

# API 说明文档

时间：2018 年 12 月 21 日 17:00:00 版本：V2.0

## 历史记录

时间	版本	作者	说明
2017 年 4 月 13 日	V1.0	Larry	创建
2017 年 7 月 11 日	V1.1	Larry	新增挂断呼叫的 API
2017 年 8 月 29 日	V2.0.6	LARRY	
2018 年 9 月 3 日	V2.0.6	Amier	
2018 年 12 月 12 日	V2.0.6	Amier	
2018 年 12 月 21 日	V2.0.6	Amier	新增任务状态推送的 API

## 目录

一、 名词说明.....	7
二、 错误码.....	7
三、 说明.....	9
1. API 请求消息认证.....	9
2. 通用 API 中推送消息的认证.....	10
3. 请求格式.....	10
4. 响应格式.....	11
四、 语音验证码.....	12
1. 描述: .....	12
2. URL: .....	12
3. 消息格式: .....	12
4. 参数说明: .....	12
5. 响应格式: .....	13
6. 响应说明: .....	13
五、 查询坐席状态.....	14
1. 描述.....	14
2. URL.....	14
3. 消息格式.....	14
4. 参数说明.....	15
5. 响应格式.....	15
6. 响应说明.....	16
六、 直接获取坐席状态.....	16
1. 描述.....	16
2. URL.....	16
3. 消息格式.....	16
4. 参数说明.....	17
5. 响应格式.....	18
6. 响应说明.....	18
7. 坐席登录.....	22
1. 描述.....	22
2. URL.....	22
3. 消息格式.....	22
4. 参数说明.....	23
5. 响应格式.....	23
6. 响应说明.....	24
8. 坐席退出登录.....	24
1. 描述.....	24
2. URL.....	24
3. 消息格式.....	24
4. 参数说明.....	25
5. 响应格式.....	26

6. 响应说明.....	26
9. 坐席置忙.....	27
1. 描述.....	27
2. URL.....	27
3. 消息格式.....	27
4. 参数说明.....	27
5. 响应格式.....	28
6. 响应说明.....	28
10. 坐席置闲.....	29
1. 描述.....	29
2. URL.....	29
3. 消息格式.....	29
4. 参数说明.....	30
5. 响应格式.....	30
6. 响应说明.....	31
11. 坐席点击呼叫.....	31
1. 描述.....	31
2. URL.....	31
3. 消息格式.....	31
4. 参数说明.....	32
5. 响应格式.....	33
6. 响应说明.....	33
12. 坐席挂断呼叫.....	34
1. 描述.....	34
2. URL.....	34
3. 消息格式.....	34
4. 参数说明.....	34
5. 响应格式.....	35
6. 响应说明.....	35
13. 创建外呼任务.....	36
1. 描述.....	36
2. URL.....	36
3. 消息格式.....	36
4. 参数说明.....	37
5. 响应格式.....	38
6. 响应说明.....	38
14. 外呼任务查询.....	39
1. 描述.....	39
2. URL.....	39
3. 消息格式.....	39
4. 参数说明.....	40
5. 响应格式.....	41
6. 响应说明.....	41
15. 外呼任务控制.....	42

1. 描述.....	42
2. URL.....	42
3. 消息格式.....	42
4. 参数说明.....	43
5. 响应格式.....	43
6. 响应说明.....	43
16. 外呼任务追加号码.....	44
1. 描述.....	44
2. URL.....	44
3. 消息格式.....	44
4. 参数说明.....	45
5. 响应格式.....	46
6. 响应说明.....	46
17. 呼叫接通通知.....	47
1. 描述.....	47
2. URL.....	47
3. 消息格式.....	47
4. 消息说明.....	48
5. 响应格式.....	48
6. 响应说明.....	49
18. 通话结束通知.....	49
1. 描述.....	49
2. URL.....	49
3. 消息格式.....	49
4. 消息说明.....	50
5. 响应格式.....	50
6. 响应说明.....	51
19. 呼叫弹屏通知.....	51
1. 描述.....	51
2. URL.....	51
3. 消息格式.....	51
4. 消息说明.....	52
5. 响应格式.....	52
6. 响应说明.....	52
20. 获取录音列表.....	53
1. 描述.....	53
2. URL.....	53
3. 消息格式.....	53
4. 参数说明.....	54
5. 响应格式.....	54
6. 响应说明.....	55
7. 获取录音文件.....	55
21. 通知呼入.....	55
1. 描述.....	55

2. URL.....	55
3. 消息格式.....	56
4. 消息说明.....	56
5. 响应格式.....	57
6. 响应说明.....	57
22. 通知呼出.....	57
1. 描述.....	57
2. URL.....	57
3. 消息格式.....	57
4. 消息说明.....	58
5. 响应格式.....	58
6. 响应说明.....	59
23. 呼叫结束通知话单.....	59
1. 描述.....	59
2. URL.....	59
3. 消息格式.....	59
4. 消息说明.....	60
5. 响应格式.....	61
6. 响应说明.....	61
24. 请求获取话单.....	61
1. URL.....	61
2. 消息格式.....	61
3. 消息说明.....	62
4. 响应格式.....	63
5. 响应说明.....	65

# 公共 API 文档

## 一、 名词说明

1. 呼叫中心：SCCC 系统；
2. 呼叫平台：SCCC 系统；
3. 业务系统：客户业务系统；
4. 业务平台：客户业务系统；
5. 业务平台客户端：客户业务系统客户端；
6. 业务系统客户端：客户业务系统客户端；
7. 请求（消息）：由客户服务器向 SCCC 系统发送的 HTTP 请求；
8. 推送（消息）：由 SCCC 系统向客户服务器发送的 HTTP 请求；
9. 通用 API：SCCC 系统提供一些基础功能的 API；
10. 专用 API：SCCC 为客户开发的定制 API，可以覆盖专用 API；
11. 请求序列号：HTTP 请求的序列号，主要用来生成认证域的，响应中会携带；
12. 命令序列号：每个命令的序列号，主要区别命令，同时在异步推送的时候，会回传回去；

## 二、 错误码

名称	值	说明	备注
E_FAIL	-1	未知错误	
E_SUCC	0	成功	
E_S_AUTH_FAIL	1	认证失败	
E_S_SESS_TIMEOUT	2	SESSION 超时	API 系统中可能存在 token 超时的可能
E_S_SESS_BREAKEN	3	SESSION 被破坏	
E_W_DB_OPT_FAIL	4	数据库操作失败	
E_C_INVALID_PARAM	5	参数错误	
E_S_ACCESS_FORBIDDEN	6	访问受限	

E_C_CONFIG_ERROR	7	配置错误	
E_C_INVALID_MSG	8	请求数据格式错误	
E_C_UNSUPPORTED_API_VERSION	10	不支持的 API 版本	URL 中版本号或者版本号格式填写错误
E_C_UNSUPPORTED_API_REQUEST	11	不支持的 API 请求	URL 中接口名称或者 URL 格式错误
E_G_INVALID_PARAM	0xF1000001	参数错误	
E_G_DB_OPT_FAIL	0xF1000002	数据库操作失败	
E_G_RES_IN_USE	0xF1000003	资源正在被使用	
E_G_RES_NAME_EXIST	0xF1000004	资源名称被使用	
E_G_BILLING_RATE_CONFLICT	0xF1000005	计费费率冲突	在操作计费规则时可能会产生
E_G_INTER_ERROR	0xF1000006	内部错误	SCCC 系统中出现错误
E_G_QUERY_FAIL	0xF1000007	查询失败	一般使用 E_G_DB_OPT_FAIL 代替
E_G_RES_NOT_EXIST	0xF1000008	资源不存在	
E_G_RES_ALLOC_FAIL	0xF1000009	申请资源失败	
E_G_EXEC_FAIL	0xF100000A	模块之间通讯失败	
E_G_PASSWORD_ERROR	0xF100000B	密码错误	
E_G_BALANCE_TOO_LARGE	0xF100000C	余额超出系统范围	账户的最大余额为 100000000
E_G_MONEY_TOO_LARGE	0xF100000D	单次操作金额太大	单次操作金额最大 100000
E_G_INVALID_OPT	0xF100000E	非法操作	
E_G_BALANCE_LACK	0xF100000F	余额不足	
E_G_READ_FILE_FAIL	0xF1000010	文件读取失败	



E_G_FILE_NOT_EXIST	0xF1000011	文件不存在	
E_G_UPLOAD_FAIL	0xF1000012	上传失败	
E_G_STAFF_NO_EXIST	0xF1000013	指定工号不存在	
E_G_CONFIG_ERROR	0xF1000014	配置错误	
E_G_NUMBER_EXIST	0xF1000015	号码已经存在	
E_G_DATASYN_FAIL	0xF1000016	数据同步失败	
E_G_NUMBER_INVALID	0xF1000017	号码非法	
E_G_DID_NO_EXIST	0xF1000018	DID 号码不存在	
E_G_FEE_ERROR	0xF1000019	费率错误	
E_G_INVALID_MSG	0xF1000020	指定格式解析数据 失败	
E_G_API_TO_FREQUENTLY	0xF1000021	API 请求太频繁	

注：有些错误码为系统内部错误码，API 响应接口中并不会出现。

### 三、 说明

#### 1. API 请求消息认证

请求消息体中必须携带 authentication 字段，该字段包括如下内容：

名称	必填	说明	值说明
customer	Y	客户名称，客户在申请 API 时，会生产。	[A-Za-z0-9-_,]
timestamp	Y	时间戳	毫秒级别时间戳
seq	Y	消息序列号，保证一段时间内序列号不重复即可	整数
digest	Y	认证域	

digest 域计算方法：

客户在申请开通 API 时，运营商创建客户 API 相关数据，包括请求认证用户名，密码，推送消息的目的 IP 地址端口，推送消息的用户名密码有效期。其中我们需要将用户名和密码两

个字段发送给客户。

计算 digest 算法如下：

MD5(<customer> + "@" + <timestamp> + "@" + <seq> + "@" + 密码)

如：

客户申请的用户名和密码如下：

客户名称：C2

密码：E416D0F85DC0CB67C85E070A32D9F0123AFF4762

需要发送的消息如下

```
{
  authentication: { customer: "C2", timestamp: 1471918933001, seq: 5271, digest: "", agent: "" },
  request: { ..... }
}
```

这时 digest 字段为 下面这个字符串的 MD5 校验和：

C2@1471918933001@5271@E416D0F85DC0CB67C85E070A32D9F0123AFF4762

$\Rightarrow$  MD5(C2@1471918933001@5271@E416D0F85DC0CB67C85E070A32D9F0123AFF4762)

**注：密码非常重要，不能在任何信息中直接传输，否则可能导致致命安全问题。**

## 2. 通用 API 中推送消息的认证

我们在发送推送消息时，首先使用推送消息的用户名和密码，如果有其中任何一个没有定义，将使用请求认证用户名密码安装上面的算法计算 digest

## 3. 请求格式

1. `http://ip:port/openapi/V2.0.x/ApiName` （版本号 V2.0.x ，根据当前使用的版本号决定，比如当前是用是 V2.0.1，url 为：`http://ip:port/openapi/V2.0.x/ApiName`）
2. 所有内容必须是 UTF-8 无 Bom 格式；
3. 所有内容必须是合法的 json 格式；后续可能开发 xml 格式接口；
4. 请求的数据域可以携带如下内容

```
“param”:{
    “debug”: bool,
    “lang”: “zh_CN|en_US”
}
```

其中 `debug` 表示，开启调试，`lang` 表示响应中的语种

#### 5. 请求和命令

一个请求即一个 HTTP 请求，一个请求中可以携带一个或者多个命令，如果携带一个命令即 HTTP 消息中为 `request` 域，如果携带多个命令这时 HTTP 消息中为 `requests` 域，每个命令携带自己的序列号；一个请求中只能携带同类型的命令。

#### 6. http 请求头中必须携带如下信息

**Accept: application/json**

**Content-Type: application/json**

## 4. 响应格式

一般情况响应都会如下格式:

```
{
  "data": {
    "seq": "",
    "response": {
      "seq": "",
      "userData": "",
      "result": {
        "error": "", "msg": ""
      }
    }
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

Result 域描述请求执行情况，logs 域携带日志信息，data 域携带每个命令执行情况；

## 四、 语音验证码

### 1. 描述：

客户业务平台需要审核某一电话号码确实可以呼叫，即可使用该 API。用户将自己的号码发送给业务平台，业务平台生成好验证码，并生成请求发送到呼叫平台，呼叫平台根据请求向被审核的号码呼叫，呼叫接通之后播放验证码。用户将听到的验证码填写，并提交到业务服务器，业务服务器对比用户提交的验证码和之前生成的验证码，如果一致即可审核通过。

### 2. URL：

http://ip:port/openapi/V2.0.x/VoiceVerify (x=1,2,3,4,6)

### 3. 消息格式：

```
{
  "authentication": {
    "customer": "C2",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "callee": "",
    "playTimes": "",
    "seq": "",
    "verifyCode": "",
    "userData": ""
  }
}
```

### 4. 参数说明：

Param 域说明：（可选）

字段名	必填	说明	值说明
Debug	N	是否开启调试	Bool
Lang	N	响应中的语种	zh_CH en_US

Request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
verifyCode	Y	验证码	Number length[3-8]
playTimes	N	播放次数	Number 1-5 default:3
callee	Y	被叫号码	string(合法的电话号码)
userData	N	用户数据, 异步响应会回传	string 最长 100 的字符, 如果超过 100, 将截断

**userData 说明:** 用户可以在 **userData** 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式:

```
{
  "data": {
    "seq": ""
    "response": { "seq": "", "userData": "", "result": "" },
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

**注:** Error 为 0 标识成功, 但只能标识系统已经收到该请求, 并且已经准备开始发起呼叫了。

## 6. 响应说明:

Logs 域为日志域, 如果开启了调试会携带相关调试信息

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

Data 域说明

字段名	必填	说明	值说明
-----	----	----	-----

seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 response 域的数组。

## 五、 查询坐席状态

### 1. 描述

通过 API 查询坐席状态, 状态信息会通过推送消息, 异步推送给对端

### 2. URL

<http://ip:port/openapi/V2.0.x/AgentQueryStatus> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  }
}
```

```

    },
    "request": {
        "agent": "",
        "seq": "",
        "userData": "",
    }
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

pram 域说明: [见 3.2](#)

rquest 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```

{
    "data": {
        "seq": ""
        "response": {"seq": "", "userData": "", "result":{"error":0}},
    },
    "logs": "",
    "result": {"error": "", "msg": ""}
}

```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经开始处理，请等待呼叫中心系统主动推送状态。

## 6. 响应说明

Logs 域未日志域，如果开启了调试会携带相关调试信息

Result 域说明：见 3.4

Data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 六、 直接获取坐席状态

### 1. 描述

通过 API 获取坐席状态，最新状态会直接通过响应数据返回

### 2. URL

<http://ip:port/openapi/V2.0.x/getAgentStatus> (x=1,2,3,4,6)

### 3. 消息格式

{



```

"authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
},
"param": {
    "debug": "true",
    "lang": "en_US"
},
"request": {
    "agent": "1000",
    "seq": "",
    "userData": ""
}
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

pram 域说明: [见 3.2](#)

rquest 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{
  "data": {
    "seq": ""
    "response": {
      "seq": "",
      "userData": "",
      "result":{"error":0}
      "workStatus": "ONLINE",
      "serverStatus": "IDLE",
    },
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

注：Error 为 0 标识成功，接口会直接返回坐席状态，状态表示枚举值见下表

## 6. 响应说明

Logs 域未日志域，如果开启了调试会携带相关调试信息

Result 域说明：[见 3.4](#)

Data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据

userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object
workStatus	Y	工作状态	OFFLINE 离线 , ONLINE 在线, BUSY 忙, AWAY 离开
serverStatus	Y	业务状态	IDLE 空闲 CALL_OUT 呼出接通 CALL_IN 呼入已接通 RINGING 振铃 PROCESS 整理中

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 七、 查询坐席信息

### 7. 描述

通过 API 获取坐席的工号名称信息

### 8. URL

<http://ip:port/openapi/V2.0.x/getAgentInfo> (x=6)

### 9. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  }
}
```

```

    },
    "param": {
        "debug": "true",
        "lang": "en_US"
    },
    "request": {
        "seq": "",
        "userData ": "",
    }
}

```

## 10. 参数说明

authentication 域说明: [见 3.1](#)

pram 域说明: [见 3.2](#)

rquest 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道时那个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 11. 响应格式

```

{
    "data": {
        "seq": ""
        "response": {
            "seq": "",
            "userData": "",
            "result":{"error":0}
        }
    }
}

```

```

    "data": [
        {id:"", name:"", staffNo:""},
        {id:"", name:"", staffNo:""},
        {id:"", name:"", staffNo:""},
        .....
    ]
},
},
"logs": "",
"result": { "error": "", "msg": "" }
}

```

注：Error 为 0 标识成功，接口会直接返回坐席状态，状态表示枚举值见下表

## 12. 响应说明

Logs 域为日志域，如果开启了调试会携带相关调试信息

Result 域说明：见 3.4

Data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object
data	Y	坐席信息数据	Array

data 字段:

字段名	必填	说明	值说明
id	Y	坐席 ID 标识	
staffNo	Y	坐席工号	
name	Y	坐席名称	

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 responses 域的数组。

## 7. 坐席登录

### 1. 描述

业务系统的业务员(坐席)登录业务系统时, 业务系统需要向呼叫中心平台通知业务员登录, 一边呼叫系统确定业务员的状态。

### 2. URL

<http://ip:port/openapi/V2.0.x/AgentLogin> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  }
}
```

```

    },
    "request": {
        "agent": "",
        "seq": "",
        "userData": "",
    }
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

pram 域说明: [见 3.2](#)

rquest 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```

{
    "data": {
        "seq": ""
        "response": { "seq": "", "userData": "", "result":{"error":0}},
    },
    "logs": "",
    "result": { "error": "", "msg": "" }
}

```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经开始处理，请等待呼叫中心系统主动推送状态。

## 6. 响应说明

Logs 域未日志域，如果开启了调试会携带相关调试信息

Result 域说明：见 3.4

Data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 8. 坐席退出登录

### 1. 描述

业务系统的业务员（坐席）退出业务系统时，业务系统需要向呼叫中心平台通知业务员登录，一边呼叫系统确定业务员的状态。

### 2. URL

<http://ip:port/openapi/V2.0.x/AgentLogout> (x=1,2,3,4,6)

### 3. 消息格式

{



```

"authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
},
"param": {
    "debug": "true",
    "lang": "en_US"
},
"request": {
    "agent": "",
    "seq": "",
    "userData": ""
}
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

**userData 说明:** 用户可以在 **userData** 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{  
  "data": {  
    "seq": ""  
    "response": { "seq": "", "userData": "", "result":{"error": ""}},  
  },  
  "logs": "",  
  "result": { "error": "", "msg": "" }  
}
```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经准备开始处理。

## 6. 响应说明

logs 域为日志域，如果开启了调试会携带相关调试信息

result 域说明：[见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 9. 坐席置忙

### 1. 描述

坐席将自己置忙，暂时不接受呼叫。

### 2. URL

<http://ip:port/openapi/V2.0.x/AgentSetBusy> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "userData": "",
    "agent": ""
  }
}
```

### 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{  
  "data": {  
    "seq": ""  
    "response": { "seq": "", "userData": "", "result": "" },  
  },  
  "logs": "",  
  "result": { "error": "", "msg": "" }  
}
```

注: Error 为 0 标识成功, 但只能标识系统已经收到该请求, 并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域为日志域, 如果开启了调试会携带相关调试信息

result 域说明: [见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 responses 域的数组。

## 10. 坐席置闲

### 1. 描述

坐席将自己置闲, 可以接受呼叫。

### 2. URL

<http://ip:port/openapi/V2.0.x/AgentSetIdle> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  }
}
```

```

    },
    "request": {
        "seq": "",
        "userData": "",
        "agent": ""
    }
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]

**userData 说明:** 用户可以在 **userData** 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```

{
    "data": {
        "seq": ""
        "response": { "seq": "", "userData": "", "result": "" },
    },
    "logs": "",
    "result": { "error": "", "msg": "" }
}

```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域未日志域，如果开启了调试会携带相关调试信息

result 域说明：见 3.4

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 11. 坐席点击呼叫

### 1. 描述

业务系统通过发送该 API 将发起呼叫，先呼叫触发该 API 的坐席，再呼叫指定的被叫号码；

### 2. URL

<http://ip:port/openapi/V2.0.x/CallNumber> (x=1,2,3,4,6)

### 3. 消息格式

{

```

"authentication": {
  "customer": "C2",
  "digest": "",
  "seq": "5271",
  "timestamp": "1471918933001"
},
"param": {
  "debug": "true",
  "lang": "en_US"
},
"request": {
  "seq": "",
  "userData": "",
  "agent": "",
  "callee": ""
}
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]
callee	Y	被叫号码, 需要呼叫的号码	Number

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与



坐席相关的呼叫，用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{
  "data": {
    "seq": ""
    "response": { "seq": "", "userData": "", "result": "" },
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域为日志域，如果开启了调试会携带相关调试信息

result 域说明：[见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果，通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为

responses 域的数组。

## 12. 坐席挂断呼叫

### 1. 描述

业务系统通过发送该 API 将发起呼叫，先呼叫触发该 API 的坐席，再呼叫指定的被叫号码；

### 2. URL

<http://ip:port/openapi/V2.0.x/HangupCall> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "agent": "",
    "userData": ""
  }
}
```

### 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
agent	Y	坐席工号	Number (1-8)
userData	Y		String(30)

## 5. 响应格式

```
{  
  "data": {  
    "seq": ""  
    "response": { "seq": "", "result": "" },  
  },  
  "logs": "",  
  "result": { "error": "", "msg": "" }  
}
```

注: Error 为 0 标识成功, 但只能标识系统已经收到该请求, 并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域为日志域, 如果开启了调试会携带相关调试信息

result 域说明: [见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
-----	----	----	-----

seq	Y	命令序列号	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object

注：如果一个请求中携带了多个请求，则 response 域，将被替换为 responses 域，结果为 responses 域的数组。

## 13. 创建外呼任务

### 1. 描述

给指定坐席创建呼叫任务

### 2. URL

<http://ip:port/openapi/V2.0.x/CallTaskCreate> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "userData": ""
  }
}
```

```

    "agent": "",
    "taskName": "",
    "callers": ["", "", "" .....],
    "callees": [ "", "", "" ..... ],
  }
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number length[1-8]
taskName	N	任务名称	给任务创建提供名称, 不填或没有该字段系统会自动生成
callees	Y	被叫号码列表	Array 最多 1000 个
callers	N	主叫号码列表	Array 最多 100 个
mode	Y	呼叫模式	Number(2 5)
multiplier	N	呼叫倍率	Number(1<a<5) 不填或没有该字段默认为 1

Mode 域说明:

填写 2, 呼叫任务会先呼叫客户, 客户接通后再呼叫坐席;

填写 5, 呼叫会先呼叫坐席, 在呼叫客户;

Callers 域说明:

如果 Mode 域为 5, callers 域将不会被处理, 呼叫过程中会根据呼叫设定选择主叫号码;

如果该字段, 不填写, 系统会根据已有的主叫设定通过坐席去匹配主叫号码或号码组;

如果没有相关的主叫设定, 系统将随便选择一个号码组, 作为任务的主叫来源; (此时, 如果系统中不存在呼叫号码组, 该 API 会执行失败)

如果该字段填写主叫号码，系统会判断这些主叫号码是否是该客户的，或者有主叫号码处于不可用状态，该 API 将执行失败。

**userData** 说明：用户可以在 **userData** 中携带一些用户自定义的数据。如对端系统是多用户的，客户把用户的唯一标识传过来。我们回传回去，业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫，用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{
  "data": {
    "seq": ""
    "response": { "seq": "", "userData": "", "result": "", "taskID": "",
    "report": {
      "total": "",
      "error": "",
      "repeat": "",
    },
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

**注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经准备开始发起呼叫了。**

## 6. 响应说明

**logs** 域为日志域，如果开启了调试会携带相关调试信息

**result** 域说明：见 [3.4](#)

**data** 域说明

字段名	必填	说明	值说明
-----	----	----	-----

seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object
taskID	Y	任务 ID	用于业务服务器记录, 以后操作任务就要靠这个 ID

report 域说明: 号码导入结果汇报

字段名	必填	说明	值说明
total	Y	号码总数	Number
error	Y	错误数	Number
report	Y	重复数	Number

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 responses 域的数组。

## 14. 外呼任务查询

### 1. 描述

查询指定业务员是否有对应的外呼任务

### 2. URL

<http://ip:port/openapi/V2.0.x/CallTaskQuery> (x=1,2,3,4,6)

### 3. 消息格式

{

```

"authentication": {
    "customer": "C2",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
},
"param": {
    "debug": "true",
    "lang": "en_US"
},
"request": {
    "seq": "",
    "agent": "",
    "userData": ""
}
}

```

#### 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
agent	Y	坐席工号	Number

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。



## 5. 响应格式

```
{  
  "data": {  
    "seq": ""  
    "response": { "seq": "", "userData": "", "result": "", "taskID": ""},  
  },  
  "logs": "",  
  "result": { "error": "", "msg": "" }  
}
```

注: Error 为 0 标识成功, 但只能标识系统已经收到该请求, 并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域为日志域, 如果开启了调试会携带相关调试信息

result 域说明: [见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object
taskID	Y	任务 ID	用于业务服务器记录

注：如果一个请求中携带了多个请求，则 `response` 域，将被替换为 `responses` 域，结果为 `responses` 域的数组。

## 15. 外呼任务控制

### 1. 描述

呼叫任务控制；

### 2. URL

<http://ip:port/openapi/V2.0.x/CallTaskControl> (x=1,2,3,4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "SCCC-test",
    "agent": "1000",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "agent": "",
    "userData": "",
    "taskID": "",
    "action": ""
  }
}
```

```
}
```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
taskID	Y	任务 ID	Number
action	Y	控制选项	Start(开始任务) stop(停止任务)

userData 说明: 用户可以在 userData 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{  
  "data": {  
    "seq": ""  
    "response": { "seq": "", "userData": "", "result": "", "taskID": "" },  
  },  
  "logs": "",  
  "result": { "error": "", "msg": "" }  
}
```

**注: Error 为 0 标识成功, 但只能标识系统已经收到该请求, 并且已经准备开始发起呼叫了。**

## 6. 响应说明

logs 域为日志域, 如果开启了调试会携带相关调试信息

result 域说明: [见 3.4](#)

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段:

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据
userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object
taskID	Y	任务 ID	用于业务服务器记录

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 responses 域的数组。

## 16. 外呼任务追加号码

### 1. 描述

呼叫任务追加号码

### 2. URL

<http://ip:port/openapi/V2.0.x/CallTaskAddNumbers> (x=1,2,3,4,6)

### 3. 消息格式

```
{  
  "authentication": {  
    "customer": "C2",  
    "agent": "1000",
```

```

    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "userData": "",
    "taskID": "",
    "agent": "",
    "callees": ["", ...],
  }
}

```

## 4. 参数说明

authentication 域说明: [见 3.1](#)

param 域说明: [见 3.2](#)

request 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号, 推送消息时使用	Number
userData	N	用户数据	String(30)
taskID	Y	任务 ID	Number
agent	Y	坐席工号	Number
callees	Y	被叫号码	添加的号码 Array 最多 1000 个

**userData 说明:** 用户可以在 `userData` 中携带一些用户自定义的数据。如对端系统是多用户的, 客户把用户的唯一标识传过来。我们回传回去, 业务系统就可以知道是哪个客户的。再如与坐席相关的呼叫, 用户传过来的数据就是坐席的唯一标识。

## 5. 响应格式

```
{
  "data": {
    "seq": ""
    "response": { "seq": "", "userData": "", "result": "", "taskID": "", "report": {
      "total": "",
      "error": "",
      "repeat": "",
    }
  },
  "logs": "",
  "result": { "error": "", "msg": "" }
}
```

注：Error 为 0 标识成功，但只能标识系统已经收到该请求，并且已经准备开始发起呼叫了。

## 6. 响应说明

logs 域为日志域，如果开启了调试会携带相关调试信息

result 域说明：见 3.4

data 域说明

字段名	必填	说明	值说明
seq	Y	请求的序列号	Int
response	Y	对请求中每一个命令	

response 字段：

字段名	必填	说明	值说明
seq	Y	命令序列号	回传数据

userData	N	用户数据	回传数据
result	Y	结果, 通常的响应接口(error 和 msg 两个字段)	Object
taskID	Y	任务 ID	用于业务服务器记录

report 域说明: 号码导入结果汇报

字段名	必填	说明	值说明
total	Y	号码总数	Number
error	Y	错误数	Number
report	Y	重复数	Number

注: 如果一个请求中携带了多个请求, 则 response 域, 将被替换为 responses 域, 结果为 responses 域的数组。

## 17. 呼叫接通通知

### 1. 描述

如果客户的请求引起系统发起了呼叫, 在呼叫接通时将会给对端服务器推送相关数据。

客户服务器可以通过请求中的回传 userData (见 userData 说明), 用户标识, 区分是哪个请求的通知。

### 2. URL

由用户提供并配置到系统中

### 3. 消息格式

```
{
  "authentication": {
    "customer": "",
    "digest": "",
    "seq": "",
    "timestamp": ""
  }
}
```

```

    },
    "notify": {
        "type": "callstart",
        "customer": "",
        "agent": "",
        "number": "",
        "call_type": "",
        "seq": "",
        "userData": ""
    },
    "param": {
        "debug": "",
        "lang": ""
    }
}

```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
customer	Y	客户	
agent	Y	坐席工号	
number	Y	号码	
call_type	Y	呼叫类型	0 表示呼入, 1 表示呼出
type	Y	Api 类型, callstart	

## 5. 响应格式

```
{
```



```
"result": { "error": "", "msg": "" }  
}
```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 18. 通话结束通知

### 1. 描述

如果客户请求发起了呼叫，在呼叫结束之后将会给对端服务器推送呼叫过程中的相关数据。客户服务器可以通过请求中回传的回传 `userData`（见 `userData` 说明），区分是哪个请求的通知。

### 2. URL

由用户提供并配置到系统中

### 3. 消息格式

```
{  
  "authentication": {  
    "customer": "",  
    "digest": "",  
    "seq": "",  
    "timestamp": ""  
  },  
  "notify": {  
    "type": "callend",
```

```

    "customer": "",
    "agent": "",
    "number": "",
    "call_type": "",
    "seq": "",
    "userData": ""
  },
  "param": {
    "debug": "",
    "lang": ""
  }
}

```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
type	Y	Api 类型, callend	
agent	Y	坐席工号	
customer	Y	对接客户	
number	Y	号码	
call_type	Y	呼叫类型	0 表示呼入, 1 表示呼出

## 5. 响应格式

```

{
  "result": { "error": "", "msg": "" }
}

```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 19. 呼叫弹屏通知

### 1. 描述

客户侧开始振铃时，通知 API 方，方便客户弹屏使用

### 2. URL

由用户提供并配置到系统中

### 3. 消息格式

```
{  
  "authentication": {  
    "customer": "",  
    "digest": "",  
    "seq": "",  
    "timestamp": ""  
  },  
  "notify": {  
    "customer" => ,  
    "agent" => "",  
    "number" => "",  
    "call_type" => "",  
    "seq" => "",  
  },  
}
```

```

    "param": {
        "debug": "",
        "lang": ""
    }
}

```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
customer	Y	客户 ID	
agent	N	坐席工号	
number	Y	客户号码	
Call_type	Y	1 表示 呼出, 0 表示呼入	
Seq	Y	序列号	

## 5. 响应格式

```

{
    "result": { "error": "", "msg": "" }
}

```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 20. 获取录音列表

### 1. 描述

通过该接口获取指定时间段内的所有录音话单，并指定录音文件的路径，以便客户下载。

### 2. URL

<http://ip:port/openapi/V2.0.x/getRecordsList> (x=4,6)

### 3. 消息格式

```
{
  "authentication": {
    "customer": "C2",
    "digest": "",
    "seq": "5271",
    "timestamp": "1471918933001"
  },
  "param": {
    "debug": "true",
    "lang": "en_US"
  },
  "request": {
    "seq": "",
    "startKey": ""
  }
}
```

## 4. 参数说明

参数	类型	必填	说明
seq	String	Y	
startKey	Int	Y	起始 id

注：id 等于 startKey 的记录不会包含在结果里面

## 5. 响应格式

```
{
  "result": {
    "error": 0,
    "msg": "成功"
  },
  "data": {
    "seq": "5271",
    "response": {
      "seq": "1",
      "userData": null,
      "result": {
        "error": 0
      }
    },
    "record": [
      {
        "key": "1",
        "caller": "10086",
        "callee": "26104935",
        "filename": "",
        "type": "4",
        "agent": "",
        "startTime": "1505992923",
        "duration": "330"
      }
    ]
  },
  "logs": null
}
```

## 6. 响应说明

数据域为一个数组，数组中的每个成员为一个对象，表示一条录音记录，对象各个字段说明：

参数	类型	必填	说明
key	String	Y	唯一标示 录音 ID
agent	String	Y	坐席工号，如果需要分机号，请将坐席分机号和工号配置为相同
caller	String	Y	主叫号码
callee	String	Y	被叫号码
type	Int	Y	呼入 0 呼出 1
startTime	Int	Y	呼叫开始时间
duration	Int	Y	通话时长
filename	String	Y	录音文件名路径

每次返回的最大录音条数为 500 条，如果录音获取的需求量更多，可以通过每条数据的 Key 的最大值作为下一次获取请求的 startKey；

## 7. 获取录音文件

当客户拿到录音文件列表之后，便可以下载录音。录音文件的 URL 为：

<http://ip:port/<filename>>

其中 filename 为响应数据中的 filename 字段

## 21. 通知呼入

### 1. 描述

发起请求后，呼叫坐席侧时通知

### 2. URL

由用户提供 url

### 3. 消息格式

```
{  
  "authentication": {  
    "customer": "",  
    "digest": "",  
    "seq": "",  
    "timestamp": ""  
  },  
  "notify": {  
    "type": "callin",  
    "customer": "",  
    "agent": "",  
    "number": "",  
    "call_type": "",  
    "seq": "",  
    "userData": ""  
  },  
  "param": {  
    "debug": "",  
    "lang": ""  
  }  
}
```

### 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
type	Y	Api 类型, callin	



agent	Y	坐席工号	
customer	Y	对接客户	
number	Y	号码	
call_type	Y	呼叫类型	0 表示呼入，1 表示呼出

## 5. 响应格式

```
{
  "result": { "error": "", "msg": "" }
}
```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 22. 通知呼出

### 1. 描述

发其请求后，坐席侧呼出时通知

### 2. URL

由用户提供 URL

### 3. 消息格式

```
{
  "authentication": {
    "customer": "",
    "digest": "",
    "seq": "",
    "timestamp": ""
  }
}
```

```

    },
    "notify": {
        "type": "callout",
        "customer": "",
        "agent": "",
        "number": "",
        "call_type": "",
        "seq": "",
        "userData": ""
    },
    "param": {
        "debug": "",
        "lang": ""
    }
}

```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
type	Y	Api 类型, callout	
agent	Y	坐席工号	
customer	Y	对接客户	
number	Y	号码	
call_type	Y	呼叫类型	0 表示呼入, 1 表示呼出

## 5. 响应格式

```
{
```

```
"result": { "error": "", "msg": "" }  
}
```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 23. 呼叫结束通知话单

### 1. 描述

如果客户请求发起了呼叫，在呼叫结束之后将会给对端服务器推送呼叫过程中的相关数据。

客户服务器可以通过请求中回传的 `userData`，用户标识，区分是哪个请求的通知。

### 2. URL

由用户提供 URL

### 3. 消息格式

```
{  
  "authentication": {  
    "customer": "",  
    "digest": "",  
    "seq": "",  
    "timestamp": ""  
  },  
  "notify": {  
    "type": "billing",  
    "startTime": "",  
    "ringTime": "",  
  }  
}
```

```

    "answerTime": "",
    "byeTime": "",
    "callee": "",
    "caller": "",
    "recordFile": "",
    "service": "",
    "session": "",
    "seq": "",
    "userData": ""
  },
  "param": {
    "debug": "",
    "lang": ""
  }
}

```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
type	Y	Api 类型, billing	
startTime	Y	呼叫开始时间	
ringTime	Y	振铃开始时间	
answerTime	Y	呼叫接通时间	
byeTime	Y	呼叫结束时间	
callee	Y	被叫号码	
caller	Y	主叫号码	
recordFile	Y	录音文件	

service	Y	业务类型	
session	Y	通话表示	
timeLength	Y	通话时长	

## 5. 响应格式

```
{
  "result": { "error": "", "msg": "" }
}
```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	

## 24. 请求获取话单

客户向呼叫中心发送请求，来获取话单

### 1. URL

<http://ip:port/openapi/V2.0.x/getCdrList> （版本号 V2.0.x ，根据当前使用的版本号决定，比如当前是用是 V2.0.1，url 为: <http://ip:port/openapi/V2.0.1/getCdrList>）

### 2. 消息格式

```
{
  "authentication": {
    "customer": "",
    "digest": "",
    "seq": ""
  }
}
```

```

        "timestamp": ""
    },
    "param": {
        "debug": "true",
        "lang": "en_US"
    },
    "request": {
        "userData": "123",
        "seq": "123",
        "agent": ["8000", .....],
        "serviceType": "",
        "startTime": "",
        "endTime": "",
        "caller": "",
        "callee": "",
        "timeLengthMin": "",
        "timeLengthMax": ""
    }
}

```

### 3. 消息说明

request 域说明:

字段名	必填	说明	值说明
userData	N	回传字段	
seq	Y	标识	
agent	Y	坐席工号	
serviceType	Y	业务类型	1 出局呼叫 2 入局呼叫 3 内部呼叫

			4 自动外呼 5 预览外呼 7 录音业务 9 转接业务 17 租金业务 18 结算业务 19 语音验证码业务
startTime	Y	开始时间（这两个时间都不填默认查当天话单数据）	
endTime	Y	结束时间（这两个时间都不填默认查当天话单数据）	
caller	Y	主叫号码	
callee	Y	被叫号码	
timeLengthMin	Y	最小通话时间	
timeLengthMax	Y	最大通话时间	这两个是通话时长过滤条件

#### 4. 响应格式

```

{
  "result": {
    "error": 0,
    "msg": "成功"
  },
  "data": {
    "seq": 123,
    "response": {
      "seq": "123",
      "userData": "123",
    }
  }
}

```

```
"result": {
  "error": 0
},
"total": "4",
"cdr": [
  {
    "key": "6",
    "agent": "8000",
    "serviceType": "5",
    "answerTime": "2018-09-10 14:39:20",
    "caller": "15612343021",
    "callee": "13770761337",
    "timeLength": "7",
    "fee": 0.06,
    "releaseCause": "200"
  },
  {
    "key": "4",
    "agent": "8000",
    "serviceType": "5",
    "answerTime": "2018-09-10 14:36:47",
    "caller": "15612343021",
    "callee": "13813902888",
    "timeLength": "2",
    "fee": 0.06,
    "releaseCause": "200"
  },
  .....
]
}
```



```

    },
    "logs": null
}

```

## 5. 响应说明

response 域说明:

字段名	必填	说明	值说明
seq	Y	标识	
userData	N	回传字段	
Result	Y	错误码以及相关描述	
total	Y	话单条数	
cdr	Y	话单数据	

cdr 域说明:

字段名	必填	说明	值说明
key	Y	序列号	话单 ID
agent	Y	坐席工号	
serviceType	Y	业务类型	枚举值同上表
answerTime	Y	呼叫接通时间	
caller	Y	主叫号码	
callee	Y	被叫号码	
timeLength	Y	通话时长	
fee	Y	费用	
releaseCause	Y	挂断原因	标准 SIP 错误码
result	Y	呼叫结果	1 接通 2 未接通 3 空号
filename	Y	录音文件	

## 25. 推送任务状态

### 1. 描述

当任务状态发生变化时，系统会向用户推送状态和其他相关信息

### 2. URL

由用户提供 url

### 3. 消息格式

```
{
  "authentication": {
    "customer": "",
    "digest": "",
    "seq": "",
    "timestamp": ""
  },
  "notify": {
    "type": "taskinfo",
    "taskID": "",
    "taskName": "",
    "calleeAmount": "",
    "remainAmount": "",
    "status": "",
    "seq": "",
    "userData": ""
  },
  "param": {
    "debug": "",
    "lang": ""
  }
}
```

## 4. 消息说明

Notify 域说明:

字段名	必填	说明	值说明
seq	Y	命令序列号	
userData	N	用户回传数据	
type	Y	Api 类型, taskinfo	
taskID	Y	任务的 ID	
taskName	Y	任务名称	
calleeAmount	Y	任务号码总数	
remainAmount	Y	任务剩余号数	
status	Y	任务状态	0 进行中, 1 暂停, 2 结束
purpose	Y	对象类型	0 表示给班组的任务 1 表示给坐席的任务
group	Y	班组	对应 purpose 为 0
staffNo	Y	坐席工号	对应 purpose 为 1

## 5. 响应格式

```
{  
  "result": { "error": "", "msg": "" }  
}
```

## 6. 响应说明

Result 域说明:

字段名	必填	说明	值说明
error	Y	错误码	Int
msg	N	错误描述	