列状态列表。

QueuestatusInfo

参数	类型	描述
callercount	Integer	当前队列等候人数。
members	Array <MemberInfo>	队列成员信息列表。
callers	Array <CallerInfo>	来电信息。

MemberInfo

参数	类型	描述
agent	Integer	坐席号码。
agentstatus	Integer	坐席状态。 • 0-Unknown：未知 • 1-Not In Use：坐席处于空闲状态 • 2-In Use：坐席处于通话中 • 3-Busy：未知 • 4-Invalid：未知 • 5-Unavailable：坐席分机未注册上 • 6-Ringing：坐席分机正在响铃 • 7-In Use Ringing：坐席通话中收到新来电 • 8-On Hold：坐席保持当前通话 • 9-Paused：坐席被暂停服务，不接收来电业务 • 10-Warp-up time：坐席处于休息状态，不接收来电业务

CallersInfo

参数	类型	描述
callernum	Integer	来电号码。
waittime	String	等待时间。
channelid	String	每个通话的唯一 ID。

示例

请求示例

查询队列 6700 的状态。

```
POST /api/v2.0.0/queue/query_status?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6700"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "queues": [
  {
   "number": "6700",
   "queuestatus": [
    {
     "callercount": "",
     "members": [
      {
       "agent": "1000",
       "agentstatus": "2"
      },
      {
```

```
        "agent": "4000",
        "agentstatus": "6"
      }
     ]
    }
   ],
                        "callers":[
                                {
                                        "callernum": "7000",
                                        "waittime": "0:37",
                                        "channelid": "PJSIP/7000-00000031"
                                }
                                ]
  }
 ]
}
```


修改队列

修改队列的配置，如：队列名称、响铃策略等。

请求地址

```
{base_url}/{api_path}/queue/update?token={token}
```

请求参数

参数	是否必选	类型	说明
number	是	String	队列号码。
queuename	否	String	队列名称。
password	否	String	加入队列动态坐席的密码。
ringstrategy	否	String	响铃策略。 **取值范围**： • Ring All：全部响铃。 • Least Recent：最近最少被叫响铃。 • Fewest Calls：最少接通响铃。 • Random：随机响铃。 • Rrmemory：顺序响铃。 • Linear：线性响铃。
failoveraction	否	String	呼入失败目的地。

参数	是否必选	类型	说明
			取值范围： • Hang up：挂机。 • Extension：分机。 • Voicemail：语音信箱。 • IVR：自动话务员。 • Ring Group：响铃组。 • Queue：队列。 • Conference：会议室。 • Fax to Email：传真到邮件。 • Dial by name：通过名字呼叫。
failoverdest	否	String	呼入失败具体目的地。
agents	否	String	固定座席。
agenttimeout	否	String	坐席响铃时间。单位：秒。
wrapuptime	否	Integer	休息时间。单位：秒。
ringinuse	否	String	使用中振铃。 **取值范围**： • on：开启。 • off：关闭。
retry	否	Integer	重试间隔时间。单位：秒。 **取值范围**： • 10 • 20 • 30 • 40 • 50
maxwaittime	否	Integer	最大等待时间。单位：秒。
joinempty	否	String	无坐席时允许呼入。 **注：** 如果你设置此参数，则不要再设置 `disallow_to_join_when_empty`，因为这两个参数的作用是一样的。 **取值范围**：

参数	是否必选	类型	说明
			• on：开启。 • off：关闭。 **注：** 如果你设置此参数为 `off`，你可以通过配置 `empty_defined_for_disallow_to_join_when_empty` 来定义无坐席的场景。
disallow_to_join_when_empty	否	String	无坐席时禁止呼入。 **注：** 如果你设置此参数，则不要再设置 `joinempty`，因为这两个参数的作用是一样的。 **取值范围**： • on：无坐席时禁止呼入。 **注：** 如果你设置此参数为 `on`，你可以通过配置 `empty_defined_for_disallow_to_join_when_empty` 参数来定义无坐席的场景。 • off：无坐席时允许呼入。
empty_defined_for_disallow_to_join_when_empty	否	String	视为无坐席的场景。在此场景下禁止呼叫者呼入队列。 **取值范围**： • 1：所有坐席都未注册在线。 • 2：所有坐席都处于暂停状态。 • 3：所有坐席都在使用中。 **注：** 如果你未配置此参数，则此参数默认赋值为 `"1,2"`。
leavewhenempty	否	String	无坐席时结束等待。 **取值范围**： • on：开启。

参数	是否必选	类型	说明
			注： 如果你设置此参数为 `on`，你可以通过配置`empty_defined_for_leave_empty`来定义无坐席的场景。 • off：关闭。
empty_defined_for_leave_empty	否	String	视为无坐席的场景。在此场景下强制呼叫者退出队列。 **取值范围**： • 1：所有坐席都未注册在线 • 2：所有坐席都处于暂停状态 • 3：所有坐席都在使用中 **注：** 如果你未配置此参数，则此参数默认赋值为 `"1,2"`。
announcepos	否	String	公告当前位置。 **取值范围**： • on：开启。 • off：关闭。
announcefreq	否	String	播放用户公告的频率。 **取值范围**： • 0 • 15 • 30 • 45 • 60 • 120 • 180 • 240 • 300 • 600 • 1200
announceholdtime	否	Integer	公告等待时间。 **取值范围**： • on：开启。

参数	是否必选	类型	说明
			• off：关闭。
userannounce	否	String	系统公告提示音。
userannouncefreq	否	Integer	播放系统公告的频率。 **取值范围**： • 0 • 15 • 30 • 45 • 60 • 120 • 180 • 240 • 300 • 600 • 1200
breakoutkey	否	String	按键 DTMF。 **取值范围**： • None • 1 • 2 • 3 • 4 • 5 • 6 • 7 • 8 • 9 • * • #
breakoutaction	否	String	按键目标。 **取值范围**： • Hang up：挂机。 • Extension：分机。 • Voicemail：语音信箱。 • IVR：自动话务员。 • Ring Group：响铃组。 • Queue：队列。 • Conference：会议室。 • Fax to Email：传真到邮件。

参数	是否必选	类型	说明
			• Dial by Name：通过名字呼叫。
breakoutdest	否	String	按键目标的最终目的地。
satisfactionsurvey	否	String	满意度调查提示音。没有设置则默认不播报。 **默认值**：None。
idannouncement	否	String	播放坐席工号的提示音。没有设置则默认不播报。 **默认值**：None。

响应参数

参数	类型	说明
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

将队列6700的响铃策略修改为全部响铃。

```
POST /api/v2.0.0/queue/update?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "6700",
 "ringstrategy": "Ring All"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
```

```
}
```

签入队列

将指定动态坐席签入指定队列。

请求地址

```
{base_url}/{api_path}/queue/add_dynamicagent?token={token}
```

请求参数

参数	是否必选	类型	描述
queuenumber	是	Integer	队列号码。
extnumber	是	Integer	动态坐席的号码。
password	否	Integer	加入队列动态坐席的密码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

将坐席1000签入队列6700，队列密码为6700123。

```
POST /api/v2.0.0/
queue/add_dynamicagent?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "queuenumber": "6700",
 "extnumber": "1000",
```

```
 "password": "6700123"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

签出队列

将指定动态坐席签出指定队列。

请求地址

```
{base_url}/{api_path}/queue/delete_dynamicagent?token={token}
```

请求参数

参数	是否必选	类型	描述
queuenumber	是	Integer	队列号码。
extnumber	是	Integer	动态坐席的号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

将坐席1001 签出队列6700。

```
POST /api/v2.0.0/
queue/delete_dynamicagent?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "queuenumber": "6700",
 "extnumber": "1001"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

暂停坐席服务

暂停某个坐席的服务。坐席被暂停之后，将不会接听到呼入到该队列的来电。

请求地址

```
{base_url}/{api_path}/queue/pause_agent?token={token}
```

请求参数

参数	是否必选	类型	描述
queuenumber	是	Integer	队列号码。
extnumber	是	Integer	坐席号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

暂停队列6700中坐席1000的服务。

```
POST /api/v2.0.0/queue/pause_agent?token=1e3b3ebb6a974cb42ed31de5413df52d
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "queuenumber": "6700",
 "extnumber": "1000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

恢复坐席服务

恢复某个坐席的服务。坐席被恢复之后，可以接听到呼入到该队列的来电。

请求地址

```
{base_url}/{api_path}/queue/unpause_agent?token={token}
```

请求参数

参数	是否必选	类型	描述
queuenumber	是	Integer	队列号码。
extnumber	是	Integer	坐席号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

恢复队列6700中坐席1000的服务。

```
POST /api/v2.0.0/queue/unpause_agent?token=1e3b3ebb6a974cb42ed31de5413df52d
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "queuenumber": "6700",
 "extnumber": "1000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
```

```
{
    "status": "Success"
}
```

语音留言

查询语音留言

查询单个分机、多个分机或所有分机的语音留言。

请求地址

```
{base_url}/{api_path}/voicemail/query?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。表示要查询哪个分机的语音留言。 **取值范围**： • 单个分机号码：查询指定分机的语音留言。 例如：`"number":"1000"` • 多个分机号码：查询多个分机的语音留言。多个分机之间用半角逗号（,）隔开。 例如：`"number":"1000,1001"` • all：查询所有分机的语音留言。 例如：`"number":"all"`

响应参数

> **注：**
> 如果查询的分机没有语音留言文件，PBX 只响应请求成功或请求失败的参数。

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

参数	类型	描述
voicemails	Array <VoicemailInfo>	语音留言列表。

VoicemailInfo

参数	类型	描述
number	String	分机号码。
voicemail	Array <ExtensionVoicemail>	指定分机的语音留言列表。

ExtensionVoicemail

参数	类型	描述
voicemailfile	String	语音留言文件名。
messagefrom	String	语音留言是哪个号码发送的。
voicemailstatus	String	语音留言的状态。 • read：已读。 • unread：未读。
voicemaildate	String	收到语音留言的日期和时间。
voicemaillength	String	语音留言的时长。

示例

请求示例

查询分机4000的语音留言。

```
POST /api/v2.0.0/voicemail/query?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "4000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
```

```
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "voicemails": [
  {
   "number": "4000",
   "voicemail": [
    {
     "voicemailfile": "msg0000.wav",
     "messagefrom": "3000",
     "voicemailstatus": "unread",
     "voicemaildate": "2019-10-20 21:41:42",
     "voicemaillength": "77"
    },
    {
     "voicemailfile": "msg0001.wav",
     "messagefrom": "2000",
     "voicemailstatus": "unread",
     "voicemaildate": "2019-10-20 21:49:35",
     "voicemaillength": "18"
    }
   ]
  }
 ]
}
```

删除语音留言

删除分机的语音留言。

请求地址

```
{base_url}/{api_path}/voicemail/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。表示要删除哪个分机的语音留言文件。 **取值范围**:

参数	是否必选	类型	描述
			• 单个分机号码：删除单个分机的语音留言文件。 例如：`"number": "4000"` • 多个分机号码：删除多个分机的语音留言文件。 例如：`"number": "4000,4001"` • all：删除所有分机的语音留言文件。 例如：`"number": "all"`
voicemailfile	是	String	语音留言的文件名称。 **取值范围**： • 单个语音文件名称：删除指定文件。 例如：`"voicemailfile": "msg0000.wav"` • 多个语音文件名称：删除多个文件。多个文件名称用半角逗号（,）隔开。 例如：`"voicemailfile": "msg0000.wav,msg0001.wav"` • all：删除所有语音文件。 例如：`"number":"all"`
voicemailstatus	是	String	语音留言的状态。 **取值范围**： • read：已读。 • unread：未读。 • all：忽略语音留言状态，删除所有语音留言文件。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除分机2023的未读语音留言文件“msg0000.wav”。

```
POST /api/v2.0.0/voicemail/delete?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
```

```
Host: api.yeastarcloud.com

{
 "number": "2023",
 "voicemailfile": "msg0000.wav",
 "voicemailstatus": "unread"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

下载语音留言

下载分机的语音留言文件。

下载步骤

1. 获取语音留言文件的随机串。
2. 使用随机串，组合成语音留言文件的下载链接。

> **注：**
> 随机串的有效时长为30秒，并且每个随机串只能使用一次。

请求地址

```
{base_url}/{api_path}/voicemail/get_random?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。表示要下载哪个分机的语音留言文件。
voicemailfile	是	String	语音留言文件名。
voicemailstatus	是	String	语音留言的状态。 **取值范围**： • read：要下载的语音留言已读，从 read 文件夹获取指定的语音留言文件。 • unread：要下载的语音留言未读，从 unread 文件夹获取指定的语音留言文件。
allowedip	否	String	设置允许下载语音留言的设备 IP 地址。默认不带此参数，表示只能通过应用服务器下载文件。 **取值范围**： • 空值：只能通过应用服务器下载文件。 例如：`"allowedip":" "` • IP 地址：只能通过该 IP 地址或应用服务器下载文件。 例如：`"allowedip":"192.168.5.20"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
number	String	分机号码。表示下载的是哪个分机的语音留言。
voicemailstatus	String	语音留言的状态 • read：已读。 • unread：未读。
voicemailfile	String	语音留言文件的名称。
random	String	语音留言文件的随机串。 使用这个随机串组合成该语音留言文件的下载链接。

示例

请求示例

获取分机1023语音留言文件“msg0000.wav”的随机串。

```
POST /api/v2.0.0/
voicemail/get_random?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8

Host: api.yeastarcloud.com

{
 "number": "1023",
 "voicemailfile": "msg0000.wav",
 "voicemailstatus": "unread"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "number": "1023",
 "voicemailstatus": "unread",
 "voicemailfile": "msg0000.wav",
 "random": "7d48ba0d1fb647346ed81d6b677f7423"
}
```

语音留言下载链接示例

使用随机串（random）组合成语音留言的下载链接。

语音留言下载链接格式：

```
https://{pbx domain}/api/v2.0.0/voicemail/download?
number={number}&voicemailstatus={voicemailstatus}&token={token}&voicemailfile=
{voicemailfile}&random={random}
```

语音留言下载链接示例：

```
https://api.yeastarcloud.com/api/v2.0.0/voicemail/download?
number=1005&voicemailstatus=unread&token=72d6640a05811af51e0279461e5272ce&
voicemailfile=msg0000.wav&random=1635079c6517f8b82f811cb11e5860d8
```

通话记录

下载通话记录

下载指定分机或所有分机的通话记录。

下载步骤

1. 获取通话记录文件的随机串。
2. 使用随机串，组合成通话记录文件的下载链接。

> **注：**
> 随机串的有效时长为30秒，并且每个随机串使用一次。

请求地址

```
{base_url}/{api_path}/cdr/get_random?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。表示要下载哪些分机的通话记录。 **取值范围**： • 单个分机号码：下载指定分机的通话记录。 例如：`"number":"1000"` • 多个分机号码：下载多个分机的通话记录。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1000,1001,1002"` • all：下载所有分机的通话记录。 例如：`"number":"all"`
starttime	是	String	通话记录的起始时间。 时间格式：`yyyy-mm-dd hh:mm:ss`

参数	是否必选	类型	描述
endtime	是	String	通话记录的结束时间。 时间格式：`yyyy-mm-dd hh:mm:ss`
allowedip	否	String	设置允许下载通话记录的设备 IP 地址。默认不带此参数，表示只能通过应用服务器下载文件。 **取值范围**： • 空值：只能通过应用服务器下载文件。 例如：`"allowedip":" "` • IP 地址：只能通过该 IP 地址或应用服务器下载文件。 例如：`"allowedip":"192.168.5.20"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
number	String	分机号码。
starttime	String	通话记录的起始时间。
endtime	String	通话记录的结束时间。
random	String	通话记录文件的随机串。 使用这个随机串组合成该通话录音文件的下载链接。

示例

请求示例

获取分机4000通话记录文件的随机串。

```
POST /api/v2.0.0/cdr/get_random?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "4000",
 "starttime": "2019-10-24 00:00:00",
 "endtime": "2019-10-24 15:59:59"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "number": "4000",
 "starttime": "2019-10-24 00:00:00",
 "endtime": "2019-10-24 15:59:59",
 "random": "7b5a748e176d1ed625793d3a231808dc"
}
```

通话记录下载链接

使用随机串（random）组合成通话记录的下载链接。

通话记录文件的格式为CSV。查看通话记录参数说明。

通话记录下载链接格式：

```
https://{pbx domain}/api/v2.0.0/cdr/download?
number={number}&starttime={starttime}&endtime={endtime}&token={token}&
random={random}
```

通话记录下载链接示例：

```
https://api.yeastarcloud.com/api/v2.0.0/cdr/download?
number=4000&starttime=2019-10-24%2000:00:00&endtime=2019-10-24%2015:59:59&
token=4555b319609ca31410f7b946671fcc69&random=7b0ad48124d720235b05d2435a48b
c30
```

录音

下载录音文件

获取 PBX 中的自动录音文件。

下载步骤

1. 通过NewCdr（通话记录事件）中的 `recording` 参数获取到自动录音文件的名称。
2. 获取自动录音文件的随机串。
3. 使用随机串，组合成录音文件的下载链接。

> **注：**
> 随机串的有效时长为30秒，并且每个随机串只能使用一次。

请求地址

```
{base_url}/{api_path}/recording/get_random?token={token}
```

请求参数

参数	是否必选	类型	描述
recording	是	String	自动录音文件名称。
allowedip	否	String	设置允许下载录音文件的设备 IP 地址。默认不带此参数，表示只能通过应用服务器下载文件。 **取值范围**： • 空值：只能通过应用服务器下载文件。 例如：`"allowedip":" "` • IP 地址：只能通过该 IP 地址或应用服务器下载文件。 例如：`"allowedip":"192.168.5.20"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
recording	String	自动录音文件名称。
random	String	自动录音文件的随机串。 使用这个随机串组合成该录音文件的下载链接。

示例

请求示例

获取录音文件的随机串。

```
POST /api/v2.0.0/
recording/get_random?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "recording": "20191024012719-1571909237.191-4000-4001-Internal.wav"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "recording": "20191024012719-1571909237.191-4000-4001-Internal.wav",
 "random": "2d2f919b742cc62f5e03c77351350f7c"
}
```

自动录音下载链接

使用随机串（random）组合成录音文件的下载链接。

录音下载链接格式：

```
https://{pbx
 domain}/api/v2.0.0/recording/download?recording={recording}&random={random}
&token={token}
```

录音下载链接示例：

```
https://api.yeastarcloud..com/api/v2.0.0/recording/download?
recording=20170901181806-1504261084.7-1001-1003-Internal.wav&random=120732c
546381fb020f17fba676b
0ea0&token=75c5891b32203d0615f9e3753a7cb779
```

酒店应用

查询闹铃

查询单个分机、多个分机或所有分机的闹铃信息。

请求地址

```
{base_url}/{api_path}/wakeupcall/query?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：查询指定分机的闹铃。 例如：`"number":"1000"` • 多个分机号码：查询多个分机的闹铃。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1000,1001"` • all：查询所有分机的闹铃。 例如：`"number":"all"`

响应参数

> **注：**
> 如果查询的分机没有闹铃信息，PBX 只响应请求成功或请求失败的参数。

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
wakeups	Array <WakeupInfo>	闹铃列表。

WakeupInfo

参数	类型	描述
number	String	分机号码。
wakeup	Array <ExtensionWakeup>	分机闹铃信息列表。

ExtensionWakeup

参数	类型	描述
wakeupid	Integer	闹铃的唯一 ID。
time	String	闹铃时间。
type	String	闹铃类型。 • onetime：一次。 • everyday：每天。 • custom：自定义。
weekdays	Integer	`type` 为 `custom` 时，闹铃适用的星期。 • 0 • 1 • 2 • 3 • 4 • 5 • 6 其中，0表示周日，1-6表示周一到周六。
repeats	Integer	闹铃的重复次数。 • 1 • 2 • 3
repeatinterval	Integer	重复响铃间隔时间，以分钟为单位。
prompt	String	闹铃提示音。

示例

请求示例

查询分机5000的闹铃信息。

```
POST /api/v2.0.0/wakeupcall/query?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

```
{
 "number": "5000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "wakeups": [
  {
   "number": "5000",
   "wakeup": [
    {
     "wakeupid": "9",
     "time": "15:00",
     "type": "onetime",
     "prompt": "macroform-cold_day",
     "repeats": "3",
     "repeatinterval": "5"
    }
   ]
  }
 ]
}
```

添加闹铃

为指定分机设置闹铃。

请求地址

```
{base_url}/{api_path}/wakeupcall/create?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：为指定分机添加闹铃。 例如：`"number":"1000"` • 多个分机号码：为多个分机添加同一个闹铃。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1000,1001"`
wakeup	是	Object<ExtensionWakeup>	分机闹铃信息。 **注：** 一次只能添加一个闹铃。如果一次添加多个，则只有第一个生效。

ExtensionWakeup

参数	是否必选	类型	描述
time	是	String	闹铃时间。
type	是	String	闹铃类型。 **取值范围**： • onetime：一次。 当闹铃完成设置的重复次数之后，该条闹铃将被自动删除。 • everyday：每天。 • custom：自定义。 `type` 为 `custom` 时，`weekdays` 不为空。
weekdays	否	Integer	闹铃适用的星期。 **取值范围**：0~6。 其中，0表示周日，1-6表示周一到周六。 **注：**

参数	是否必选	类型	描述
			type 为 custom 时，此项不为空。
repeats	是	Integer	闹铃的重复次数。 **取值范围**：1~3。
repeatinterval	否	String	重复响铃间隔时间，以分钟为单位。 同时设置重复次数和重复间隔时间，话机第一次响铃后，会在间隔时间后再次响铃。
prompt	否	String	闹铃提示音。 此提示音必须为自定义提示音。不加字段则用系统默认的 macroform-cold_day 提示音。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

为分机1000添加闹铃。

```
POST /api/v2.0.0/wakeupcall/create?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "4000",
 "wakeup":
  {
   "time": "11:00",
   "type": "onetime",
   "repeats": "3",
   "repeatinterval": "5"
  }
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

修改闹铃

更改特定分机的闹铃设置。

请求地址

```
{base_url}/{api_path}/wakeupcall/update?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。
wakeupid	是	Integer	闹铃的唯一 ID。
wakeup	是	Object <ExtensionWakeup>	分机闹铃信息。

ExtensionWakeup

参数	是否必选	类型	描述
time	是	String	闹铃时间。
type	是	String	闹铃类型。 **取值范围**： • onetime：一次。 当闹铃完成设置的重复次数之后，该条闹铃将被自动删除。 • everyday：每天。 • custom：自定义。

参数	是否必选	类型	描述
			`type` 为 `custom` 时，`weekdays` 不为空。
weekdays	否	Integer	闹铃适用的星期。 **取值范围**：0~6。 其中，0表示周日，1-6表示周一到周六。 **注：** `type` 为 `custom` 时，此项不为空。
repeats	否	Integer	闹铃的重复次数。 **取值范围**：1~3。
repeatinterval	否	String	重复响铃间隔时间，以分钟为单位。 同时设置重复次数和重复间隔时间，话机第一次响铃后，会在间隔时间后再次响铃。
prompt	否	String	闹铃提示音。 此提示音必须为自定义提示音。不加字段则用系统默认的 macroform-cold_day 提示音。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

更改分机4000的闹铃设置。

```
POST /api/v2.0.0/wakeupcall/update?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "4001",
 "wakeupid": "25",
```

```
 "wakeup":
  {
   "time": "12:00",
   "type": "onetime"
  }
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

删除闹铃

删除指定分机的闹铃。

请求地址

```
{base_url}/{api_path}/wakeupcall/delete?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • 单个分机号码：删除指定分机的闹铃。 例如：`"number":"1005"` • 多个分机号码：删除多个分机的闹铃。多个分机号码用半角逗号（,）隔开。 例如：`"number":"1005,1006"` • all：删除所有分机的闹铃。 例如：`"number":"all"`

参数	是否必选	类型	描述
wakeupid	否	Integer	闹铃的唯一 ID。不带此参数，表示删除指定分机的所有闹铃。 **取值范围**： • 单个闹铃 ID：删除指定分机的指定闹铃。 例如：`"wakeupid":"1"` • 多个闹铃 ID：删除指定分机的多个闹铃。多个闹铃 ID 用半角逗号（,）隔开。 例如：`"wakeupid": "1,2"` • all：删除指定分机的所有闹铃。 例如：`"wakeupid": "all"`

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

删除分机1000的闹铃。

```
POST /api/v2.0.0/wakeupcall/delete?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "1000"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
```

```
{
    "status": "Success"
}
```

Linkus 客户端

启用或禁用 Linkus 客户端

为分机启用或禁用 Linkus 客户端。

请求地址

```
{base_url}/{api_path}/linkus/update_client?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	分机号码。 **取值范围**： • all： 为所有分机启用或禁用 Linkus 客户端。 • 单个分机号码：为指定分机启用或禁用 Linkus 客户端。 • 多个分机号码：为多个分机启用或禁用 Linkus 客户端。分机号码间用半角逗号（,）隔开。
enablelinkus	是	String	Linkus 客户端状态。 **取值范围**： • on: 启用 Linkus 客户端。 • off: 禁用 Linkus 客户端。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
failed	String	无法启用或禁用 Linkus 客户端的分机及错误码。

示例

请求示例

```
POST /api/v2.0.0/
linkus/update_client?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "all",
        "enablelinkus":"on"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status":"Success",
    "failed":"1001:10006"
}
```

查询 Linkus 客户端状态

查询分机的 Linkus 功能处于启用或禁用的状态。

请求地址

```
{base_url}/{api_path}/linkus/query_client?token={token}
```

请求参数

参数	是否必选	类型	描述
number	Yes	String	分机号码。 **取值范围**： • all：查询所有分机的 Linkus 客户端状态。

参数	是否必选	类型	描述
			• 单个分机号码：查询指定分机的 Linkus 客户端状态。 • 多个分机号码：查询多个分机的 Linkus 客户端状态。分机号码间用半角逗号（,）隔开。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
linkusclients	Array <LinkusClientStatusInfo>	Linkus 客户端状态。

LinkusClientStatusInfo

参数	类型	描述
number	String	分机号码。
enablelinkus	String	Linkus 客户端号码： • on: Linkus 客户端已启用。 • off: Linkus 客户端已禁用。

示例

请求示例

```
POST /api/v2.0.0/linkus/query_client?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "1001,1002"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
```

```
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{"status":"Success",
     "linkusclients":[
            {
                "number":"1001",
                "enablelinkus":"on"
            },
           {
                "number":"1002",
                "enablelinkus":"off"
            },
      ]
}
```

发送 Linkus 邮件

向分机用户发送 Linkus 邮件。

请求地址

```
{base_url}/{api_path}/linkus/send_email?token={token}
```

请求参数

参数	是否必选	类型	描述
number	Yes	String	分机号码。 **取值范围**： • all：发送 Linkus 邮件给所有分机用户。 • 单个分机号码：发送 Linkus 邮件给指定的分机用户。 • 多个分机号码：发送 Linkus 邮件给多个分机用户。分机号码间用半角逗号（,）隔开。

响应参数

参数	类型	描述
status	String	请求结果。

参数	类型	描述
		• Success：成功。 • Failed：失败。
unsent	String	无法发送 Linkus 邮件的分机号及错误码。

示例

请求示例

```
POST /api/v2.0.0/linkus/send_email?token=1e3b3ebb6a974cb42ed31de5413df52d
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "all"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
        "status":"Success",
        "unsent":"1001:10004"
}
```

呼叫操控接口

查询通话

查询指定通话或指定类型的通话详情。

请求地址

```
{base_url}/{api_path}/call/query?token={token}
```

请求参数

查询指定通话

参数	是否必选	类型	描述
callid	是	String	每个通话的唯一 ID。

查询指定类型的通话信息

参数	是否必选	类型	描述
type	是	String	查询的通话类型。 **取值范围**： • inbound：查询所有呼入类型的通话。 • outbound：查询所有呼出类型的通话。

响应参数

参数	类型	描述
calls	Array <CallInfo>	通话信息列表。
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。

CallInfo

参数	类型	描述
members	Array <MemberType>	通话中的不同类型成员的信息列表。
callid	String	每个通话的唯一 ID。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过 IVR、队列、响铃组、广播组或会议室与分机建立通话时，才会携带 callpath。 例如：分机 1000 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。

参数	类型	描述
		• BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过中继、IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。 例如：外线 2203303 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	从哪条中继呼出。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过 IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。 例如：使用 IVR （6500）拨打外线号码 5503302，callpath 为 6500。

示例 1

查询指定通话信息。

请求示例

```
POST /api/v2.0.0/call/query?token=4dfa215a2a835361425878c43e25dc8d HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "callid": "1571386069.8"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "Calls": [
  {
  "members": [
   {
   "inbound": {
    "from": "500",
    "to": "1023",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-API-endpoint-00000005",
    "memberstatus": "ANSWERED"
    }
   },
   {
    "ext": {
     "number": "1023",
     "channelid": "PJSIP/1023-00000006",
     "memberstatus": "ANSWER"
    }
   }
   ],
   "callid": "1571386069.8"
  }
 ],
 "status": "Success"
    }
```

示例 2

查询指定类型的通话信息

请求示例

```
POST /api/v2.0.0/call/query?token=4dfa215a2a835361425878c43e25dc8d HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "type": "inbound"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
"Calls": [
  {
   "members": [
    {
     "inbound": {
      "from": "3000",
      "to": "4000",
      "trunkname": "siptrunk",
      "channelid": "PJSIP/trunk-siptrunk-endpoint-000000a2",
      "memberstatus": "ANSWERED"
     }
    },
    {
     "ext": {
      "number": "4000",
      "channelid": "PJSIP/4000-000000a3",
      "memberstatus": "ANSWER"
     }
    }
   ],
   "callid": "1571639442.264"
  },
```

```
  {
   "members": [
    {
     "inbound": {
      "from": "2000",
      "to": "1023",
      "trunkname": "siptrunk",
      "channelid": "PJSIP/trunk-siptrunk-endpoint-00000098",
      "memberstatus": "ANSWERED"
     }
    },
    {
     "ext": {
      "number": "1023",
      "channelid": "PJSIP/1023-00000099",
      "memberstatus": "ANSWER"
     }
    }
   ],
   "callid": "1571639198.248"
  }
 ],
 "status": "Success"
}
```


发起呼叫

通过本接口可以发起不同类型的呼叫。

通过该接口可以实现以下类型的呼叫：

- **分机呼叫**

  呼叫对象包括：分机、IVR、响铃组、队列、广播组（实时广播）、会议室、外部号码。

- **IVR 外呼**：IVR 作为主叫，呼叫外部号码。
- **响铃组外呼**：响铃组作为主叫，呼叫外部号码。
- **队列外呼**：队列作为主叫，呼叫外部号码。
- **外线外呼**：外部号码作为主叫，呼叫另外一个外部号码。

请求地址

```
{base_url}/{api_path}/call/dial?token={token}
```

请求参数

参数	是否必选	类型	描述
caller	是	Integer	主叫号码。
callee	是	Integer	被叫号码。 **注：** 被叫号码为外部号码时，需匹配呼出路由规则。
dialpermission	否	String	设置使用哪个分机号的权限呼出外线号码。 **注：** • 当 `caller` 为分机号码时，无需设置 `dialpermission` 。可直接使用分机权限呼出。 • 当 `caller` 不是分机号码时（例如响铃组或队列），必须设置 `dialpermission` 为具备外呼权限的分机号码，借助分机的权限呼出外部号码。
autoanswer	否	String	当 `caller` 为分机号码时，可以通过本项控制话机自动应答。即话机无需响铃，自动接听电话。 **注：** 该功能仅在 SIP 话机支持的情况下可用。 **取值范围**： • yes：启用自动应答。 • no：禁用自动应答。 **默认值**：no。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

分机 1023 呼叫分机 2000。

```
POST /api/v2.0.0/call/dial?token=55e169f41421df602d41a08c441bc948 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "caller": "1023",
 "callee": "2000",
 "autoanswer": "no"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571620483.18"
}
```

接受来电

通过该接口接听外线来电。

场景

如果你在 PBX 网页启用了中继的来电接受控制，当外线来电通过该中继呼入时，PBX 将发送Invite（来电呼叫请求）到应用服务器，并预留 10 秒时间让应用服务选择接受或拒绝来电。

> **注：**
> 如果应用服务器没有在 10 秒内作出操作，PBX 将接受该来电，并将该来电引导到呼入路由的目的地。

请求地址

```
{base_url}/{api_path}/call/accept_inbound?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/accept_inbound?token=623a2d1c21f69e9f7d78ce1c5ef56da5
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000001f"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
   "status":"Success",
   "callid":"1571621890.58"
}
```

拒接来电

通过本接口拒接外线来电。

场景

如果你在 PBX 网页启用了中继的来电接受控制，当外线来电通过该中继呼入时，PBX 将发送Invite（来电呼叫请求）到应用服务器，并预留 10 秒时间让应用服务选择接受或拒绝来电。

> **注：**
> 如果应用服务器没有在 10 秒内作出操作，PBX 将接受该来电，并将该来电引导到呼入路由的目的地。

请求地址

```
{base_url}/{api_path}/call/refuse_inbound?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。

响应参数

参数	类型	描述
status	String	请求结果。

参数	类型	描述
		• Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/refuse_inbound?token=3c423b4c0e7cab4e164a01523b8ac512
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000029"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status":"Success",
    "callid":"1571623247.69"
}
```

监听通话

如果分机启用了监听和被监听的功能，应用服务器可通过该接口监听该分机的通话。

请求地址

```
{base_url}/{api_path}/call/listen?token={token}
```

请求参数

参数	是否必选	类型	描述
monitor	是	Integer	监听者的分机号。
extension	是	Integer	被监听者的分机号。
type	是	String	监听模式。 **取值范围**： • listen：普通监听。 • whisper：密语监听。 • barge：强插监听。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/listen?token=3c423b4c0e7cab4e164a01523b8ac512
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "monitor": "2023",
    "extension":"1023",
    "type": "listen"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
```

```
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status":"Success",
    "callid":"1571624363.72"
}
```

通话保持

通过本接口保持指定通话。

请求地址

```
{base_url}/{api_path}/call/hold?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/hold?token=3c423b4c0e7cab4e164a01523b8ac512 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
     "channelid": "PJSIP/1023-00000030"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status":"Success",
    "callid":"1571625574.80"
}
```

恢复通话

通过该接口恢复被保持的通话。

请求地址

```
{base_url}/{api_path}/call/unhold?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/unhold?token=3c423b4c0e7cab4e164a01523b8ac512
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000002d"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status":"Success",
    "callid":"1571625300.77"
}
```

通话静音

通过该接口，将通话中的指定成员静音。

请求地址

```
{base_url}/{api_path}/call/mute?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。 **取值范围**：支持多个 `channelid`，多个 `channelid` 间使用半角逗号（,）隔开。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/mute?token=1166cd917bfc7bb4622cc4a636616cd5 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "channelid": "PJSIP/1023-000000a9"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
   "status":"Success",
   "callid":"1571640953.274"
}
```

取消静音

通过该接口，将通话过程中的指定成员取消静音。

请求地址

```
{base_url}/{api_path}/call/unmute?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。 **取值范围**：支持多个 `channelid`，多个 `channelid` 间使用半角逗号（,）隔开。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/unmute?token=1166cd917bfc7bb4622cc4a636616cd5
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "channelid": "PJSIP/1023-000000a9"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
   "status":"Success",
   "callid":"1571640953.274"
}
```

呼叫停泊

通话期间，将当前通话停泊到特定的停泊号中。

请求地址

```
{base_url}/{api_path}/call/callpark?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，被停泊方的通话通道 ID。
parkingslot	否	String	呼叫停泊号码，通话将被停泊到此停泊号中。 **注：** 若未添加此参数，系统将自动分配一个空闲的停泊号。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
parkingslot	String	呼叫停泊号码，通话被停泊到此停泊号中。

示例

请求示例

```
POST /api/v2.0.0/call/callpark?token=0320b5213220f4063479ca2d19c62b56
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "channelid": "PJSIP/1017-00000020",
  "parkingslot":"6900"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success",
  "parkingslot":"6900"
 }
```

取回被停泊的通话

可使用 API 请求 `call/dial` 拨打呼叫停泊号码取回被停泊的通话。

更多信息，参考发起呼叫。

查询被停泊的通话

查询被停泊通话的信息，包括发起停泊的号码，被停泊的号码，呼叫停泊号码，呼叫停泊时间。

请求地址

```
{base_url}/{api_path}/call/callpark_status?token={token}
```

请求参数

无参数，直接发送查询呼叫停泊状态信息列表的请求即可。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。
parkinfos	Array<ParkInfos>	呼叫停泊状态信息列表。

ParkInfors

参数	类型	描述
parker	Integer	发起呼叫停泊请求的分机号码。
parkedparty	Integer	被停泊的分机号码。
parkingslot	Integer	呼叫停泊号码，通话被停泊到此停泊号中。
parktime	String	在取回通话之前通话可停泊的时间。
channelid	String	通话中，被停泊方的通话通道 ID。

示例

请求示例

```
POST /
api/v2.0.0/call/callpark_status?token=28cbf51d30f569485e2e5d3f18c2d331
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
```

响应示例

```
HTTP/1.1 200 OK
Date: Tue, 18 May 2021 07:26:42 GMT
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
   "status": "Success",
   "parkinfos":
        [
          {
            "parker": "1017",
            "parkedparty": "1000",
            "parkingslot": "6900",
            "parktime": "20",
            "channelid": "PJSIP/1000-0000000d"
            }
          ]
}
```

呼叫转移(盲转)

两方通话时，通过本接口将当前通话直接转移给第三方。

请求地址

```
{base_url}/{api_path}/call/transfer?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。
number	是	String	转移通话的目标号码。
dialpermission	否	String	`number` 为外线号码时，需要设置使用哪个分机号的权限将通话转移给外线号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/transfer?token=3507199f6c51460857610e781ea89932
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "channelid": "PJSIP/1023-000000b7",
 "number": "2023"
}
```

响应示例

```
HTTP/1.1 200 OK
```

```
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571646222.312"
}
```

呼叫转移（咨询转）

建立咨询转

两方通话时，通过本接口呼叫第三方，建立咨询转。

请求地址

```
{base_url}/{api_path}/call/attended_transfer?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。
tonumber	是	Integer	咨询转目标号码。
dialpermission	否	String	`tonumber` 为外线号码时，需要设置使用哪个分机号的权限将通话转移给外线号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

建立咨询转，转移电话给号码2023。

请求参数

```
POST /api/v2.0.0/
call/attended_transfer?token=2584a8eb3fe8739c185c534235ee9421 HTTP/1.1
Content-Type:application/json; charset=utf-8

{
 "channelid": "PJSIP/1020-0000003b",
 "tonumber": "2023"
}
```

响应参数

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571983052.150"
}
```

咨询转操作

建立咨询转之后，通过本接口确认转移、取消转移、切换通话方或建立三方通话。

请求地址

```
{base_url}/{api_path}/call/attended_transfer_operate?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。
operate	是	String	执行咨询转操作。 **取值范围**:

参数	是否必选	类型	描述
			• abort：取消转移通话。 • complete：确认执行通话转移。 • threeway：建立三方通话。 • swap：切换通话方。例如：A 与 B 通话过程中，B 发起咨询转到 C，此时 A 的通话被保持。执行 swap 操作，可以将通话切换回 A 与 B。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例1

确认咨询转。

请求示例

```
POST /api/v2.0.0/
call/attended_transfer_operate?token=2584a8eb3fe8739c185c534235ee9421
 HTTP/1.1
Content-Type:application/json; charset=utf-8

{
 "channelid": "PJSIP/1020-0000003b",
 "operate": "complete"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
```

```
 "callid": "1571983052.150"
}
```

示例2

建立三方通话。

请求示例

```
POST /api/v2.0.0/
call/attended_transfer_operate?token=2584a8eb3fe8739c185c534235ee9421
 HTTP/1.1
Content-Type:application/json; charset=utf-8

{
 "channelid": "PJSIP/1020-0000003b",
 "operate": "threeway"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571983052.151"
}
```

建立多方通话

通话过程中，调用此接口邀请新成员加入通话，建立三方通话或多方通话。

请求地址

```
{base_url}/{api_path}/call/add_member?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。
number	是	String	邀请加入通话的目标号码。 **注：** 每次调用接口，只能邀请一个成员。
dialpermission	否	String	`number` 为外线号码时，需要设置使用哪个分机号的权限将通话转移给外线号码。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/add_member?token=1a4d2fb22b1942ca34a3691b0c7234ab
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "channelid": "PJSIP/1023-000000c0",
 "number": "2023"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
```

```
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571646222.312"
}
```

播放语音

通过本接口播放音乐文件给分机、外线号码或广播组。

请求地址

```
{base_url}/{api_path}/call/playprompt?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	播放音乐的目标号码。 **取值范围**： • 分机号。 • 外部号码。 • 广播组号码。
prompt	是	String	播放的语音文件。 **取值范围**： • 播放 1 个语音文件：音频文件名称，如 music1。 • 播放多个语音文件：使用符号 “+” 链接多个文件，如 music1+music2。 **注：** • 音乐文件需要提前上传到 PBX。 • 文件名称必须包含字母，且不得包含符号 “+” 。
count	否	Integer	播放语音文件的次数。

参数	是否必选	类型	描述
dialpermission	否	String	`number` 为外线号码时，需要设置使用哪个分机号的权限将通话转移给外线号码。
autoanswer	否	String	当 `caller` 为分机号码时，可以通过本项控制话机自动应答。即话机无需响铃，自动接听电话。 **注：** 该功能仅在 SIP 话机支持的情况下可用。 **取值范围**： • yes：启用自动应答。 • no：禁用自动应答。 **默认值**：no。
volume	否	Integer	播放音乐的音量。 **取值范围**：0~20。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求参数

```
POST /api/v2.0.0/call/playprompt?token=16cfa49e417009cf1017aa284217f3ff
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "number": "1023",
 "prompt": "music1+music2",
 "count": "2"
```

```
 "autoanswer": "yes",
 "volume": "10"
}
```

响应参数

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571714617.6"
}
```

挂断广播语音

如果系统当前正在向广播组成员播放语音文件，你可以通过本接口挂断广播语音。

请求地址

```
{base_url}/{api_path}/paginggroup/hangup/music?token={token}
```

请求参数

参数	是否必选	类型	描述
number	是	String	广播组号码。

实际示例

请求示例

```
POST /api/v2.0.0/
paginggroup/hangup/music?token=0d184583046ac1e83cc66bd60746c7a4 HTTP/1.1
Content-Type:application/json; charset=utf-8


{
    "number": "6300"
```

```
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
    "status": "Success"
}
```

挂断通话

通话过程中，通过该接口挂断某个通话成员的电话。

请求地址

```
{base_url}/{api_path}/call/hangup?token={token}
```

请求参数

参数	是否必选	类型	描述
channelid	是	String	通话中，每个成员的通话通道 ID。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：请求成功。 • Failed：请求失败。
callid	String	每个通话的唯一 ID。

示例

请求示例

```
POST /api/v2.0.0/call/hangup?token=1a4d2fb22b1942ca34a3691b0c7234ab
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com

{
 "channelid": "PJSIP/1023-000000c0"
}
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.14rc21
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE

{
 "status": "Success",
 "callid": "1571646222.312"
}
```

uaCSTA 呼叫操控接口

uaCSTA 呼叫操控接口概览

Yeastar 云 PBX 支持通过 uaCSTA 控制 IP 话机的通话。

前提条件

要实现对 IP 话机进行 uaCSTA 控制操作，需满足以下前提要求。

- 使用符合 Yeastar uaCSTA 的型号和固件版本要求的 IP 话机。
- 启用 IP 话机的 uaCSTA 功能。可参考以下文章：
  - 如何启用 Yealink 话机的 uaCSTA 功能
  - 如何启用 Fanvil 话机的 uaCSTA 功能
- 启用 PBX 服务器的 uaCSTA 功能（路径：**配置 > PBX > 常规设置 > SIP > 高级 > 启用 uaCSTA 连接**）

支持的 IP 话机

已兼容的 Yealink 话机

Yealink 话机型号	固件版本
SIP-T21P_E2	52.84.0.125 或更高版本
SIP-T21_E2	52.84.0.125 或更高版本
SIP-T23P	44.84.0.125 或更高版本
SIP-T23G	44.84.0.125 或更高版本
SIP-T27G	69.84.0.125 或更高版本
SIP-T29G	46.83.0.120 或更高版本
SIP-T31	124.85.0.15 或更高版本
SIP-T31P	124.85.0.15 或更高版本
SIP-T31G	124.85.0.15 或更高版本
SIP-T33P	124.85.0.15 或更高版本
SIP-T33G	124.85.0.15 或更高版本
SIP-T40P	54.84.0.125 或更高版本
SIP-T40G	76.84.0.125 或更高版本
SIP-T41S	66.84.0.125 或更高版本
SIP-T41P	36.83.0.120 或更高版本
SIP-T42S	66.84.0.125 或更高版本
SIP-T42G	29.83.0.120 或更高版本
SIP-T46S	66.84.0.125 或更高版本
SIP-T46G	28.83.0.120 或更高版本
SIP-T48S	66.84.0.125 或更高版本
SIP-T48G	35.83.0.120 或更高版本
SIP-T41U	108.84.0.119 或更高版本
SIP-T42U	108.84.0.119 或更高版本
SIP-T43U	108.84.0.119 或更高版本
SIP-T46U	108.84.0.119 或更高版本
SIP-T48U	108.84.0.119 或更高版本
SIP-T52S	70.84.0.70 或更高版本
SIP-T54S	70.84.0.70 或更高版本

Yealink 话机型号	固件版本
SIP-T53	95.84.0.125 或更高版本
SIP-T53W	95.84.0.125 或更高版本
SIP-T54W	96.84.0.125 或更高版本
SIP-T57W	97.84.0.125 或更高版本
SIP-T56A	58.83.0.15 或更高版本
SIP-T58A	58.84.0.20 或更高版本
SIP-VP59	91.283.0.40 或更高版本

已兼容的 Fanvil 话机

Fanvil 话机型号	固件版本
X4U	2.0.3 或更高版本
X5U	2.0.3 或更高版本
X5S	2.0.3 或更高版本
X6	2.0.3 或更高版本
X6U	2.0.3 或更高版本
X7	2.0.3 或更高版本
X7C	2.0.3 或更高版本
X210	2.0.3 或更高版本
X210i	2.0.3 或更高版本

uaCSTA - 接受来电

通过 uaCSTA 接听来电。

前提条件

要实现对 IP 话机进行 uaCSTA 控制操作，需满足以下前提要求。

- 使用符合 Yeastar uaCSTA 的型号和固件版本要求的 IP 话机。
- 启用 IP 话机的 uaCSTA 功能。可参考以下文章：
  - 如何启用 Yealink 话机的 uaCSTA 功能
  - 如何启用 Fanvil 话机的 uaCSTA 功能
- 启用 PBX 服务器的 uaCSTA 功能（路径：**配置 > PBX > 常规设置 > SIP > 高级 > 启用 uaCSTA 连接**）

请求地址

```
{base_url}/{api_path}/uacstacall/accept_inbound?token={token}
```

请求参数

参数	是否可选	类型	描述
extnumber	是	String	要控制的分机号。
cstacallid	是	Integer	每个 uaCSTA 通话的唯一 ID。uaCSTA 功能启用后，系统会自动为每个通话分配一个 `cstacallid`。 **注：** 可从 uaCSTA 通话事件中获取 `cstacallid` 。
ipaddress	是	String	分机号注册的 IP 话机的 IP 地址。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/
uacstacall/accept_inbound?token=5d87a5f8771d0d4451afa1dd6035c48a HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "extnumber":"1000",
  "cstacallid":"32784",
  "ipaddress":"192.168.66.70:5060"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
```

```
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success"
 }
```

uaCSTA - 拒接来电

通过 uaCSTA 拒接来电。

前提条件

要实现对 IP 话机进行 uaCSTA 控制操作，需满足以下前提要求。

- 使用符合 Yeastar uaCSTA 的型号和固件版本要求的 IP 话机。
- 启用 IP 话机的 uaCSTA 功能。可参考以下文章：
  - 如何启用 Yealink 话机的 uaCSTA 功能
  - 如何启用 Fanvil 话机的 uaCSTA 功能
- 启用 PBX 服务器的 uaCSTA 功能（路径：**配置 > PBX > 常规设置 > SIP > 高级 > 启用 uaCSTA 连接**）

请求地址

```
{base_url}/{api_path}/uacstacall/refuse_inbound?token={token}
```

请求参数

参数	是否必选	类型	描述
extnumber	是	String	要控制的分机号。
cstacallid	是	Integer	每个 uaCSTA 通话的唯一 ID。uaCSTA 功能启用后，系统会自动为每个通话分配一个 `cstacallid`。 **注：** 可从 uaCSTA 通话事件中获取 `cstacallid` 。
ipaddress	是	String	分机号注册的 IP 话机的 IP 地址。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /api/v2.0.0/
uacstacall/refuse_inbound?token=38a0563849a15cd7187b835d1acbf5a5 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
  "extnumber":"1000",
  "cstacallid":"32785",
  "ipaddress":"192.168.66.70:5060"
 }
```

响应示例

```
HTTP/1.1 200 OK
Date: Wed, 19 May 2021 08:06:22 GMT
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success"
 }
```

uaCSTA - 挂断通话

通过 uaCSTA 挂断通话。

前提条件

要实现对 IP 话机进行 uaCSTA 控制操作，需满足以下前提要求。

- 使用符合 Yeastar uaCSTA 的型号和固件版本要求的 IP 话机。
- 启用 IP 话机的 uaCSTA 功能。可参考以下文章：
  - 如何启用 Yealink 话机的 uaCSTA 功能
  - 如何启用 Fanvil 话机的 uaCSTA 功能
- 启用 PBX 服务器的 uaCSTA 功能（路径：**配置 > PBX > 常规设置 > SIP > 高级 > 启用 uaCSTA 连接**）

请求地址

```
{base_url}/{api_path}/uacstacall/hangup?token={token}
```

请求参数

参数	是否必选	类型	描述
extnumber	是	String	要控制的分机号。
cstacallid	是	Integer	每个 uaCSTA 通话的唯一 ID。uaCSTA 功能启用后，系统会自动为每个通话分配一个 `cstacallid`。 **注：** 可从 uaCSTA 通话事件中获取 `cstacallid` 。
ipaddress	是	String	分机号注册的 IP 话机的 IP 地址。

响应参数

参数	类型	描述
status	String	请求结果。 • Success：成功。 • Failed：失败。

示例

请求示例

```
POST /
api/v2.0.0/uacstacall/refuse_inbound?token=38a0563849a15cd7187b835d1acbf5a5
 HTTP/1.1
Content-Type:application/json; charset=utf-8
Host: api.yeastarcloud.com
{
```

```
  "extnumber":"1000",
  "cstacallid":"32785",
  "ipaddress":"192.168.66.70:5060"
 }
```

响应示例

```
HTTP/1.1 200 OK
Server: Boa/0.94.101wk
Accept-Ranges: bytes
Connection: close
Content-Type: application/json
Access-control-allow-origin: *
Access-control-allow-methods: GET, POST, OPTIONS, PUT, DELETE
{
  "status":"Success"
 }
```

事件推送

系统事件

BootUp（系统启动）

应用服务端连接上 API 服务器，并且完成认证。当 PBX 系统重启后，PBX 会向应用服务器推送该报告。

报告参数

参数	类型	描述
event	String	事件类型。
sn	String	PBX 的 SN 码。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
```

```
Accept: application/json

{
 "event": "BootUp",
 "sn": "369451090405"
}
```

ConfigChange（配置变更）

当 PBX 的配置发生变化时，PBX 会向应用服务器发送配置变更报告，以便于应用服务器及时更新和同步 PBX 的相关配置。

> **注：**
>
> - 目前只支持发送分机和中继配置变更的报告。
> - PBX 只报告哪个分机或哪个中继配置有变更，具体变更信息，你需要登录 PBX 网页查看。

报告参数

参数	类型	描述
event	String	事件类型。
type	String	变更类型。 • extension：分机。 • trunk：中继。
extid	Integer	配置有变更的分机号码。
trunkname	String	配置有变更的中继名称。
operation	String	操作类型。 • add：添加。 • del：删除。 • update：更新。
sn	String	PBX 的 SN 码。

报告示例

```
POST / HTTP/1.1
```

```
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
{
 "event": "ConfigChange",
 "type": "extension",
 "extid": "1000",
 "operation": "update",
 "sn": "369451090405"
}
```

来电呼入控制流程事件

Invite（来电呼叫请求）

在开启中继的 **来电接听控制** 的情况下，当来电通过该中继呼入时，PBX 会向应用服务器推送 Invite 事件。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
inbound	Object <InboundInfo>	外线来电信息。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。

参数	类型	描述
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "Invite",
 "callid": "1571896991.2",
 "members": [
  {
   "inbound": {
    "from": "2000",
    "to": "1023",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000001"
   }
  }
 ],
 "sn": "369451090405"
}
```

Incoming（来电呼入）

在开启中继的 **呼入目的地控制** 情况下，当来电通过该中继呼入时，在 PBX 应答该来电前，会向应用服务器推送 Incoming 事件。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
inbound	Object <InboundInfo>	外线来电信息。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "Incoming",
 "callid": "1571897079.3",
 "members": [
  {
   "inbound": {
    "from": "2000",
    "to": "1023",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000002"
   }
  }
 ],
 "sn": "369451090405"
}
```

ExtensionStatus (分机状态变更事件)

在分机开启 **状态监控** 的情况下，分机状态发生变更时，PBX 主动向应用服务器推送变更信息。

报告参数

参数	类型	描述
event	String	事件类型。
extension	Integer	状态变更的分机号。
status	String	分机当前状态。 • Ringing：响铃。分机有来电正在响铃时，PBX 会向应用服务器推送该报告。 • Busy：忙碌。分机待机时摘机或接听来电后，PBX 会向应用服务器推送该报告。 • Registered：分机已注册。 以下情况下，PBX 会向应用服务器推送该报告。 ◦ SIP 分机由未注册变为注册上。 ◦ SIP 分机的注册地址变更时。 ◦ SIP 分机由忙线状态变为空闲状态。 • Unregistered：未注册。当 SIP 分机由注册上变为未注册时，PBX 会向应用服务器推送该报告。
registeredIP	String	分机注册的设备 IP 地址。
sn	String	PBX 的 SN 码。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "ExtensionStatus",
 "extension": "1023",
 "status": "Ringing",
 "registeredIP": "192.168.5.100",
        "sn": "369451090405"
}
```

CallStatus（通话状态变更事件）

PBX 实时向应用服务器推送当前所有通话的状态信息。

> **注：**
> 如需监控中继的通话状态，你需要在 PBX API 网页开启中继的 **状态监控**。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。

参数	类型	描述
		注： 通过 IVR、队列、响铃组、广播组或会议室与分机建立通话时，才会携带 callpath。 例如：分机 1000 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过中继、IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。 例如：外线 2203303 呼叫 IVR （6500），通过 IVR 转拨分机 2000，callpath 即为 6500。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。

参数	类型	描述
trunkname	String	从哪条中继呼出。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 **注：** 通过 IVR、队列或响铃组与外线号码建立通话时，才会携带 callpath。 例如：使用 IVR （6500）拨打外线号码 5503302，callpath 为 6500。

报告示例

外部号码 12345678 拨打分机号码 1023，分机 1023 响铃。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "CallStatus",
 "callid": "1571887584.74",
 "members": [
  {
   "inbound": {
    "from": "12345678",
    "to": "1023",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000030",
```

```
    "memberstatus": "ALERT"
   }
  },
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-00000031",
    "memberstatus": "RING"
   }
  }
 ],
 "sn": "369451090405"
}
```

NewCdr（通话记录事件）

当通话结束后，PBX 向应用服务器实时发送通话详细记录（CDR）。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
timestart	String	开始时间。
callfrom	String	主叫号码。
callto	Integer	被叫号码。
callduraction	String	通话时长。
talkduraction	String	接听时长。
srctrunkname	String	源中继名称。
dsttrcunkname	String	目的中继名称。
pincode	Integer	密码。
status	String	通话状态。 • ANSWERD：被叫接听电话。 • NO ANSWER：被叫未接听电话。 • BUSY：被叫正在忙。 • VOICEMAIL：被叫号码未接听，且有新的语音留言。
type	String	通话类型。

参数	类型	描述
		• Internal：内部通话。 • Inbound：外部呼入。 • Outbound：呼出外部。
recording	String	全局录音文件名。
didnumber	Integer	来电用户拨打的号码。
agentringtime	String	队列坐席从响铃到接听的时间。
sn	String	PBX 的 SN 码。

报告示例

分机 1023 呼叫分机 2023。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
        "event": "NewCdr",
 "callid": "1571897930.5",
 "timestart": "2019-10-23 22:18:50",
 "callfrom": "1023",
 "callto": "2023",
 "callduraction": "19",
 "talkduraction": "14",
 "srctrunkname": "",
 "dsttrcunkname": "",
 "pincode": "",
 "status": "ANSWERED",
 "type": "Internal",
        "recording": "20191024201242-1571897930.5-203-201-Internal.wav",
 "didnumber": "",
 "agentringtime": "0",
 "sn": "369451090405"
}
```

Transfer（呼叫转移事件）

当 PBX 分机用户转移通话时，PBX 向应用服务器发送 Transfer 报告。

注：

- 转移操作为分机拨打特征码 *03 或 *3 将通话转接。
- 通过 call/transfer 和 call/attended_transfer 调用的呼叫转移不发送事件报告。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	从哪条中继呼出。
channelid	String	通话中，每个成员的通话通道 ID。

报告示例

分机 2000 与分机 1020 通话中，分机 1020 将电话转移到另外一个号码。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json
{
 "event": "Tranfer",
 "callid": "1571981311.105",
 "members": [
  {
   "ext": {
    "number": "2000",
    "channelid": "PJSIP/2000-00000025"
   }
  },
  {
   "ext": {
    "number": "1020",
    "channelid": "PJSIP/1020-00000026"
   }
  }
 ],
 "sn": "369451090405"
}
```

外部用户 12345678 呼入 PBX，与分机 1020通话，分机 1020 将电话转移到另外一个号码。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
```

```
Content-Type: application/json
Accept: application/json

{
 "event": "Tranfer",
 "callid": "1571981647.117",
 "members": [
  {
   "inbound": {
    "from": "12345678",
    "to": "1020",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000002b"
   }
  },
  {
   "ext": {
    "number": "1020",
    "channelid": "PJSIP/1020-0000002c"
   }
  }
 ],
 "sn": "369451090405"
}
```

Forward（呼叫前转事件）

分机开启呼叫转移后，当该分机的来电被转移时，PBX 向应用服务器发送 Forward 报告。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberTye

参数	类型	描述
inbound	Object <InboundInfo>	外线来电的信息。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。

报告示例

外部用户 12345678 呼叫分机 1020，分机 1020 将电话转移到另外一个号码。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "Forward",
 "callid": "1571901993.46",
 "members": [
  {
   "inbound": {
    "from": "12345678",
    "to": "1020",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000001c"
   }
  }
 ],
 "sn": "369451090405"
}
```

CallFailed（呼叫失败事件）

当发起的呼叫失败时，PBX 会向应用服务器发送 CallFailed 报告。

注：

> 通过 API 接口调用的通话或用话机建立的通话，呼叫失败都会触发该事件。呼叫失败不包括通话被转到语音留言的情况。

报告参数

参数	类型	描述
callid	String	每个通话的唯一 ID。
event	String	事件类型。
reason	String	呼叫失败原因： • NO Outbound Restriction：分机无呼出权限。 • Circuit/channel congestion：占线，通话通道被占用。 • DND：被叫启用了 DND。 • Line Unreachable：外线不可用。 • User Busy：被叫拒接来电。 • 410 Gone：离开。 • 404 NOT found：未找到被叫号码。 • 408 Request Time：被叫未接听来电，或被叫关机。 • 480 Temporatily Unavailable：被叫没有接听来电。 • 484 Address incomplete：呼出号码格式错误。 • 486 Busy here：被叫拒接。 • 603 Declined：被叫响铃超时。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "callid": "1573097568.15",
 "event": "CallFailed",
 "reason": "NO Outbound Restriction",
 "members": [
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-0000000a",
   }
  }
 ],
 "sn": "369451090405"
}
```

DTMF（按键信息事件）

通话过程中，当用户根据提示输入相关的按键（如通过 IVR 输入按键），PBX 向应用服务器推送 DTMF 报告。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
info	String	单个 DTMF 按键。
flag	Integer	标识 DTMF 按键的顺序。 **注：**

参数	类型	描述
		flag 从 0 开始标识。0 表示第一个 DTMF 按键；1 表示第二个 DTMF 按键；以此类推。
infos	String	DTMF 按键串。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 例如，分机呼叫 IVR （6500），输入按键信息，callpath 即为 6500。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。

参数	类型	描述
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	该成员的通话状态。 • ALERT：该成员作为主叫，发起呼叫后，处于回铃状态。 • RING：该成员作为被叫，处于响铃状态。 • ANSWERED：该成员作为主叫，发起呼叫后，电话被接听。 • ANSWER：该成员作为被叫，接听来电。 • HOLD：该成员的通话被保持住。 • BYE：该成员主动挂断通话。
callpath	Integer	通话建立使用的路径。 例如，外线用户呼叫 IVR（6500），输入按键信息，callpath 即为 6500。

报告示例

分机 1023 拨打 IVR 号码，输入按键 2。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1571905397.97",
 "members": [
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-00000039",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
  }
 ],
 "info": "2",
 "flag": "0",
 "sn": "369451090405"
}
```

外部号码 192929500 呼入 IVR 6500，拨打分机 1020。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1573197470.151",
 "members": [
  {
   "inbound": {
    "from": "192929500",
    "to": "6500",
    "trunkname": "To6.36",
    "channelid": "PJSIP/trunk-To6.36-endpoint-0000003d",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
  }
 ],
 "info": "1",
 "flag": "0",
 "sn": "369451090405"
}
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1573197470.151",
 "members": [
  {
   "inbound": {
    "from": "192929500",
    "to": "6500",
    "trunkname": "To6.36",
    "channelid": "PJSIP/trunk-To6.36-endpoint-0000003d",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
```

```
  }
 ],
 "info": "0",
 "flag": "1",
 "sn": "369451090405"
}
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1573197470.151",
 "members": [
  {
   "inbound": {
    "from": "192929500",
    "to": "6500",
    "trunkname": "To6.36",
    "channelid": "PJSIP/trunk-To6.36-endpoint-0000003d",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
  }
 ],
 "info": "2",
 "flag": "2",
 "sn": "369451090405"
}
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1573197470.151",
 "members": [
  {
   "inbound": {
    "from": "192929500",
    "to": "6500",
```

```
    "trunkname": "To6.36",
    "channelid": "PJSIP/trunk-To6.36-endpoint-0000003d",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
  }
 ],
 "info": "0",
 "flag": "3",
 "sn": "369451090405"
}
POST / HTTP/1.1
Host: api.yeastarcloud.com
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "DTMF",
 "callid": "1573197470.151",
 "members": [
  {
   "inbound": {
    "from": "192929500",
    "to": "6500",
    "trunkname": "To6.36",
    "channelid": "PJSIP/trunk-To6.36-endpoint-0000003d",
    "memberstatus": "ANSWERED",
    "callpath": "6500"
   }
  }
 ],
 "infos": "1020",
 "sn": "369451090405"
}
```

分机 1023 拨打外部号码，进入对方的 IVR，输入按键 1。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
```

```
 "event": "DTMF",
 "callid": "1573092021.6",
 "members": [
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-00000002",
    "memberstatus": "ANSWERED"
   }
  }
 ],
 "info": "1",
 "flag": "0",
 "sn": "369451090405"
}
```

PlayPromptEnd（语音播放结束事件）

PBX 向分机号码或外线号码播放音乐结束后，会向应用服务器推送 PlayPromptEnd 报告。

报告参数

语音播放结束包括两种类型：IVR 语音播放结束、指定语音文件播放结束。

指定语音播放结束

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
prompt	String	播放的语音文件名称。
sn	String	PBX 的 SN 码。

IVR 语音播放结束

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
members	Array <MemberType>	通话中的不同类型成员的信息列表。

参数	类型	描述
ivr	Object <IVRInfo>	IVR 提示音对象。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
inbound	Object <InboundInfo>	外线来电的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
callpath	Integer	通话建立使用的路径。 例如，IVR （6500）语言播放结束后，callpath 即为 6500。

InboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过哪条中继呼入。
channelid	String	通话中，每个成员的通话通道 ID。
callpath	Integer	通话建立使用的路径。 例如，IVR （6500）语言播放结束后，callpath 即为 6500。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	从哪条中继呼出。

参数	类型	描述
channelid	String	通话中，每个成员的通话通道 ID。
callpath	Integer	通话建立使用的路径。 例如，IVR （6500）语言播放结束后，callpath 即为 6500。

IVRInfo

参数	类型	描述
number	String	IVR 号码。

示例 1

面向广播组成员播放语音结束后，PBX 向应用服务器推送的报告。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "PlayPromptEnd",
 "callid": "1571965641.241",
 "members": [
  {
   "ext": {
    "number": "2023",
    "channelid": "PJSIP/2023-0000008f"
   }
  },
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-00000090"
   }
  }
 ],
 "prompt": "music1",
 "sn": "369451090405"
}
```

示例 2

面向外线播放语音结束后，PBX 向应用服务器推送的报告。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "PlayPromptEnd",
 "callid": "1571909302.194",
 "members": [
  {
   "outbound": {
    "from": "prompt",
    "to": "2000",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000074"
   }
  }
 ],
 "prompt": "music1",
 "sn": "369451090405"
}
```

示例 3

面向分机播放 IVR 提示音，播放结束后，PBX 向应用服务器推送的报告。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "PlayPromptEnd",
 "callid": "1571910100.203",
 "members": [
  {
   "ext": {
    "number": "1023",
    "channelid": "PJSIP/1023-00000079",
```

```
                              "callpath":"6500"
    }
   }
  ],
  "ivr": {
   "number": "6500"
  },
  "sn": "369451090405"
}
```

示例 4

面向外线呼入后播放 IVR 提示音，播放结束后，PBX 向应用服务器推送的报告。

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "PlayPromptEnd",
 "callid": "1571910690.206",
 "members": [
  {
   "inbound": {
    "from": "2000",
    "to": "6500",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000007b",
                              "callpath":"6500"
    }
   }
  ],
  "ivr": {
   "number": "6500"
  },
  "sn": "369451090405"
}
```

示例 5

使用 IVR 呼出外线，提示音播放结束后，PBX 向应用服务器推送的报告。

```
POST / HTTP/1.1
```

```
Host: 192.168.5.150: 8260Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "PlayPromptEnd",
 "callid": "1571979960.74",
 "members": [
  {
   "outbound": {
    "from": "6501",
    "to": "2100",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-00000014",
                            "callpath":"6501"
   }
  }
 ],
 "ivr": {
  "number": "6501"
 },
 "sn": "369451090405"
}
```

satisfaction（满意度评价事件）

进行满意度调查时，当收到用户的按键信息后，PBX 主动向应用服务器发送满意度调查的结果。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
surveyresult	String	满意度调查按键结果。
sn	String	PBX 的 SN 码。

报告示例

```
POST / HTTP/1.1
```

```
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "satisfaction",
 "callid": "1571972541.133",
 "surveyresult": "1",
 "sn": "369451090405"
}
```

ConferenceStatus（与会成员变更事件）

当新成员加入会议或有成员退出会议时，PBX 会向应用服务器推送当前的会议室成员信息。

报告参数

参数	类型	描述
event	String	事件类型。
callid	String	每个通话的唯一 ID。
conferencenumber	Integer	会议号码。
conferencename	Integer	会议名称。
members	Array <MemberType>	通话中的不同类型成员的信息列表。
sn	String	PBX 的 SN 码。

MemberType

参数	类型	描述
ext	Object <ExtensionInfo>	分机的信息。
outbound	Object <OutboundInfo>	外线呼出的信息。

ExtensionInfo

参数	类型	描述
number	String	分机号码。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	与会成员的状态。

参数	类型	描述
		• ANSWERED：该成员拨打会议室号码，主动进入会议室。 • ANSWER：该成员接受邀请，进入会议室。
callpath	Integer	通话建立使用的路径，即会议室号码。

OutboundInfo

参数	类型	描述
from	Integer	主叫号码。
to	Integer	被叫号码。
trunkname	String	通过那条中继呼出。
channelid	String	通话中，每个成员的通话通道 ID。
memberstatus	String	与会成员的状态。 • ANSWERED：该成员拨打会议室号码，主动进入会议室。 • ANSWER：该成员接受邀请，进入会议室。
callpath	Integer	通话建立使用的路径，即会议室号码。

示例

```
POST / HTTP/1.1
Host: 192.168.5.150: 8260
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
 "event": "ConferenceStatus",
 "callid": "1572313740.138",
 "conferencenumber": "6409",
 "conferencename": " 6409",
 "members": [
  {
   "outbound": {
    "from": "6409",
    "to": "2100",
    "trunkname": "siptrunk",
    "channelid": "PJSIP/trunk-siptrunk-endpoint-0000001f",
```

```
    "memberstatus": "ANSWERED",
                            "callpath":"6409"
   }
  },
  {
   "ext": {
    "number": "1020",
    "channelid": "PJSIP/1020-0000001e",
    "memberstatus": "ANSWERED",
                            "callpath":"6409"
   }
  },
  {
   "ext": {
    "number": "2023",
    "channelid": "PJSIP/2023-0000001d",
    "memberstatus": "ANSWERED",
                            "callpath":"6409"
   }
  }
 ],
 "sn": "369451090405"
}
```


uacstacall（uaCSTA 通话事件）

若启用 uaCSTA 功能，在分机响铃或使用 uaCSTA 功能接受或拒绝分机通话后，PBX 会向应用服务器发送事件报告。

前提条件

要实现对 IP 话机进行 uaCSTA 控制操作，需满足以下前提要求。

- 使用符合 Yeastar uaCSTA 的型号和固件版本要求的 IP 话机。
- 启用 IP 话机的 uaCSTA 功能。可参考以下文章：
  - 如何启用 Yealink 话机的 uaCSTA 功能
  - 如何启用 Fanvil 话机的 uaCSTA 功能
- 启用 PBX 服务器的 uaCSTA 功能（路径：**配置 > PBX > 常规设置 > SIP > 高级 > 启用 uaCSTA 连接**）

报告参数

参数	类型	描述
event	String	事件类型。
operation	String	执行的操作。 • `callstart`：分机正在响铃。 • `callanwer`：通过 uaCSTA 接听来电。 • `callover`：通过 uaCSTA 挂断通话或拒接来电。
extnumber	Integer	要控制的分机号。
cstacallid	String	每个 uaCSTA 通话的唯一 ID。uaCSTA 功能启用后，系统会自动为每个通话分配一个 `cstacallid`。 **注：** 可从 uaCSTA 通话事件中获取 `cstacallid` 。
ipaddress	String	分机号注册的 IP 话机的 IP 地址。
sn	String	PBX 序列号。

报告示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
{
  "event":"uacstacall",
  "operation":"callanswer",
  "extnumber":"1000",
  "cstacallid":"32788",
  "ipaddress":"192.168.5.70:5060",
  "sn":"369451090405"
 }
```

QueueAgentRingNoAnswerTimeout (队列坐席响铃超时)

队列坐席响铃超时时，PBX 会向应用服务器推送未接来电相关信息。

报告参数

参数	类型	说明
event	String	事件类型。
queue	String	功能名称 (Queue) 和队列号码。
agent	String	坐席名称。
callid	String	每个通话的唯一 ID。
caller	String	主叫号码。
sn	String	PBX 的 SN 码。

示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
    "event": "QueueAgentRingNoAnswerTimeout",
    "queue": "queue-6700",
    "agent": "Phillip Huff",
    "callid": "1711516130.30",
    "caller": "2000",
    "sn": "369451090405"
}
```

QueueAutoPause (队列坐席自动暂停)

队列坐席因未接来电达到最大值而被自动暂停服务时，PBX 会向应用服务器推送相关信息。

报告参数

参数	类型	说明
event	String	事件类型。
queue	String	队列号码。
agent	String	坐席号码。
misscall	Array<Missed_Calls>	未接来电列表。

参数	类型	说明
sn	String	PBX 的 SN 码。

Missed_Calls

参数	类型	说明
caller	String	主叫号码。
callid	String	每个通话的唯一 ID。

示例

```
POST / HTTP/1.1
Host: api.yeastarcloud.com
Accept: application/json
User-Agent: WebAPI
Content-Type: application/json
Accept: application/json

{
    "event": "QueueAutoPause",
    "queue": "6700",
    "agent": "1000",
    "misscall": [
        {
            "caller": "2000",
            "callid": "1711360264.92"
        }
    ],
    "sn": "369451090405"
}
```

错误代码

错误代码	说明
10002	不支持 XML 数据封包格式
10003	API 不支持的请求号处理
10004	部分参数为空
10005	callid 不存在
10006	分机不存在

错误代码	说明
10007	通话不存在
10008	分机不处于空闲状态
10009	分机不允许被监听
10010	监听模式不匹配
10011	被叫号不存在
10012	IVR 不存在
10013	队列不存在
10014	响铃组不存在
10015	通话未桥接成功
10016	接受来电/拒绝来电超时
10017	配置分机失败
10018	配置 IVR 失败
10019	配置队列失败
10020	添加闹铃失败或者修改闹铃失败
10021	分机闹铃不存在
10022	分机闹铃已存在
10023	语音文件不存在
10024	分机没有呼出权限
10025	apply 太过频繁
10028	号码不能为空
10031	呼出号码不符合要求
10032	密码错误
10033	分机路由配置失败
10042	外线名称错误
10043	外线名称重复
10044	外线类型错误
10045	外线主机地址错误
10046	外线端口错误
10047	外线域名错误
10048	外线用户名错误
10049	外线用户名重复

错误代码	说明
10050	外线认证名称错误
10051	外线 From 头域错误
10052	外线认证密码错误
10053	外线 DOD 配置不存在分机
10054	外线 DOD 配置不存在分机组
10055	外线 DOD 号码错误
10056	外线 DOD 号码长度错误
10057	外线 DOD 名称错误
10058	外线 DOD 名称长度错误
10059	外线 DOD 格式错误
10060	外线添加达到上限
10061	外线删除失败
10062	外线不存在
10063	呼入路由名称错误
10064	呼入路由重复
10065	呼入路由 DID 配置错误
10066	呼入路由来电匹配错误
10067	呼入路由外线不存在
10068	呼入路由目的类型错误
10069	呼入路由目的地错误
10070	呼入路由添加达到上限
10071	呼入路由删除失败
10072	呼入路由不存在
10073	呼出路由名称错误
10074	呼出路由重复
10075	呼出路由前缀错误
10076	呼出路由删除前缀数错误
10077	呼出路由前置错误
10078	呼出路由外线不存在
10079	呼出路由分机不存在
10080	呼出路由分机组不存在

错误代码	说明
10081	呼出路由添加达到上限
10082	呼出路由删除失败
10083	呼出路由不存在
10088	外线 DID 号码错误
10089	外线 DID 号码长度错误
10090	外线 DID 名称错误
10091	外线 DID 名称长度错误
10092	外线 DID 添加达到上限
10100	监听类型不存在
10101	调用参数超出该接口定义的功能范围，请检查 caller 参数和 callee 参数
10102	会议室设置失败，因为部分参数有问题
10103	channelid 不存在
10104	提示音音量不在 0-20 范围内
10105	咨询转操作类型不存在
10106	不存在会话中的会议室
10107	未咨询转前，不能执行咨询操作
10110	会议室号不存在
10112	成员都不在会议中
10113	会议室号已存在
10115	值必须为数字和字母
10116	系统连接 AMI 出错
10117	临时会议室号不能在固定的会议室中
10118	提示音音量不是纯数字
10119	管理员成员不存在
10120	会议室成员数量达到上限
10121	会议室名字只能为数字和字母
10122	会议室创建时，呼叫管理员失败
10123	会议室 id 不存在
10124	会议室数量达到上限
10125	会议室静音失败
10126	会议室解除静音失败

错误代码	说明
10127	会议室踢出成员失败
10128	会议室 id 只能为数字和字母
10129	没有权限进行邀请
10130	外线不能进行咨询转
10131	号码超出范围
10132	系统未安装 Linkus 应用
10133	`enablelinkus` 值无效
10134	分机对应的邮箱地址未设置
10135	分机未启用 Linkus 客户端
10136	系统未设置邮箱服务器
10137	Linkus 用户数超出限制
10138	联系人 id 不存在
10139	服务器不支持通讯录功能
10140	填写的值不符合要求
10141	通讯录中没有联系人
10142	企业联系人达到添加上限
10143	分机组 id 输入错误
10144	分机组名称格式错误
10145	分机组名称重复
10146	添加失败，请确认数据库是否损坏
10148	分机组数量已达上限
10149	更新的分机组中不存在有效分机号
10151	停泊号码异常
10152	呼叫停泊操作失败
10153	uaCSTA操作失败
20000	只支持 JSON 的格式
20001	只支持 POST 的请求
20002	用户登录失败（被锁）
20003	用户登录失败（无效的用户名或密码）
20004	无此类 Token
20005	Token 为空

错误代码	说明
20006	Token 超时
20007	数据为空
20008	端口错误
20009	请求号为空
20010	请求发送给 apisrv 处理失败
20011	HeartBeat 请求配置失败
20012	录音文件随机串无效
20013	验证码错误
20014	录音文件不存在
20015	allowedip 错误
20016	语音留言状态参数有误
20017	语音留言随机串无效
20018	语音留言文件不存在 **注：** 可确认下 下载链接 是否已过期。
20019	CDR 时间格式错误
20020	CDR 随机串无效
20021	CDR 不存在
20022	YMPIP 无效
30001	JSON 申请空间失败
30002	请求处理 AMI 的返回数据超时
30003	apisrv进程未启动

API 场景应用

酒店客房管理应用方案

通过 API 对接 云 PBX 和酒店管理系统，实现酒店客房分机管理、叫醒服务、通话统计、迷你吧消费统计、房间清洁状态变更和退房的自动化管理。

应用场景

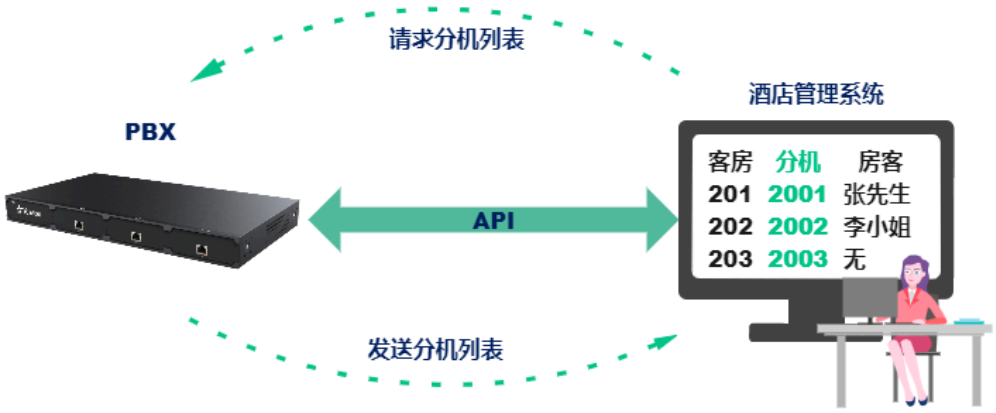
传统酒店采用程控电话交换机为房客提供话音业务，依靠酒店管理系统，人工记录和传递客房信息，服务单一且联动性相对较差。云 PBX 提供的 API 接口，可以将酒店管理系统与 PBX 结合，为房客提供多样化的话务处理，提升客户服务品质；同时，为服务员提供自动化的客房管理，提高工作效率，有效地减少人工成本，改进服务水平。

客房分机管理

房客入住时，前台可以根据房客等级，分配分机功能、设置呼叫权限、绑定房客信息等。当房客呼叫前台时，前台可以准确识别房客身份，提供专业服务；同时，房客可以享受 PBX 提供的通信服务，如内部客房分机互拨免费、语音信箱等。

- **获取分机列表**
  1. 前台在酒店管理系统上查看客房分机时，酒店管理系统向 PBX API 接口发送分机列表请求。
  2. PBX 收到请求，将分机列表信息发送给酒店管理系统，前台即可获取客房分机信息。

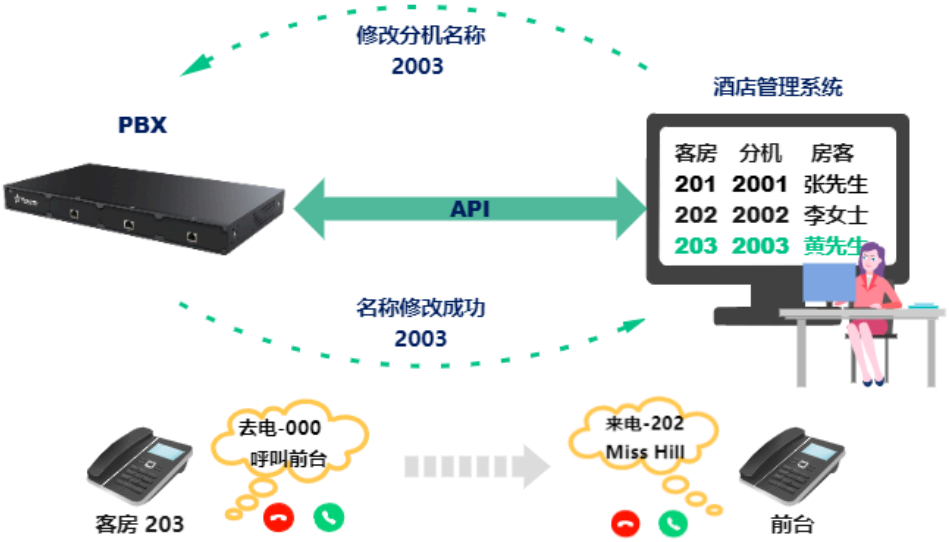
对接接口请参阅 查询分机详情。



- **更改分机设置**

1. 前台在酒店管理系统上录入房客信息。酒店管理系统向 PBX API 接口发送分机设置请求，将客房分机名称设置为房客名称，并根据房客等级设置分机功能，如语音信箱、国际长途呼叫权限等。
2. PBX 收到请求，变更分机设置，并向酒店管理系统发送分机变更结果。

对接接口请参阅 修改分机。

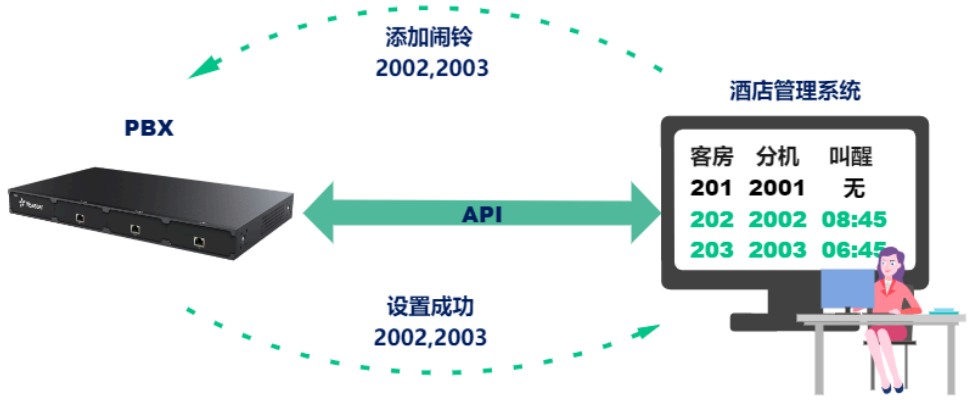


叫醒服务

前台根据房客入住的房号、入住时长及叫醒需求，在酒店管理系统上设置闹铃方式，通过客房分机响铃和语音提示的方式，提供叫醒服务。操作简单，且一次设置，团房生效。尤其是旅游团房客入住时，批量自动化的叫醒服务可以提升酒店服务质量和服务员的工作效率。

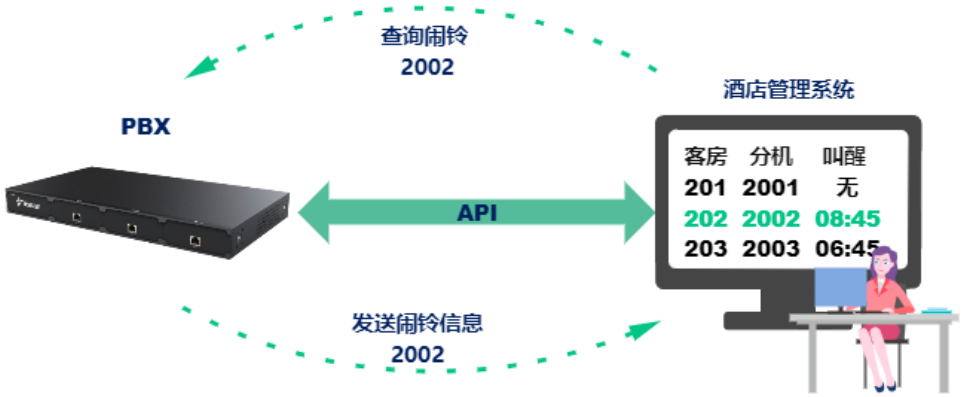
- **设置闹铃叫醒**
  1. 前台在酒店管理系统上设置响铃方式以及需要响铃的客房，酒店管理系统向 PBX API 接口发送闹铃设置请求，设置响铃方式及应用的分机。
  2. PBX 收到请求，为分机启用闹铃，并向酒店管理系统发送闹铃设置结果。

对接接口请参阅 添加闹铃。



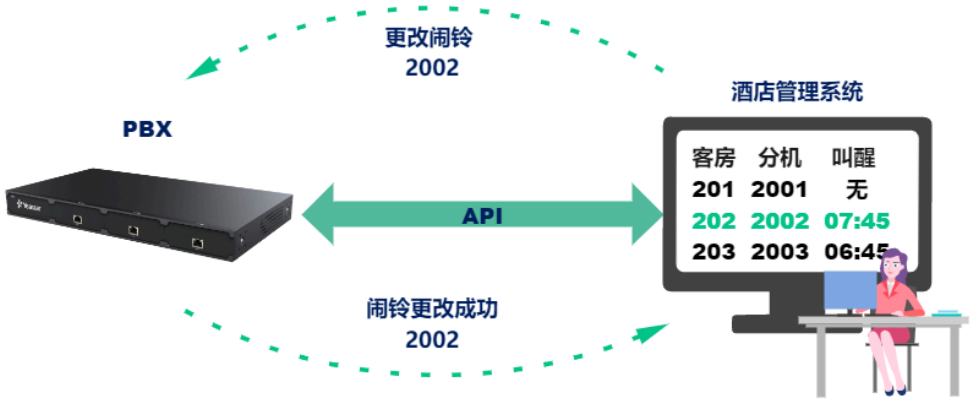
- **查询闹铃**
  1. 前台在酒店管理系统上查看个别客房闹铃设置情况时，酒店管理系统向 PBX API 接口发送闹铃查询请求。
  2. PBX 收到请求，将分机对应的闹铃信息发送给酒店管理系统，前台即可获取客房闹铃设置情况。

对接接口请参阅 查询闹铃。



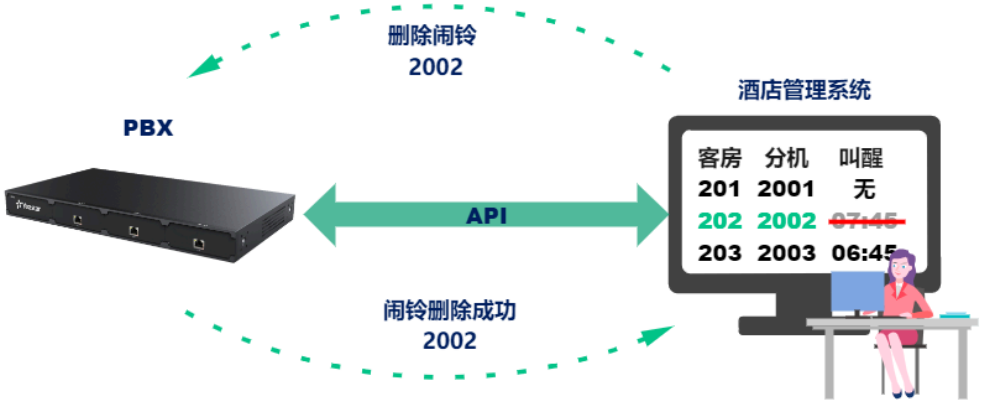
- **更改闹铃**
  1. 前台在酒店管理系统上修改个别客房闹铃设置时，酒店管理系统向 PBX API 接口发送特定分机的闹铃修改请求。
  2. PBX 收到请求，修改分机闹铃设置，并向酒店管理系统发送闹铃更改结果。

对接接口请参阅 修改闹铃。



- **删除闹铃**
  1. 个别房客不再需要叫早服务时，前台在酒店管理系统上删除个别客房闹铃，酒店管理系统向 PBX API 接口发送特定分机的闹铃删除请求。
  2. PBX 收到请求，删除分机闹铃，并向酒店管理系统发送闹铃删除结果。

  对接接口请参阅 删除闹铃。



通话计费

房客通话结束，PBX 实时向系统发送 通话记录 报告（CDR），系统可以实时统计房客的通话情况。

迷你吧消费统计

房客退房后，服务员可以通过客房分机录入房客消费信息，提高工作效率，节省房客的等待时间。

1. 服务员清点迷你吧商品，统计客户消费清单后，通过客房分机拨打迷你吧号码，根据语音提示输入位商品编号和数量。
2. PBX 根据收集到的按键信息，生成按键信息报告，发送给酒店管理系统，实现客户消费清单快速录入。

DTMF 按键报告示例

分机（2003）拨打迷你吧号码 778，输入商品编码（123456789）个数 2。

```
{
 "action": "DTMF",
 "callid": "1575425107.149",
 "call": [
  {
   "ext": {
    "extid": "2003"
   }
```

```
  }
 ],
 "infos": "778*123456789*2",
 "sn": "369462038851"
}
```

设置迷你吧操作和语音提示

联系Yeastar技术支持获取迷你吧操作的配置文件。本文示例使用 extensions_support.conf。

1. 设置迷你吧商品录入操作。
   a. 登录 PBX 网页，使用 FTP 或 SSH 方式配置房态变更操作。
      - FTP：上传 extensions_support.conf 到 FTP 目录 `/support/customcfg/`。
      - SSH：进入`/ysdisk/support//customcfg/extensions_support.conf`，编辑 extensions_support.conf 文件。

```
[hotelba]
exten = 778,1,NoOp(778)
same = n,Background(product_id)
same = n,Read(digito,,9,,,4)
  // 9位商品编号
same = n,Gotoif($[${LEN(${digito})}>0]?check,1)
same = n,Background(product_id)
same = n,Read(digito,,9,,,4)
same = n,Gotoif($[${LEN(${digito})}>0]?check,1)
same = n,Background(product_id)
same = n,Read(digito,,9,,,4)
same = n,Gotoif($[${LEN(${digito})}>0]?check,1)
same = n,Goto(h,1)                    //三次没有输入
商品编号，则挂断
exten = check,1,Macro(productlist,${digito},1)
same = n,Goto(778,2)
exten = h,1,Hangup()

[macro-productlist]
exten = s,1,Set(product_id=${ARG1})
same = n,Background(product_idc)
  //您输入的商品编码是
same = n,SayDigits(${product_id})
same = n,Background(product_num)
same = n,Read(digito,,2,,,4)
  // 2位商品个数，超时4秒
```

```
same = n,Gotoif($[${LEN(${digito})}>0]?num,1)
same = n,Background(product_num)
same = n,Read(digito,,2,,,4)
same = n,Gotoif($[${LEN(${digito})}>0]?num,1)
same = n,Background(product_num)
  //两次没有输入数量则返回上级菜单
exten = num,1,Set(product_num=${digito})
same = n,Background(product_numc)         //
 您输入的商品个数是...
same = n,SayDigits(${product_num})         //
same = n,Goto(confirm,1)
exten = confirm,1,Background(confirm)
same = n,Read(digito,,1,,,4)
same = n,Gotoif($[${digito}=1]?yes,1)
  //按1确认
same = n,Gotoif($[${digito}=*]?778,1)
  //按*返回
same = n,Gotoif($[${digito}=9]?h,1)
  //按9挂掉
same = n,Background(vm-goodbye)
exten = yes,1,UserEvent(DtmfEndString,Info: 778*$
{product_id}*${product_num})   //发出DTMF串
exten = yes,2,Background(success)    //输入成功
exten = h,1,hangup()
```

2. 设置迷你吧商品录入语音提示。
   你可以直接使用语音吧文件，或自行制作语音，通过 FTP 方式上传到 `/ysapps/pbxcenter/var/lib/asterisk/sounds/zh/`。

> **注：**
> 语音格式要求 8K 16Bit 单声道 wav 格式。可以使用合成语音。

文件名称	语音内容
confirm.wav	确认请按1，取消并返回请按*
playnum.wav	您输入的编号是
pro_name.wav	请输入商品编号和个数
product_id.wav	请输入商品编码
product_idc.wav	您输入的商品编码是
product_n um.wav	请输入商品个数

文件名称	语音内容
product_numc.wav	您输入的商品个数是
success.wav	成功输入

房间清洁状态变更

保洁员清理客房后，可以通过客房分机修改房间清洁状态；前台可实时查看房间的清洁状态。

1. 保洁员通过客房分机拨房态变更号码，根据语音提示输入房态号码。
2. PBX 根据收集到的按键信息，生成按键信息报告，发送给酒店管理系统，实现房态变更。

房态变更报告示例

话机拨打 777，根据语音提示，输入 *01 设置为已清洁。

```
{
 "event": "DTMF",
 "callid": "1576311280.555",
 "members": [
  {
   "ext": {
    "number": "1001",
    "channelid": "PJSIP/1001-0000013c",
    "memberstatus": "ANSWERED"
   }
  }
 ],
 "infos": "777*01",
 "sn": "369462038851"
```

```
}
```

设置房态变更操作和语音提示

联系Yeastar技术支持获取房态变更操作的配置文件。本文示例使用 extensions_support.conf。

1. 设置房态变更操作。

   登录 PBX 网页，使用 FTP 或 SSH 方式配置房态变更操作。

   - FTP：上传 extensions_support.conf 到 FTP 目录 `/support/customcfg/`。
   - SSH：进入`/ysdisk/support//customcfg/extensions_support.conf`，编辑 extensions_support.conf 文件。

```
[Local_Default_CallingRules]
include = clean
include = hotelba
[clean]
exten = 777,1,NoOp(777)
same = n,Goto(clean-action,play,1)

[clean-action]
exten = play,1,Background(roomstatus)
same = n,WaitExten(4)
same = n,Goto(h,1)
exten = *00,1,Playback(room)
exten = *00,n,SayDigits(${CALLERID(num)})
exten = *00,n,UserEvent(DtmfEndString,Info:777${EXTEN}
)
exten = *00,n,Playback(uncleaned)  //,设置为未清洁,再见
exten = *00,n,Hangup()

exten = *01,1,Playback(room)

exten = *01,n,SayDigits(${CALLERID(num)})

exten = *01,n,UserEvent(DtmfEndString,Info:777${EXTEN}
)
exten = *01,n,Playback(cleaned)   //,设置为已清洁,再见
exten = *01,n,Hangup()

exten = h,1,Hangup()
```

```
exten = i,1,Goto(always-Hangup,h,1)
```

2. 设置房态变更语音提示。
   你可以直接使用语音吧文件，或自行制作语音，通过 FTP 方式上传到 `/ysapps/pbxcenter/var/lib/asterisk/sounds/zh/`。

> **注：**
> 语音格式要求 8K 16Bit 单声道 wav 格式。可以使用合成语音。

文件名称	语音内容
cleaned.wav	设置为已清洁，再见
uncleaned.wav	设置为未清洁，再见
roomstatus.wav	请输入房间清洁状态
room.wav	房间

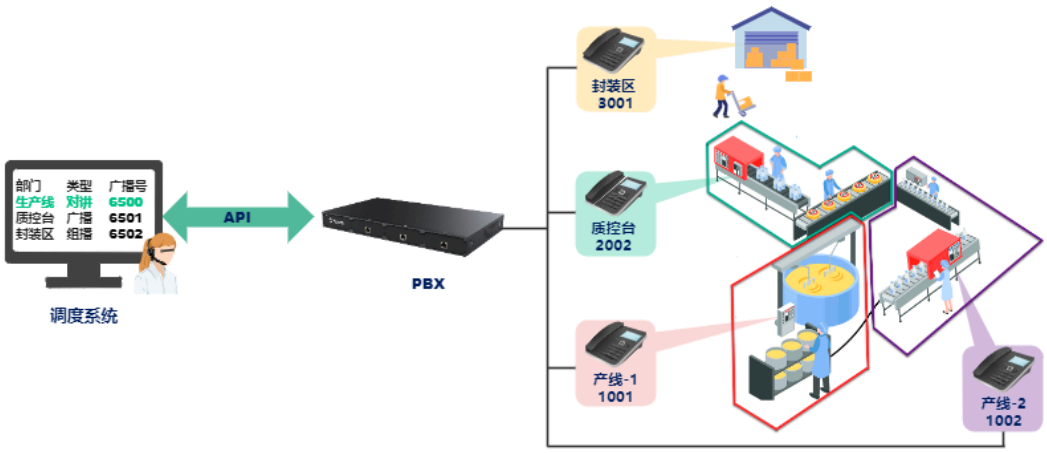
指挥调度系统应用方案

通过 API 对接 云 PBX 和调度系统，借助终端话机实现单向、双向和组播的指挥调度。

应用场景

指挥调度系统常用于工业生产、矿井作业、铁路、机场、酒店等，是企业安全生产、运营管理中的重要通信手段之一。云 PBX 提供的 API 接口，可以将调度系统与 PBX 结合。通过灵活的终端分组，实现跨部门、跨地区之间的广播和对讲，在满足日常语音通信的同时，统一指挥调度，提高协作办公效率与应急处理能力。

以工业生产为例，生产线各环节工艺流程复杂，上下工序之间衔接紧密，可按职能划分广播组，确保各职能人员之间能够及时有效地传递生产信息或处理紧急事务。

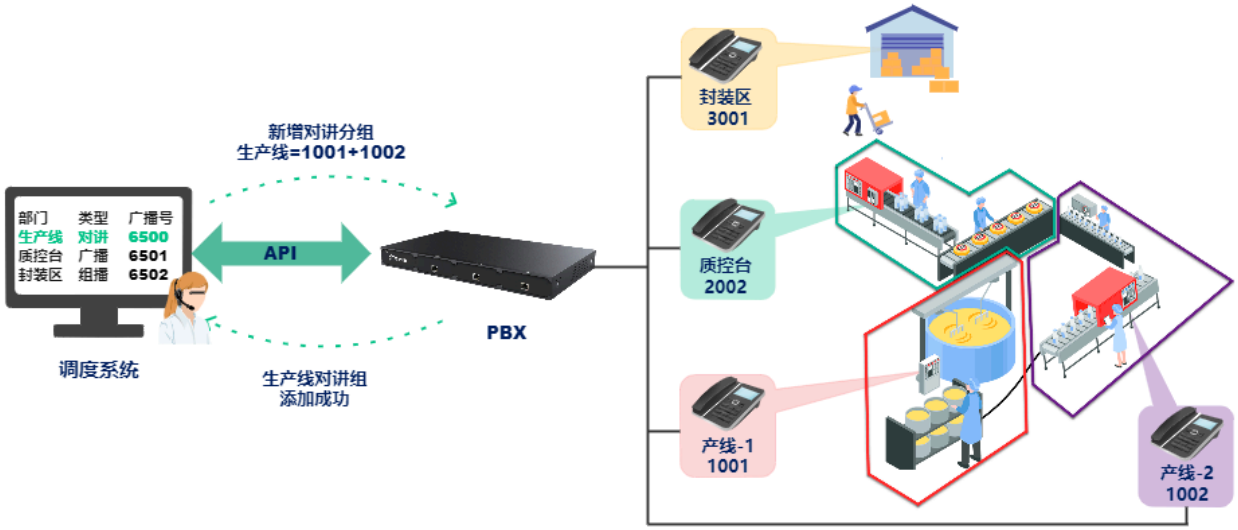


添加广播组

调度员可以将隶属于同一部门的分机划分到独立的广播组，根据各部门职能的划分，设置广播组类型。

1. 调度员将产线-1 与产线-2 设置为对讲组，调度系统向 PBX API 接口发送新建广播组请求。
2. PBX 接收请求后，新建广播组，将产线-1 与产线-2 对应的分机加入到广播组中，并向调度系统发送广播组创建结果。

对接接口请参阅 添加广播组。

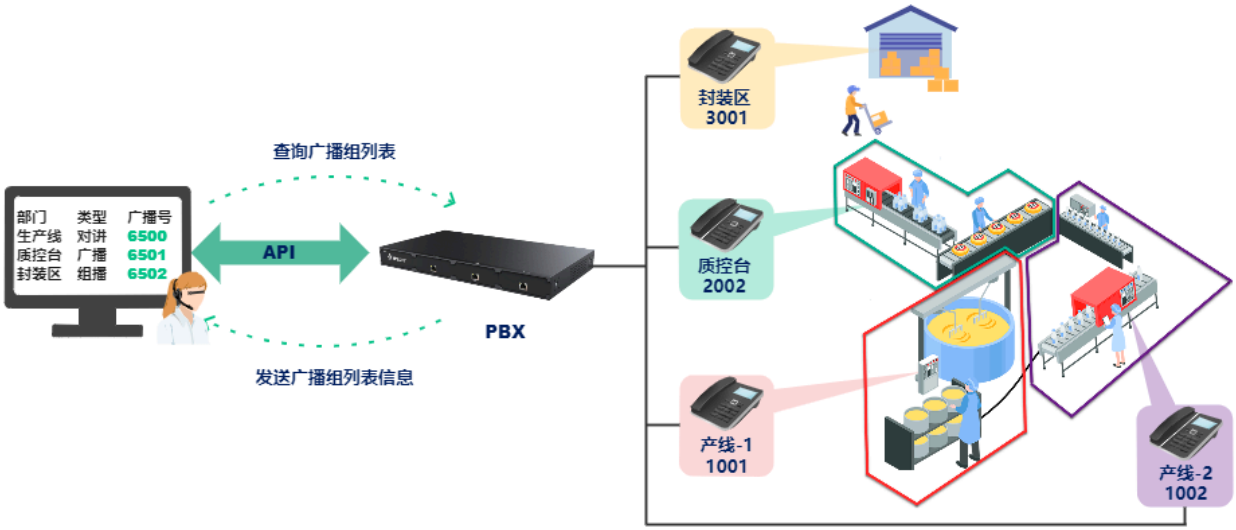


查询广播组

调度员在发布广播时，可以在调度系统上查看 PBX 广播组。

1. 调度员在调度系统上查看 PBX 广播组，调度系统向 PBX API 接口发送分机列表查询请求。
2. PBX 收到请求，将分组列表信息发送给调度系统。

对接接口请参阅 查询广播组详情。

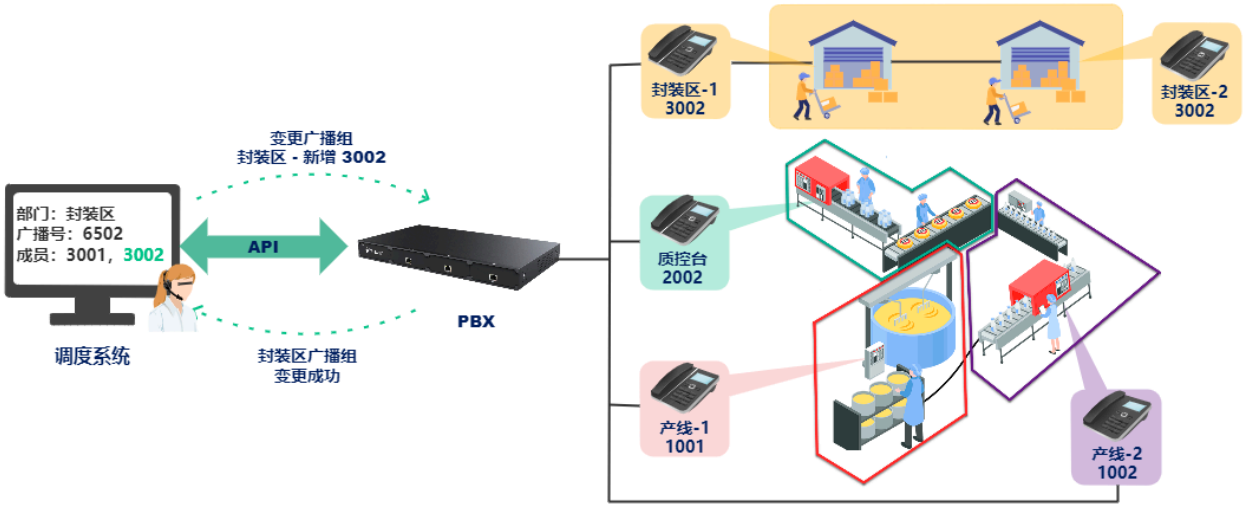


修改广播组设置

当部门组织变更，调度员可以在对应的广播组中添加或删除广播对象，修改广播组类型等。

1. 调度员在调度系统上为个别广播组新增成员，调度系统向 PBX API 接口发送广播组变更请求。
2. PBX 收到请求，变更广播组信息，并向调度系统发送广播组变更结果。

对接接口请参阅 编辑广播组。

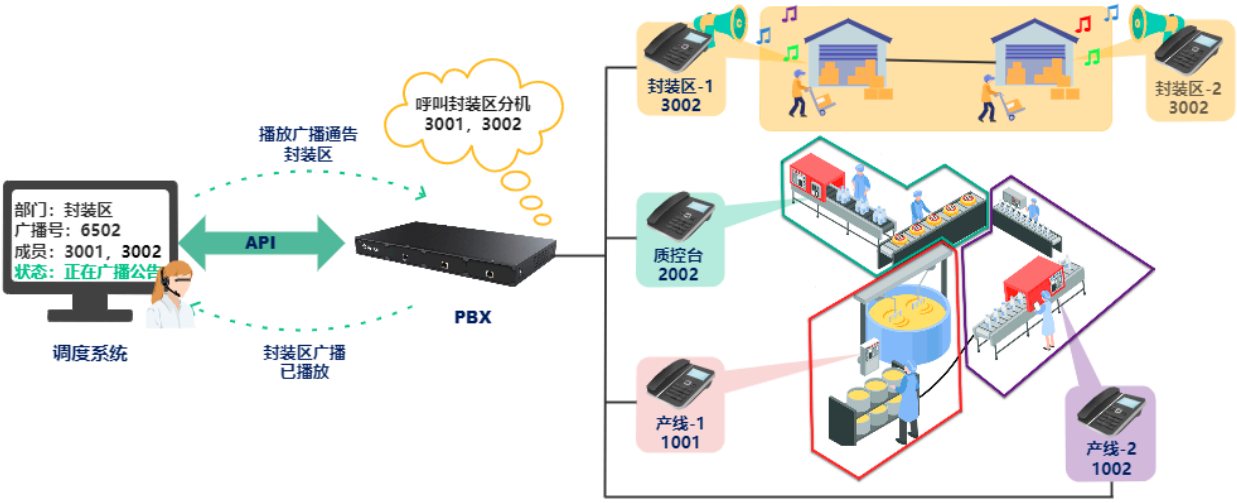


发起广播

调度员可以通过调度系统向广播组成员广播喊话；出现紧急情况时，调度员可以实时发起对讲，协调资源、讨论解决方案，远程指挥和调度；业余时间，调度员可以向广播组成员播放录制好的音频通告，宣传安全生产知识等。

1. 调度员在调度系统上向广播组播放音频通告，调度系统向 PBX API 接口发送播放请求。
2. PBX 收到请求后，呼叫广播组对应的分机，播放预先设置的音频内容，并向调度系统发送播放结果。

对接接口请参阅 播放语音。



自定义广播音频

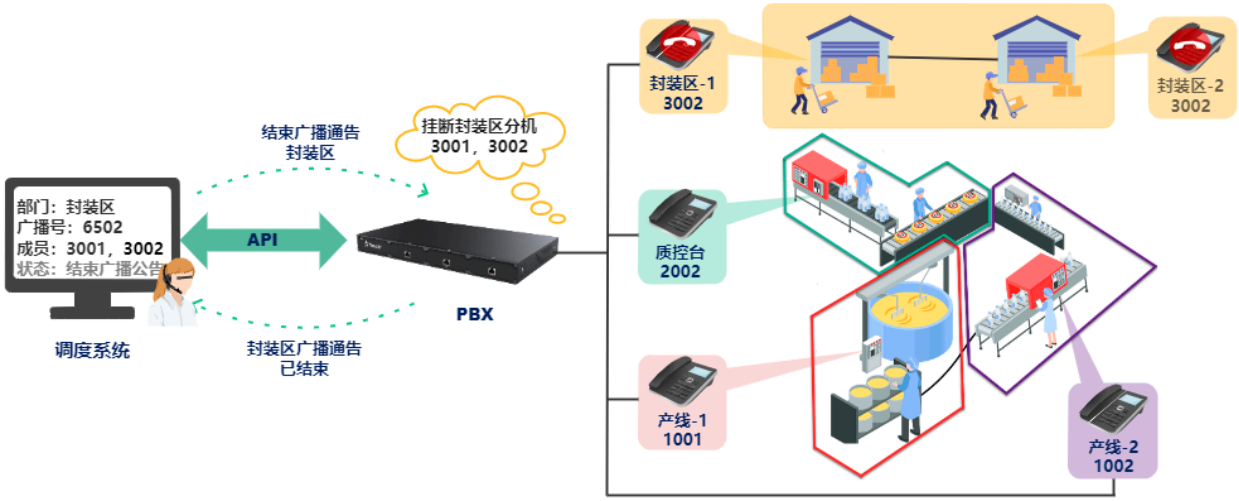
1. 登录 PBX 网页，进入 **PBX > 提示音 > 自定义提示音**，上传自定义的录音文件。

挂断广播音乐

如果系统当前正在向广播组成员播放音频，调度员可以结束音频广播。

1. 调度员在调度系统上结束广播，调度系统向 PBX API 发送结束请求。
2. PBX 收到请求后，挂断广播组分机，并向调度系统发送广播结束结果。

对接接口请参阅挂断广播语音。



呼叫中心系统应用方案

通过 API 对接 云 PBX 和呼叫中心，实现自动话务分配、坐席管理、日常通话和满意度调查。

应用场景

呼叫中心，是企业加强与客户的沟通，提高服务质量和服务效率，提高企业竞争能力的一种重要手段。随着互联网的迅速普及，怎样使呼叫中心接入互联网，同时解决呼叫中心的分布式组建和高成本等问题，成为中小型企业与跨地区企事业单位首要考虑的问题。云 PBX 提供的 API 接口，可以直接将 IP PBX 接入呼叫中心系统，直接替代昂贵的 ACD 服务器，同时能更好地兼容 CTI 系统，实现坐席管理、自动话务分配和满意度调查等功能。

队列设置

- **创建坐席队列**

  根据不同的服务类型，设置不同的坐席分组。不同类型的客户来电可以分配给不同的坐席分组。

  1. 登录 PBX 网页界面，进入 **配置 > PBX > 呼叫功能 > 队列**。
  2. 设置固定座席或者使用动态座席。

     动态座席可以通过 API 迁入、迁出（上下班），建议使用动态座席。
  3. 设置响铃策略。

     一般是使用话务平均分配（ACD），选择最少被叫响铃、或者最少应答响铃。

- **获取队列设置**

  在 PBX 上创建队列后，管理员可以在呼叫中心系统上获取队列信息，如：队列号码、队列名称、坐席成员等。

  1. 管理员在呼叫中心系统上查看队列信息，呼叫中心系统向 PBX API 接口发送队列信息请求。
  2. PBX 收到请求，向呼叫中心系统发送队列信息。

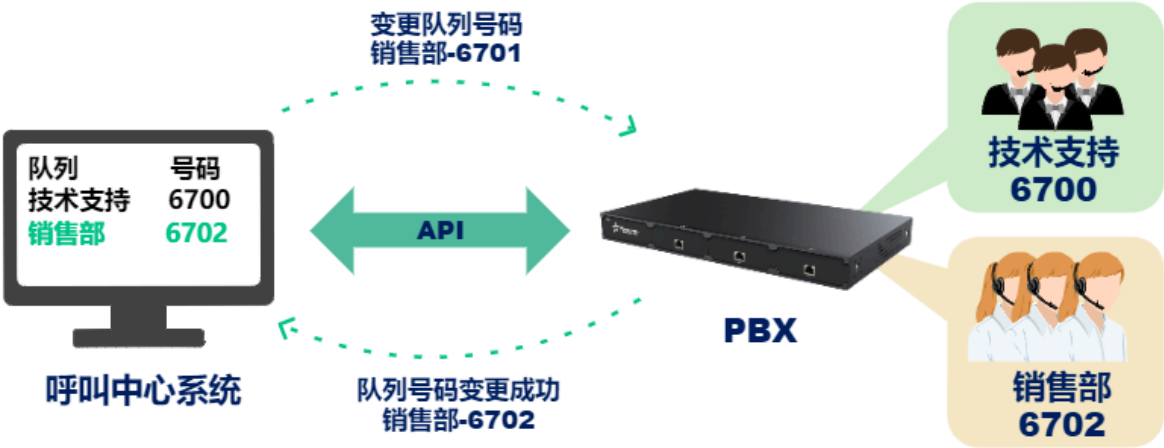
  对接接口请参阅 查询队列。

- **修改队列设置**

管理员可以在呼叫中心系统上修改队列设置，如：响铃策略、固定坐席、按键目标等。

1. 管理员在呼叫中心系统上修改队列设置，呼叫中心系统向 PBX API 接口发送队列变更请求。
2. PBX 收到请求，变更队列设置，并向呼叫中心系统发送队列设置变更结果。

对接接口请参阅 修改队列。



- **查看队列状态**

管理员可以在呼叫中心系统上查看队列话务状态及坐席状态。

1. 管理员在呼叫中心系统上查看队列 6700 的话务状态，呼叫中心系统向 PBX API 接口发送查看请求。
2. PBX 收到请求，将队列 6700 的话务状态信息发送给呼叫中心系统。

对接接口请参阅 查询队列状态。

> **注：**
> 若 API 接口启用分机状态监控后，坐席分机状态变更时，PBX 实时向呼叫中心系统发送坐席分机状态报告，管理员可以实时查看着坐席状态。

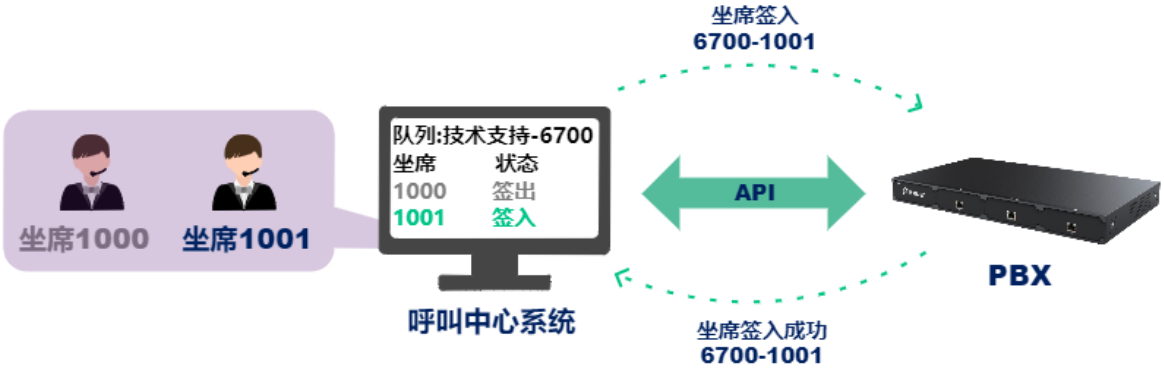
坐席功能

- **签入队列**

  坐席签入队列即开始工作，系统自动向签入队列的坐席分配来电。

  1. 坐席在呼叫中心系统上签入队列，呼叫中心系统向 PBX API 接口发送坐席签入请求。
  2. PBX 收到请求后，签入坐席，并向呼叫中心系统发送处理结果。

  对接接口请参阅 签入队列。

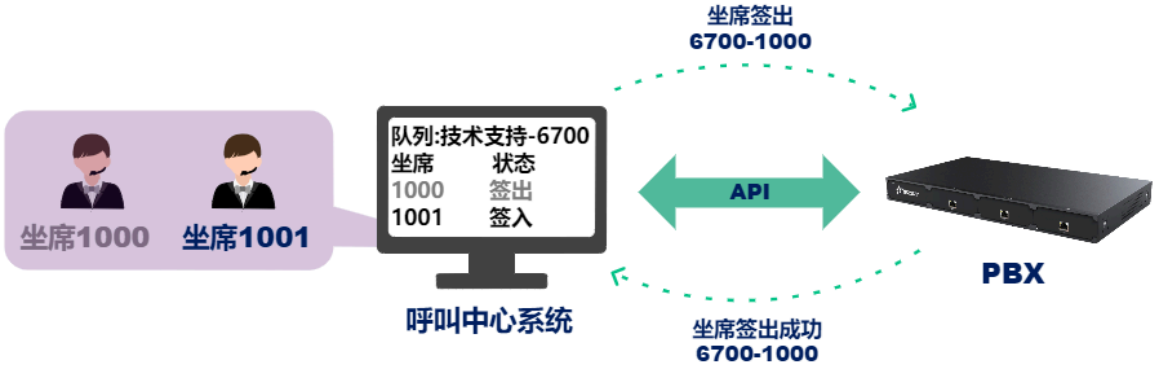


- **签出队列**

  坐席结束工作后，可以签出队列，系统停止向签出队列的坐席分配来电。

1. 坐席在呼叫中心系统上签出队列，呼叫中心系统向 PBX API 接口发送坐席签出请求。
2. PBX 收到请求后，签出坐席，并向呼叫中心系统发送处理结果。

对接接口请参阅 签出队列。

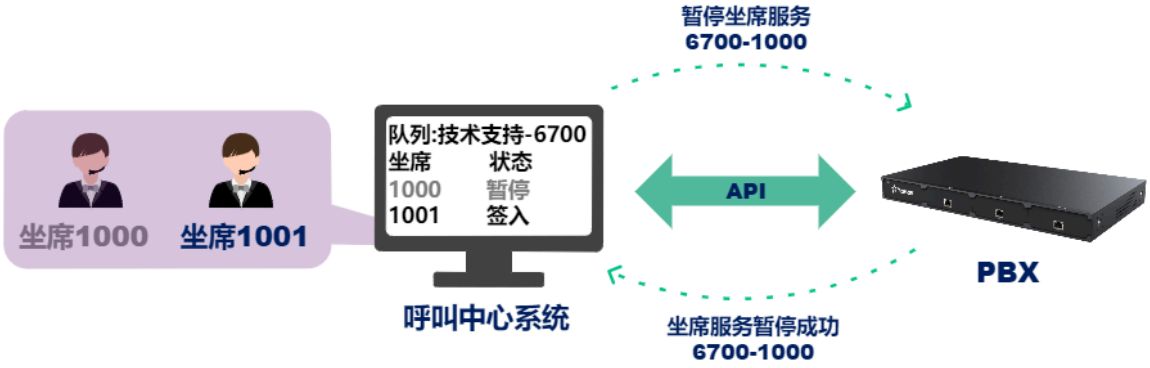


- **暂停坐席服务**

坐席暂时离开座位或短时间休息时，可以暂停坐席服务，系统停止向签出队列的坐席分配来电。

1. 坐席在呼叫中心系统上启用暂停服务，呼叫中心系统向 PBX API 接口发送暂停坐席服务的请求。
2. PBX 收到请求后，暂停坐席服务，并向呼叫中心系统发送处理结果。

对接接口请参阅暂停坐席服务。

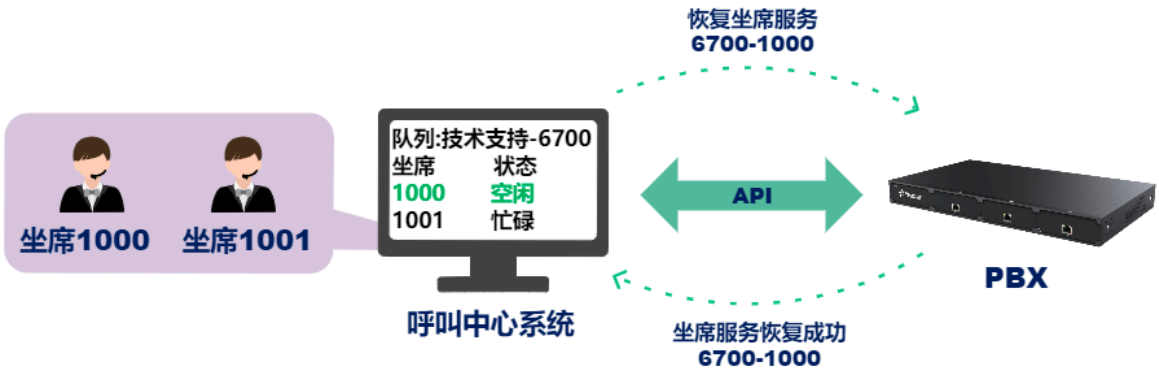


- **恢复坐席服务**

坐席重新投入工作后，可以恢复坐席服务，坐席状态变更为空闲时，系统自动向签出队列的坐席分配来电。

1. 坐席在呼叫中心系统启用恢复服务，呼叫中心系统向 PBX API 接口发送恢复坐席服务的请求。
2. PBX 收到请求后，恢复坐席服务，并向呼叫中心系统发送处理结果。

对接接口请参阅 恢复坐席服务。



通话功能

- **外线拨号**

  在做客户回访时，坐席直接使用分机呼叫客户。

  1. 坐席在呼叫中心系统上发起外线呼叫，呼叫中心系统向 PBX API 接口发送外呼请求。
  2. PBX 收到请求后，先呼叫坐席分机，座席分机接听后，再呼叫客户。

  对接接口请参阅 拨打外线号码。

- **内线互拨**

  1. 坐席在呼叫中心系统上发起分机呼叫，呼叫中心系统向 PBX API 接口发送分机呼叫请求。
  2. PBX 收到请求后，先呼主叫分机；主叫分机接听后，再呼被叫分机。

  对接接口请参阅 分机互拨。

- **自动外呼**

  在电话销售时，呼叫中心可以自动呼叫客户，在客户接听后，将通话转接给坐席，避免繁琐的拨号操作，提高坐席工作效率。

  1. 呼叫中心系统向 PBX API 接口发送外呼请求。
  2. PBX 收到请求后，呼叫客户，在客户接听后，将通话转接到队列中，再由分配的坐席进行对接。

  对接接口请参阅 队列外呼。

满意度调查

座席挂断电话后，系统自动引导客户完成满意度调查，快速收集客户评价，以便更好的改善服务质量。

1. 坐席挂断电话，系统自动向客户播报满意度调查提示，客户根据提示按键评价。
2. PBX 收集按键信息，生成 满意度调查报告，发送给呼叫中心系统，快速实现客服质量信息收集。

IVR 预警系统应用方案

通过 API 对接 云 PBX 和预警系统，提供语音预警播报。

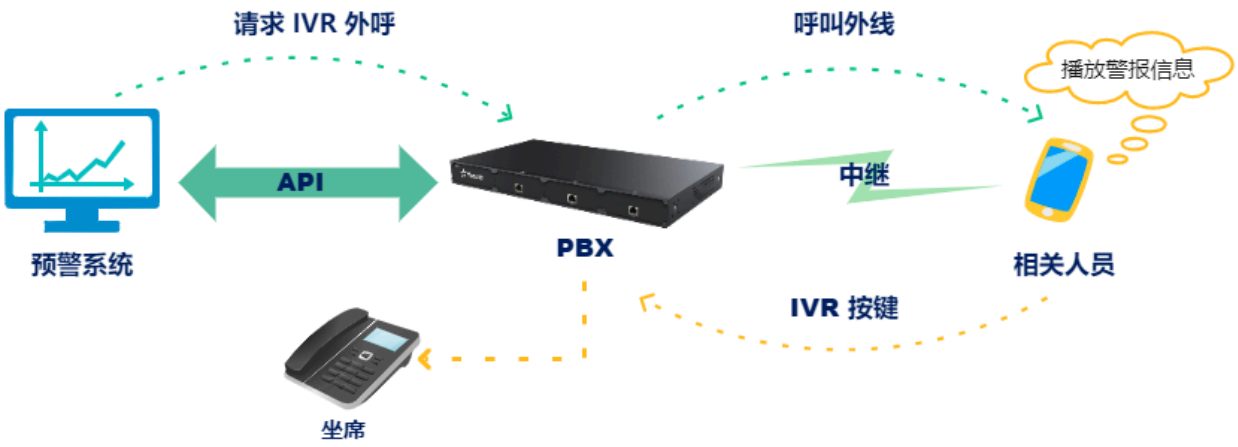
应用场景

预警系统常用于工业生产、气象灾害监测、消防等，是安全生产、防灾减灾、运营管理中的重要预防手段之一。预警信息传递渠道的单一性，往往导致险情预报时，不能够快速、及时、有效地传达警情信息。云 PBX 提供的 API 接口，可以将预警系统与 PBX 结合。当监测对象达到预警值时，可以通过 IVR 自动呼叫相关人员，快速、有效地传达警报。相关人员可以根据不同的预警等级，联系不同的部门，进行联动防控。

IVR 拨打外线

1. 预警系统向 PBX API 接口发送 IVR 外呼请求。
2. PBX 收到请求，使用 IVR 呼叫相关人员。
3. 相关人员听到报警信息后，可以根据 IVR 语音提示，按相应的按键呼叫应急中心坐席或其它人员。

对接接口请参阅 IVR 拨打外线。



IVR 预警状态报告

PBX 向预警系统实时发送当前呼叫的状态。

- 当呼出的通话被应答时，PBX 会向预警系统发送“通话被应答”报告。

> **注：**
> 如果是模拟外线，呼出的通话默认都是被应答的，如果要区分“应答”或“未被应答”，需要模拟外线开通应答反极信号，并且在 PBX 上开通“应答反极检测“。如果要准确的返回是否被应答，建议使用 E1 或 IMS 线路。

- PBX 给分机号码或外线号码播放提示音结束后，会发送“语音播放结束”报告。通过记录语音播放结束时间，可以了解相关人员是否完整的听完了语音。
- 当一通通话结束后，PBX 向预警系统实时发送 通话记录 报告（CDR）。通过此报告，可以记录警情发生时间和频率。

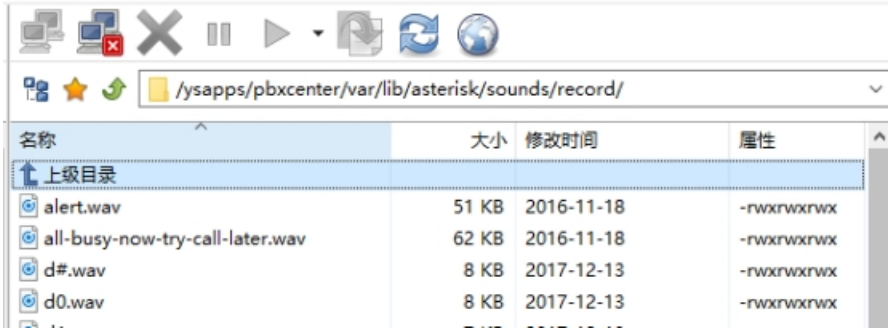
IVR 设置

1. 上传 IVR 播放的语音。
   - 通过 PBX 上传 IVR 语音。

     登录 PBX 网页界面，进入 **配置 > PBX > 提示音 > 自定义提示音**，上传自定义的提示音。

- 通过 FTP 上传录音文件。

  如下图所示，FTP 路径：/ysapps/pbxcenter/var/lib/asterisk/sounds/record/。



2. 设置 IVR。

   进入 **配置 > PBX > 呼叫功能 > IVR**，点击 **添加** 创建 IVR，选择提示音。

3. 设置按键事件。

   用户接听后可以根据预警级别的不同，通过按键转给相应部门。

IVR 语音互动服务应用方案

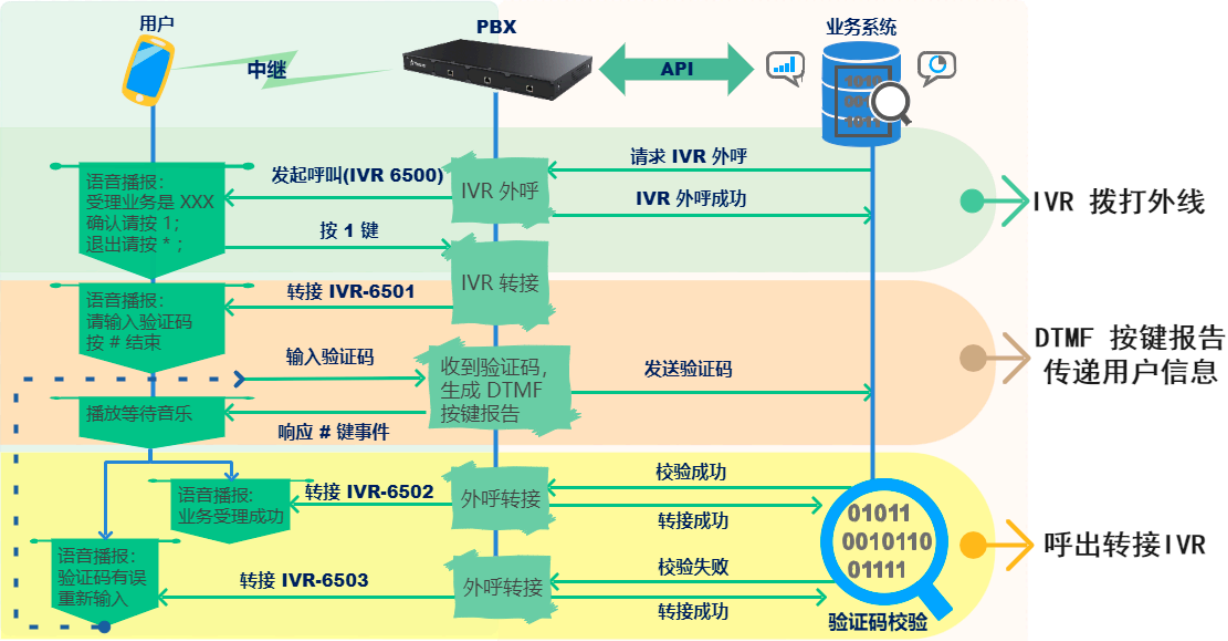
通过 API 对接 云 PBX 和业务系统，提供语音互动服务。

应用场景

语音互动服务，是一种功能强大的电话自动服务系统。它用预先录制或 TTS 文本转语音技术合成的语音进行自动应答，为客户提供语音菜单导航，常用于信息查询与确认、问卷调查、满意度调查、业务推广等。云 PBX 提供的 API 接口，可以将业务系统与 PBX 结合，通过 IVR 语音交互引导客户操作，自动完成客户的信息收集，信息确认等工作，减少人力投入，提高业务办理效率。

IVR 应用流程示例

以业务办理为例，系统呼叫用户，通过 IVR 播放业务内容和操作提示，自动引导用户输入验证码，完成业务办理。



1. 业务系统请求 PBX 通过 IVR 呼叫用户，用户接听后，向客户播报 IVR 6500 语音。
2. 客户根据语音提示操作按 1 键，PBX 响应按键事件，将通话转接给 IVR 6501，提示用户输入验证码。
3. 客户根据语音提示输入验证码。PBX 收到按键信息，向客户播放等待音乐，并生成 DTMF 按键报告，将用户信息发送给业务系统。
4. 业务系统收到按键信息报告，校验客户数据。根据校验结果向 PBX 发送通话转接请求。
    - 校验成功，PBX 将当前通话转接 IVR 6502，语音提示用户业务受理成功。
    - 校验失败，PBX 将当前通话转接 IVR 6503，提示客户验证码有误，重新输入。

IVR 拨打外线

1. 业务系统向 PBX API 接口发送 IVR 外呼请求。
2. PBX 收到请求，使用 IVR 呼叫客户。

   IVR 外呼对接接口请参阅 IVR 拨打外线。
3. PBX 向业务系统实时发送当前呼叫的状态。

   通话状态接口请参阅 CallStatus（通话状态变更事件）。

DTMF 按键报告

1. 客户根据语音提示输入相关信息。
2. PBX 收集相关的按键信息，生成 “按键信息” 报告，并发送给业务系统。

呼出转接 IVR

1. 业务系统收到 DTMF 报告，验证用户信息后，向 PBX 发送通话转接请求。
2. PBX 收到请求，将通话转接给相应的 IVR，向客户播放相应的 IVR 语音。

对接接口请参阅 [呼叫转移(盲转)](#)。

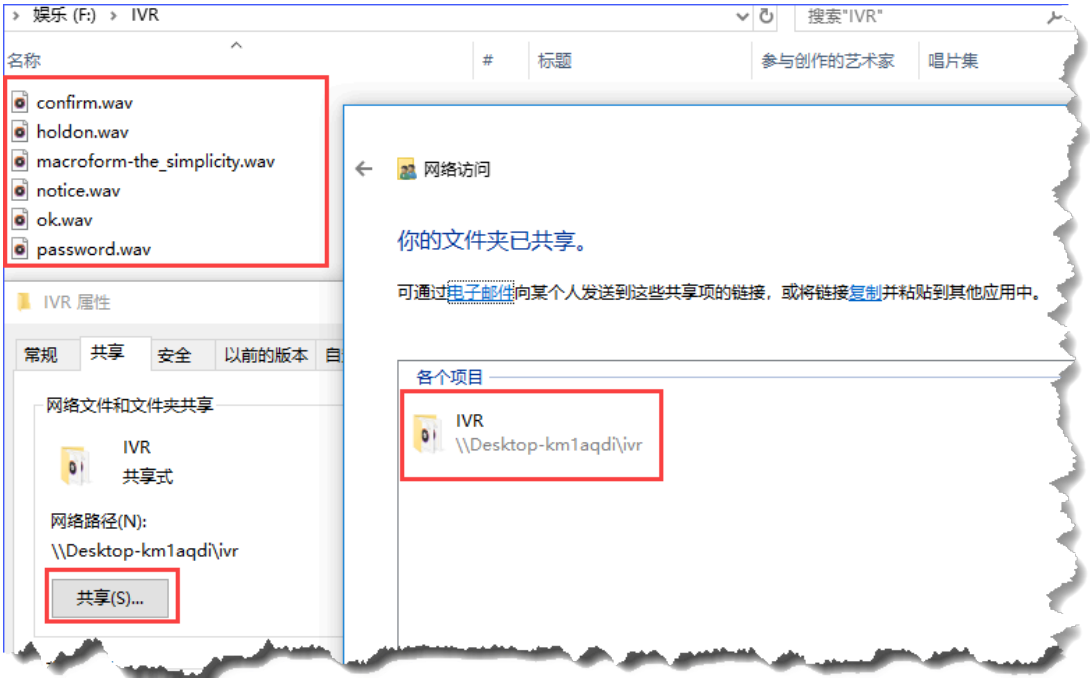
IVR 语音菜单设置

1. 在 PBX 上创建 IVR。

2. 设置 IVR 语音文件。

   语音由 TTS 产生 wav 文件，放到网络磁盘位置供 PBX 调用。

   a. 将 IVR 语音上传到网络存储盘，并将存储 IVR 语音的文件夹设置为共享。



b. 将网络存储盘挂载到 PBX。

第一个网络磁盘挂载后，路径为 `/tmp/media/networkdisk1`。

3. 设置 IVR 语音菜单。

联系Yeastar技术支持获取 IVR 语音菜单脚本。

登录 PBX 网页，使用 SSH 方式，进入`/ysdisk/support//customcfg/extensions_support.conf`，编辑 extensions_support.conf 文件。

- IVR 6500：语音播报语音当前业务类型，并告知按 1 键进预约。

```
[ivr_6500]
exten = 6500,1,NoOp(6500)
exten = 6500,n,Set(TIMEOUT(digit)=3)
exten =
 6500,n,Background(/tmp/media/networkdisk1/notice) //notice.wav
 这里是 xxx 业务受理平台
exten =
 6500,n,Background(/tmp/media/networkdisk1/${EXTEN}) //以被叫手机号为命名的文件，每个被叫的语音都不一样。文件暂命名phone.wav。语音内容：为您预约3月12日科目二考试
exten =
 6500,n,Background(/tmp/media/networkdisk1/confirm}) //confirm.wav  同意请按 1
exten = 6500,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6500,n,WaitExten(1)
exten = 6500,n,Set(TIMEOUT(digit)=3)
exten =
 6500,n,Background(/tmp/media/networkdisk1/notice) //notice.wav
 这里是 XXX 业务受理平台
exten =
 6500,n,Background(/tmp/media/networkdisk1/${EXTEN}) //以被叫手机号为命名的文件，每个被叫的语音都不一样。文件暂命名test.wav。语音内容：为您预约3月12日科目二考试
exten =
 6500,n,Background(/tmp/media/networkdisk1/confirm}) //confirm.wav   同意请按1
exten = 6500,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6500,n,WaitExten(1)
exten = 6500,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = 6500,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = 6500,n,Goto(t,1)
exten = 1,1,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = 1,2,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = 1,3,ExecIf($[${CHANDISABLERECORD} = yes]?Noop(skip
 setting disablerecord again):Set(CHANDISABLERECORD=no))
exten = 1,4,Goto(ivrs,6501,1)
```

```
exten = unknown,1,NoOp(No Action)
```

- IVR6501：当客户听到语音提示后，按1，进入 IVR6501，可以输入短信验证码，输入完毕会继续听到相关语音，以及等待音乐。

```
[ivr_6501]
exten = 6501,1,NoOp(6501)
exten = 6501,n,Set(TIMEOUT(digit)=3)
exten =
 6501,n,Background(/tmp/media/networkdisk1/password) //password.w
av  请输入您收到约考短信验证码，按#号键结束
exten = 6501,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6501,n,WaitExten(4)
exten = 6501,n,Set(TIMEOUT(digit)=3)
exten =
 6501,n,Background(/tmp/media/networkdisk1/password) //password.w
av  请输入您收到约考短信验证码，按#号键结束
exten = 6501,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6501,n,WaitExten(4)
exten = 6501,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = 6501,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = _X.#,1,UserEvent(DtmfEndString,Info: ${EXTEN})
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/holdon) //holdon.wav
 请稍后
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/macroform-the_simplicity
) //播放等待音乐
exten = _X.#,n,Playback(/tmp/media/networkdisk1/holdon) //请稍后
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/macroform-the_simplicity
) //播放等待音乐
exten = _X.#,n,NoOp(exten:${EXTEN})
exten = _X.#,n,Goto(ivrs,6501,1)
exten = 6501,n,Goto(t,1)
exten = t,1,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = t,2,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = t,3,ExecIf($[${CHANDISABLERECORD} = yes]?Noop(skip
 setting disablerecord again):Set(CHANDISABLERECORD=no))
exten = t,4,Goto(always-Hangup,h,1)
exten = unknown,1,NoOp(No Action)
```

- IVR6502：当系统验证通过后，转 IVR6502，告知客户业务办理成功，并挂断电话。

```
[ivr_6502]
exten = 6502,1,NoOp(6502)
exten = 6502,n,Set(TIMEOUT(digit)=3)
exten =
 6502,n,Background(/tmp/media/networkdisk1/ok) //已经预约成功，再
见！
exten = 6502,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6502,n,WaitExten(1)
exten = 6502,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = 6502,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = 6502,n,Goto(t,1)
exten = t,1,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = t,2,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = t,3,ExecIf($[${CHANDISABLERECORD} = yes]?Noop(skip
 setting disablerecord again):Set(CHANDISABLERECORD=no))
exten = t,4,Goto(always-Hangup,h,1)
exten = unknown,1,NoOp(No Action)
```

- IVR6503：当短信验证失败，转 IVR6503，告知客户重新输入短信验证码，并且根据验证成功与否转 IVR6502 或 6503。

```
[ivr_6503]
exten = 6503,1,NoOp(6503)
exten = 6503,n,Set(TIMEOUT(digit)=3)
exten =
 6503,n,Background(/tmp/media/networkdisk1/password1) //password1
.wav验证码错误，请输入您收到约考短信验证码，按#号键结束
exten = 6503,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6503,n,WaitExten(4)
exten = 6503,n,Set(TIMEOUT(digit)=3)
exten =
 6503,n,Background(/tmp/media/networkdisk1/password1) //password1
.wav验证码错误，请输入您收到约考短信验证码，按#号键结束
exten = 6503,n,UserEvent(PlayPromptEnd,Ivrid: ${EXTEN})
exten = 6503,n,WaitExten(4)
exten = 6503,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = 6503,n,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
```

```
exten = _X.#,1,UserEvent(DtmfEndString,Info: ${EXTEN})
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/holdon) //holdon.wav请稍
后
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/macroform-the_simplicity
) //播放保留音乐
exten = _X.#,n,Playback(/tmp/media/networkdisk1/holdon) //请稍后
exten =
 _X.#,n,Playback(/tmp/media/networkdisk1/macroform-the_simplicity
) //播放保留音乐
exten = _X.#,n,NoOp(exten:${EXTEN})
exten = _X.#,n,Goto(ivrs,6503,1)
exten = 6503,n,Goto(t,1)
exten = t,1,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(Ivr2outbound)=yes))
exten = t,2,ExecIf($[${Ivr2outbound} =
 yes]?Set(CDR(INGORELOCAL1CDR)=yes))
exten = t,3,ExecIf($[${CHANDISABLERECORD} = yes]?Noop(skip
 setting disablerecord again):Set(CHANDISABLERECORD=no))
exten = t,4,Goto(always-Hangup,h,1)
exten = unknown,1,NoOp(No Action)
```