



# FCM5600 IPPPBX

## **API Documentations**

## Table of Contents

T API Doc.....	1
API Access Instructions .....	4
Parameters of the requests .....	4
Responses of the requests .....	4
Authentication .....	5
Login and get system information.....	5
Admin account login.....	5
Current system time.....	6
IP Extension Management .....	6
Query .....	6
Query the extension info.....	9
Add extension .....	9
Modify.....	11
Delete.....	11
System Status.....	12
System info and status .....	12
CPU info .....	12
Memory info .....	13
Storage.....	15
Current call.....	15
Extension register status .....	16
Trunk info .....	17
Analog trunk.....	17
Digital trunk .....	18
Iax trunk.....	18
Sip trunk.....	18
Trunk status.....	19
Call Forward.....	20
Query .....	20
Add call forward.....	21
Modify.....	22
Delete.....	22
Wakeup calls .....	23
Query .....	23
Add.....	24
Modify.....	24
Delete.....	25
Call logs query and call recordings download .....	25
call logs query .....	25
Call recording query .....	27
Conference recordings query .....	29
One touch recordings query.....	29
Download recordings .....	30
MOH management .....	30

Obtain the MOH audio file list.....	30
Dial Permissions .....	31
Query basic Dial Permission info .....	31
Phonebook.....	32
Query .....	32
Add.....	33
Modify.....	33
Delete.....	34
Delete All.....	34
Synchronization.....	34
Time Conditions .....	35
Query the time condition list .....	35
Make and terminate calls.....	35
Dial .....	35
Terminate calls .....	36

## API Access Instructions

All interactions use https, the certificate will be “Not secure”, please ignore it  
All interfaces use default prefix https://<ip>[:port]/, for example /system-info/model-info, The actual request should be in https://<ip>[:port]/system-info/model-infoformat, for brevity of explanation, the prefix had been omitted

All requests and responses use UTF-8 encoding

Changing Web Access port from the Web UI will cause the API port change, as they share the same port, by default it's 443.

### Parameters of the requests

If not specified, all requests and responses are given in json format, Content-Type: application/json

The parameter submission of the requests, if not specified, should abide the following pattern:

- All GET request should not use parameters
- All POST requests should use the http body field to send text message in raw format
- All upload request should use POST, encoding in binary format, you should post the file with parameter name file
- All download request should use GET, and in the URL passing parameters use query string, for example  
/download?name=test&t=1664545484564

For the paged responses, explanations had been given in the IP extension query section, the same contents in other sections will not be explained again

### Responses of the requests

A correct status code for the http request is 200, the validity of the results should look into the actual responses

In all API instructions, the explanations are all about the correct responses (status code 200)

For the responses with errors, https status code and explanations are as below:

- 400, parameter error, invalid format/record does not exist
- 401, permission error, not logged in or the login does not



```
"-----token
}
```

The status code and token will be returned if the login is successful. If the login fails, the status code will be displayed as 400, and the result will be displayed as not equal. The token is empty.

When you do not log out actively, the login period is one month. When you log out and access the interface that needs to be logged in, a 401 error will occur.

## Current system time

Fetch the real-time system time

URL:/pbx/systeminfo/system-current-time

Method:GET

Response:

```
{
"code":0,-----Status Code
"message":"OK",-----Results of program execution
"data":"2022-11-12 16:50"----- token
}
```

The responded time in yyyy-MM-dd HH:mm format

## IP Extension Management

IP extensions including SIP and IAX extensions, SIP is default and IAX is optional.

### Query

URL:/pbx/extension-digital/search-extension-page Method:POST

Param:

```
{
"columnKey":"string",-----Column key, index database field
"keyWord":"string",-----
Searchusingkeywordsofextensionnumber,extensionname,departmentname
(fuzzysearchsupported),useemptystringtosearchall.
"order":"exten"-----
SortbyanyIPextensionattributeoftherespondedattributesbelow.ASC:ascendin
gsort;DESC:descendingsort.
"page":0,-----Currentextensionlistpage,startsfrom0.
"pageSize":0,-----
Extensioncountoneachpage,minimalcountis1.
```

```
}

```

Response:

```
{
"code": 0,-----Status Code
"message": "OK",-----Results of program execution
"data": {
"items":[-----
Themainsearchresultsdisplayedinanarray,ifnosearchresultsanemptyarraywil lberesponded.
{
"exten":"100",-----Extensionnumber.
"enable":true,-----Whether to enable. The extension number can only be
used after it is enabled,true/false
"username":"测试 1",-----Extensionusernamealsothecallname.
"password":"123456",-----Extensionregistrationpassword.
"email":"","-----UserEmailaddress.
"outboundCid1":"18583230080",-----OutboundCID1.
"outboundCid2":"18583230081",-----OutboundCID 2.
"mobileNumber":"","-----Usermobilephonenumber. "dialPlanId":1,
-----Dofthedialpermissions,startingfrom1.
"language":"en",-----Systemvoicepromptslanguage.
"musicOnHoldId":2,-----MusiconholdfolderID.
"voicemail":true,-----Voicemailbox,enablediftrue,disablediffalse.
"voicemailPassword":"1234",-----Voicemailboxpassword.
"remoteExten":false,-----
Remoteextensionfeature,enablediftrue,disablediffalse.Ifenabledpleaseens
ureitusesstrongpassword.
"registCount":3,-----
Maximumlyallowedsimultaneousregisters. "videoCall":true,-----
----- Videocallfeature,enablediftrue,disablediffalse.
"videoCallCodec":"h264",-----
Supportedvideocodecs,vp8,h264,h263,h261,h263p,singlecodecrequired.
"webRtc":true,-----
Extensionwebportal,enablediftrue,disablediffalse.
"app":true,-----App account,enablediftrue,disablediffalse.
"webLogin":true,----- Web
login,enablediftrue,disablediffalse.
"dnd":true,-----dnd,enablediftrue,disablediffalse.
"recording":"none",-----
Callrecordingonthisextension,donotrecordifnone,recordinboundcallsifin,r
ecordoutboundcallsifout,recordallcallsifboth.
"callSpy":false,----- Callspyfeature,enablediftrue,disablediffalse.
"addressExpiration":1800,-----Extensionregisterexpirationinseconds.
"transport":"UDP",-----
SIPsingalingprotocol,UDP/TCP/TLS,singleprotocolrequired.
"dtmf":"rfc4733",-----DTMFmode,Auto/RFC4733/Inband/Sip
info,singleDTMFmoderequired.
"srtplib":false,-----SecureReal-
timeTransportProtocol,enablediftrue,disablediffalse.
"qualify":300,-----

```

```

Qualify, sends SIPOPTIONS regularly in secondsto check whether the extension is still online, disabled if 0.
"qualifyTimeout":30,----- Qualify timeout, in seconds.
"nat":true,-----
NAT support, enabled if true, disabled if false.
"iax":false,-----
IAX protocols support, enabled if true, disabled if false. "audioCodec":"alaw,ulaw,g729",-----
Supported audio codecs, ulaw, alaw, g729, gsm, g722, g726, speex, opus, can be multiple codecs, separated by comma.
"permit":"","-----
permit the host/network to register, default is empty, there are instructions below w.
"sendPai":true,----- SendPAI, true/false.
"sendRpid":true,----- SendRPID, true/false.
"rtptimeout":60,----- RTP timeout.
"groupNumber":"1",----- Substitute group, 1-64, one or more use separation, such as '1,42'.
"inbandProgress":false,----- Voice stream sends ring tone, enabled if true, disabled if false.
"whiteListTagId":0,----- whitelist, true/false
"dialPlanName":"DialPlan1",-----
Name of the dial permission used by the extension.
"departmentName":"d1,d2"-----
Department name the extension belongs to, multiple department names separated by comma.
}
],
"count":2,----- Total number of data
"page":1----- Current page number
"pageSize":20-----
Maximum number of extensions allowed all page
}
}

```

The responded results will be provided with pagination, each query should specify the page number and the extension count of the page

The responded properties and explanations are as above, they will not be introduced hereinafter

The permit IP, empty string by default, no limitation of SIP register

- If a single IP address, only register from this IP will be accepted, registers from any other IP will be refused
- Network (e.g. 192.168.2.0/24), only register from this network will be accepted, registers from any other network will be refused
- 192.168.2.0/24,!192.168.2.34, accept 192.168.2.0/24 network except 192.168.2.34.

## Query the extension info

All extensions will be responded, and only include basic extension info

URL:/pbx/extension-digital/extension-info Method:GET

Response:

```
{
  "code": 0,-----Status Code
  "message": "OK",-----Results of program execution
  "data":[
    {
      "exten":"126",-----Extension number
      "name":"126",-----User name (caller name)
      "type":"voip"-----Extension type: voip: IP extension, analog: analogue
      extension. Here you can only query IP extensions, analog extensions will be queried by
      another interface.
    }
  ]
}
```

The response will be displayed in an array, if no extensions found, an empty array will be responded.

## Add extension

Adding will first use the server to generate an available extension number

URL:/pbx/extension-common/gen-add-extension Method:POST

Param:

name=extension

Response:

```
{
  "code": 0,-----Status Code
  "message": "OK",-----Results of program execution
  "data":"101"-----Response data
}
```

the parameter is submitted as a form, like x-www-form-urlencoded, the response is an extension number

The parameter is not a variable, should be fixed value

The extension to be added will be checked by the server for validation.

URL:/pbx/extension-digital/is-extension-available/<exten> Method:GET

Response:

```
{
"code": 0,-----Status Code
"message": "OK",-----Results of program execution
"data":true-----Response data
}
```

Send the desired extension number to the server by the above URL, response will be given as true/false

Interface of adding extensions: URL:/pbx/extension-digital/add-extension Method:POST

Param:

```
{
"addressExpiration":1800,
"audioCodec":"alaw,ulaw,g729", "callSpy":false,
"dialPlanId":1, "dnd":false,
"dtmf":"rfc4733",
"email":"","enable":true,
"exten":"102", "iax":false,
"language":"en",
"mobileNumber":"","
"musicOnHoldId":1,
"nat":true,
"outboundCid1":"","
"outboundCid2":"","
"password":"joHpm#7AN7",
"permit":"","qualify":300,
"qualifyTimeout":30,
"recording":"none",
"registCount":1,
"remoteExten":false,
"rtpTimeout":60, "sendPai":true,
"sendRpid":true, "srtp":false,
"transport":"UDP",
"username":"102", "videoCall":false,
"videoCallCodec":"h264", "voicemail":true,
"voicemailPassword":"1234",
"webRtc":false
}
```

Response:

```
{
"code": 0,-----Status Code
"message": "OK",-----Results of program execution
"data":true-----Response data
}
```

The properties of the added extension please refer to the extension query section

The dial permission ID, MOH folder ID and extension validation

will

be checked, if the extension cannot be added, there will be error

## Modify

Extension number cannot be modified

URL:/pbx/extension-digital/update-extension Method:POST

Param:

(Same as add) Response:

(Same as add)

Extension number is used for confirmation of which extension to be modified, if the extension number doesn't exist, there will be error

## Delete

Multiple extensions can be deleted at one time

URL:/pbx/extension-digital/delete-extension Method:DELETE

Param:

```
{  
  "extensions":[-----The array of the extensions to be deleted  
  "string"  
  ]  
}
```

Response:

```
{  
  "code": 0,-----Status Code  
  "message": "OK",-----Results of program execution  
  "data":1-----Delete data quantity  
}
```

Responded the deleted number of records

## System Status

### System info and status

Device model, software&hardware version, system up time, network, etc.

URL:/pbx/systeminfo/version Method:GET

Response:

```
{
"code": 0,-----Status Code
"message": "OK",-----Results of program execution
"data":{
"1model":"FIBERME-FCM5600",-----
Devicemodel
"2DeviceName":"System",-----Device name
"3WAN-IP":"192.168.11.150",-----WAN port IPaddress
"4WAN-Mac":"00:E0:70:2F:47:A7",-----WAN port
MAC address
"5LAN-IP":"192.168.10.100",-----LAN port IP
address
"6LAN-Mac":"00:E0:70:33:24:A3",-----LAN port MAX
address
"7hardwareVersion":"V4",-----Hardware version
"8version":"v4.0.0",-----Current
software version
"9Uptime":"2022-11-23 16:21 / 13 days, 23:43",-----
-----System uptime in seconds
}
}
```

### CPU info

The results need to be calculated to get an actual utilization ratio .

Formula (cpu usage information/100) \* 100%.

URL:/pbx/systeminfo/cpu

Method:GET

Response:

```
{
"code": 0,-----Status Code
"message": "OK",-----Results of program execution
"data":[26.666666665373164,27.835051547613567,17.85714285703679,24.8730
96445841846]-----CPU usage information (ratio of
usage to total number * 100) CPU
}
}
```

The data returned here is the ratio of CPU usage to the total number.

## Memory info

URL:/pbx/systeminfo/mem

The data obtained mainly uses the total memory of physical memory, and other data such as used memory and available memory have no practical significance.

Method:GET Response:

```

{"code":0, -----Status Code
"message":"OK",-----Results of program execution
"data":{
"swap":{ -----Exchange area
"total":500166656, -----Total exchange area
"used":1572864, -----Total swap area used
"free":498593792, -----Total number of free swap areas
"usedPercent":0.31446798404730125, -----Usage ratio
(total amount of exchange zones used/total amount of           exchange
zones)
"sin":1867776,
"sout":4386816,
"pgIn":0,
"pgOut":0,
"pgFault":0,
"pgMajFault":0
},
"virtual":{ -----Physical memory
"total":4017598464, -----Total memory
"available":2547400704, -----Available memory
"used":1192071168, -----Memory used

```

```

"usedPercent":29.671237150293774, -----Memory
usage ratio (used memory/total memory * 100)
"free":1970675712,-----Free memory
"active":1252728832, -----Buffer size in active
use
"inactive":620261376, -----Buffer size in infrequent use
"wired":0,
"laundry":0,
"buffers":46747648,
"cached":808103936,
"writeBack":0,
"dirty":20480,
"writeBackTmp":0,
"shared":41365504,
"slab":99045376,
"sreclaimable":55652352,
"sunreclaim":43393024,
"pageTables":10772480,
"swapCached":581632,
"commitLimit":2508963840,
"committedAS":3012792320,
"highTotal":0, "highFree":0,
"lowTotal":0,
"lowFree":0, "swapTotal":500166656,
"swapFree":498593792,
"mapped":268148736,
"vmallocTotal":35184372087808,
"vmallocUsed":0,
"vmallocChunk":0,
"hugePagesTotal":0,
"hugePagesFree":0,
"hugePageSize":2097152}

```

}

}

Storage resources is calculated by KB.

## Storage

According to different device model, the storage device will be different.

URL:/pbx/systeminfo/disk

Method:GET

Response:

```
{
"code":0,-----Status Code
"message":"OK",-----Results of program execution
"data":{
"rcdtotal":"0",-----Total recording storage
"rcdused":"","-----Recording storage used
"systotal":"5403",-----
Totalmainstorage(SDorEMMC)resources
"sysused":"2354",-----Usedmainstorageresources
"usbtotall":"0",-----TotalUSBstorageresources
"usbused":"0"-----UsedUSBstorageresources
}
}
```

Storage resources is calculated by MB.

## Current call

Get the current number of calls.

URL:/pbx/systeminfo/calls Method:GET

Response:

```
{"code":0,-----Status Code
"message":"OK",-----Results of program execution
"data":[
"calls":"","-----Calls
"maxcalls":"","-----Max calls
]
```

}

## Extension register status

URL:/pbx/systeminfo/extension-status Method:GET

Response:

```
{
"code":0,-----Status Code
"message":"OK",-----Results of program execution
"data": [
{
"exten":"100",-----Extensionnumber
"name":"100",-----Extension display name when dialing
"content":{
"newMsg":"","-----Voice message
"dnd":"","-----Don't disturb
"wakeup":null----- Wakeup
},
"roundtripUsec":"","-----Registration timeout
"addr":"","-----Address
"uri":"","-----URL
"statusText":"Unavailable"-----Status
}
]
}
```

Responses are given by each extension number

An IP extension can be registered on multiple endpoints, the responded results will be given by multiple individual objects

Possible status:

- Idle
- InUse
- Hold
- Ringing
- Unknown
- Unavailable

It will respond with all extension of the system, if there's no extension it will respond with empty object.

## Trunk info

URL:/pbx/systeminfo/trunk-info

Method:GET

Response:

```

{"code":0,
"message":"OK", "data":{
"analogOut":"1:1:FXS::\n2:2:FXS::\n3:3:none::\n4:4:none::\n5:5:none
::\n6:6:none::\n7:7:none::\n8:8:none::", ----- Analog
trunk
"digitalOut":"","-----Digital trunk
"iaxOut":"","----- Iaxtrunk
"sipOut":"eeee:server::0.00:Unregistered"}} ----- Siptrunk

```

Each data in the relay string is separated by the/n separator. Use: to separate the data in each data.

## Analog trunk

```

"analogOut":"1:1:FXS::\n2:2:FXS::\n3:3:none::\n4:4:none::\n5:5:none
::\n6:6:none::\n7:7:none::\n8:8:none::"
"analogOut":"Serial number:Channel:Trunk type
(FXO/FXS/GSM/WCDMA) :Status:Channel status:BLF\n2:2:FXS::\n

```

Possible status:

- RED-----CO line disconnected
- OK-----CO line connected
- NO SIM CARD-----SIM card not inserted(GSM/WCDMA)
- SIM ERROR-----SIM card error(GSM/WCDMA)

The signal strength of GSM and WCDMA will be given by digits

- signal strength 0, level 0, no signal
- signal strength 1-9 level 1
- signal strength 10-15, level 2
- signal strength 16-20, level 3
- signal strength 21-25, level 4
- signal strength above 25, level 5

While signal level is 0, it includes NO SIM CARD and SIM ERROR

Possible channels status:

- InUse
- Idle

## Digital trunk

```
"digitalOut": ""
```

## Iax trunk

```
"iaxOut": "iaxto10:client:192.168.17.30:2:Registered"  
"iaxOut": "Name:Trunk type:Ip:Delayed:Status"
```

Possible status:

- Idle
- Registered
- Unregistered
- Rejected

Possible trunk type status:

- client
- server

## Sip trunk

```
"sipOut": "eeee:server::0.00:Unregistered"}}
```

```
"sipOut": "Name:Trunk type:Ip:Delayed:Status" Possible
```

values of status: Same as iax.

Possible values of trunk type: Same as iax.

## Trunk status

URL:/pbx/systeminfo/trunk-status Method:GET

Response:

```
{
  "code":0,-----Status Code
  "message":"OK",-----Results of program execution
  "data": [
    {
      "delay":"","-----Delayinmilli-
      seconds,ifQualifyisdisabledthevaluewillbe'na'
      "name": "adf",----- Trunk name
      "server": "",----- Theotherparty/server'sIP/domain
      "status": "",----- Status
      "trunkType": "iax", -----Trunk type
      "model": "server"-----Trunk model, client/server
    }
  ],
  {
    "delay": "",
    "name": "33",
    "server": "",
    "status": "",
    "trunkType": "sip",
    "model": "client"
  }
]
```

The response will be arrays, empty array will be given if not trunk exists

All trunks will be listed, some properties if the trunk is not in use will be blank

Possible status:

- Unregistered
- Rejected
- Unauthenticated
- Registered
- Authenticated

## Call Forward

### Query

URL: /pbx/call-forward/list Method: POST

Param:

page:1, pageSize:20,

The parameters only used for pagination and sorting

Response:

```
{
  "code":0,
  "message":"OK", "data":{
    "items":[
      {
        "id":1,----- RecordID
        "exten":"102",----- Extensionnumber
        "type":"busy",----- Callforwardcondition
        "dest":"103",----- Callforwarddestination
        "timeout":5,----- Ringtimeoutbeforeforwarding
        "enable":true----- Availabilityofthisrecord,yesiftrue,noiffalse
      }
    ],
    "count":1,
    "page":1, "pageSize":20
  }
}
```

Available forward conditions:

- busy-----Forward when busy
- noanswer-----Forward when no answer
- noan\_busy-----Forward when busy or no answer
- always-----Forward unconditionally

## Add call forward

The list of configurable extensions is displayed first.

URL:/pbx/call-forward/available Method:GET

Response:

```
{
  "code":0,
  "message":"OK", "data":[
    {
      "exten": "100", ----- Extensionnumber
      "name": "100", ----- Username(callername)
      "type": "voip"-----
      Extensiontype,analog:analogueextension,voip:IPextensions
    },
    {
      "exten": "118",
      "name": "118",
      "type": "voip"
    }
  ]
}
```

Results will be given in arrays, empty array will be given if no extension

URL:/pbx/call-forward/add

Method:POST

Param:

```
{ "dest": "123456789",
  "enable":0,
  "exten": "2345", "id":0,
  "timeout":10,
  "type": "busy"
}
```

Response :

```
{
  "code":0, "message":"OK",
  "data":true
}
```

## Modify

URL:/pbx/call-forward/update

Method:POST

Param:

```
{
  "id":4,-----RecordID
  "dest":"123456789",
  "enable":0,
  "exten":"2345",
  "timeout":0,
  "type":"busy"
}
```

Response:

The same as /pbx/call-forward/add response

In the param, record ID should be given, if the record ID doesn't exist, there will be error

## Delete

One or more records can be deleted at once.

URL:/pbx/call-forward/delete Method:DELETE

Param:

```
{
  "ids":[-----TherecordIDtobedeleted
  4
  ]
}
```

Response:

The same as /pbx/call-forward/add response

Delete all call forwarding URL:/pbx/call-

forward/delete-all Method:DELETE

Response:

```
{
  "code":0, "message":"OK",
  "data":{}
}
```

## Wakeup calls

The time for wakeup calls can only accurate to minute

The time of a wakeup call to be submitted should not be earlier than the current system time

### Query

URL:/pbx/wakeup-call/list

Method:POST

Param:

page:1, pageSize:20,

Response:

```
{
  "code": 0, "message":
  "OK", "data": [
    {
      "callDateTime": "2022-07-25 12:00", -----
      Timeofthewakeupcall,in"yyyy-MM-dd HH:mm"format
      "exten": "501", ----- Extensionnumber
      "musicID": 4, -----Voice file number
      "id": 8----- RecordID
    },
    {
      "callDateTime": "2022-07-25 12:00",
      "exten": "888",
      "musicID": 4,
      "id": 13
    }
  ]
}
```

The format of time is always the same no matter query, add, modify

## Add

URL:/pbx/wakeup-call/add

Method:POST

Param:

```
{
  "callDateTime":"2022-12-15 12:42",-----
  Timeofthewakeupcall,in"yyyy-MM-dd HH:mm"format
  "extens":[-----Theextension/
extensionstoaddwakeupcalls
  "110"
  ],
  "musicID":0,-----Voice file number
}
```

Response:

```
{
  "code":0, "message":"OK",
  "data":{}
}
```

## Modify

URL:/pbx/wakeup-call/update

Method:POST

Param:

```
{
  "callDateTime":"2022-12-15 12:42",-----Wakeuptime
  "ids":[-----TherecordIDstobemodified
  24
  ],
  "musicID":0,-----Voice file number
}
```

Response:

The same as add

## Delete

URL:/pbx/wakeup-call/delete

Method:DELETE

Param:

```
{
  "ids":[----- TherecordIDstobedeleted
    21
  ]
}
```

Response:

The same as add

## Call logs query and call recordings download

### call logs query

URL:/pbx/call-log/search-calllog-page Method:POST

Param:

```
{
  "from":"2022-01-01",----- Startdate,inyyyy-MM-
  ddformat,itwillfilterthelogsfrom0:00:00ofthestartdate
  "to":"2022-12-12",----- Enddate,inyyyy-MM-
  ddformat,itwillfilterthelogswithin23:59:59oftheenddate "type":"all",-----
  -----
  Calltype:all/incoming/outgoing/internal
  "src":"","-----
  Caller,donotsearchbycallernumberifleaveitblank
  "dest":"","-----
  Callee,donotsearchbycalleenumberifleaveitblank
  "pinCode":"","-----
  PINnumberfromaPINSet,onlyoutboundcallsdialedwithPINnumber,inthecalllogs therewillbePIN
  "trunkName":"","-----
  Trunkname,donotsearchbytrunknameifleaveitblank
  "status":"all",-----
  Thestatusofthecalls,willbedescribedlater.
  "page":1, "pageSize":20,
```

```
"order": "string",
"columnKey": "string",
}
}
```

The filter conditions, call type will use absolute matching, start date and end date will use absolute matching as well, other conditions will use approximate string matching

The order of the logs will be given by reversed order of the time when the call took place, so the latest call logs will be given at first place

Response:

```
{
"code": 0, "message":
"OK",
"data": {
"items": [
{
"id": 1, ----- RecordID
"callFrom": "100", ----- Callname
"callSrc": "100", ----- Callernumber
"callTo": "1100", -----
Calleenumber, thenumberthecallercalled
"destNum": "trunk-test11111", -----
Final callee, depends on if the call had been transfer/forward or not
"startTime": "2022-07-01 21:45:17", -----
Time when the call started, in "yyyy-MM-dd HH:mm:ss" format
"duration": 2, ----- Call duration
"callStatus": "ANSWERED", -----
Call status: ANSWERED/BUSY/FAILED/NOANSWER "callType": "outgoing", ---
-----
Call type: all/incoming/outgoing/internal
"pinCode": "", -----
PIN number from a PIN set, only outbound calls dialed with PIN number, in the call logs
there will be PIN, the calls which didn't use PIN, the value of this field will be null "trunkName": "", -----
-----
The trunk been used by this call, if not trunk had been used, the value of this field will be null
"recordingFile": "", ----- recording file
}
],
"page": 1
```

```
"pageSize":20,
"count":332,-----Number of call records
}
}
```

Call type:

- all-----All type of calls
- incoming-----inbound calls
- outgoing-----outbound calls
- internal-----Internal (extension to extension) calls

Call status:

- all-----“All” is used to search calls of all status.
- ANSWERED-----Successfully calls, answered calls
- BUSY-----Called party busy
- FAILED-----Call failed
- NO ANSWER-----Called party didn’t answer

For the logs which had call recording, you may query the recording info and download the recordings

### Call recording query

URL:/pbx/call-log/search-general-recording-page Method:POST

Param:

```
{
"from":"2022-11-01",
"to":"2022-11-12",
"type":"all",
"src":"","
"dest":"","
"pinCode":"","
"trunkName":"","
"status":"all", "page":1,
"pageSize":20,
"order":"string",
"columnKey":"string",
}
}
```

Parameter introduction is the same as query call log

Response:

```
{
  "code": 0, "message":
  "OK",
  "data": {
    "items": [
      { "id": 1,
        "callFrom": "conference", "callSrc": "100",
        "callTo": "1100",
        "destNum": "trunk-test", "startTime": "2022-
        11-01 22:45:00",
        "duration": 1,
        "callStatus": "ANSWERED",
        "callType": "outgoing",
        "pinCode": "",
        "trunkName": "", "FileName": "/var/spool/asterisk/monitor/recording/20221129/843/16697047
        79.411-843-842-20221129-145300.gsm", -----
The path and filename of the recording
        "timeLength": "1" ----- The duration of the recording
        "playUrl": "/pbx/play/general-recording/20221129/843/1669704779.411-843- 842-20221129-
        145300.gsm" ----- System
        request path for recording file
      }
    ],
    "page": 1, "pageSize": 20,
    "count": 332,
  }
}
```

The same parameters for call logs and recording query will not be explained again

In the recording file name the path is included, it can be used for downloading the recording

### Conference recordings query

URL:/pbx/call-log/search-conference-recording-page Method:POST

Param:

```
{
  "from":"2022-11-01",
  "to":"2022-11-12",
  "type":"all",
  "src":"",
  "dest":"",
  "status":"all", "page":1,
  "pageSize":20,
  "order":"string",
  "columnKey":"string",
}
```

The same parameters as the query call logs in the parameters are not explained separately. Compared with the parameters of conference recording and call recording, the parameters of relay and pin code are only missing.

Response:

The parameters are the same as the call recording.

### One touch recordings query

URL:/pbx/call-log/search-onetouch-recording-page Method:POST

Param:

The same as call recording

Response:

The same as call recording

## Download recordings

The requests of downloading the following 3 types of recordings are the same.

- Call recordings:/pbx/play/general-recording
- Conference recordings:/pbx/play/conference-recording
- One touch recordings:/pbx/play/onetouch-recording

Param:

playUrl?token

file name should come from the recording query responses, including the path of the recordings

Response should be http packet with status code 200 . Download by directly requesting file resources.

## MOH management

### Obtain the MOH audio file list

URL:/pbx/voice-file/list-music-on-hold-voice-file-folder Method:GET

Response:

```
{
  "code": 0, "message":
  "OK",
  "data": [
    {
      "id":1,-----AudiofolderID
      "name":"default",-----Audiofoldername
      "sort":"random"-----Playmodeoftheaudiofilesinthisfolder
    },
    {
      "id":2,
      "name":"tttt",
      "sort":"random"
    }
  ]
}
```

The responses are given by arrays, it will respond with an empty array if no result found.

## Dial Permissions

### Query basic Dial Permission info

URL:/pbx/dial-context/list-dial-plan-simple-info Method:GET

Response:

```
{
  "code": 0, "message":
  "OK", "data": [
    {
      "id": 1, ----- DialPermissionID
      "name": "DialPlan1", ----- DialPermissionname
      "featureCode": true, -----Feature Code,  enablediftrue,disablediffalse.
      "queue": true, -----Call queue,  enablediftrue,disablediffalse. "parkedCall": true, -----Parked
      call,  enablediftrue,disablediffalse. "pickupExten": true, -----Call pickup,
      enablediftrue,disablediffalse. "conference": true,-----conference call,
      enablediftrue,disablediffalse.
      "pagingIntercom": true,-----paging and Intercom,  enablediftrue,disablediffalse.
      "extension": true,----- Extension,  enablediftrue,disablediffalse.
      "ivr": true, -----Automated Attendant,  enablediftrue,disablediffalse.
      "disa": true, ----- DISA,  enablediftrue,disablediffalse. "spy": true, ----
      ---Call spy,  enablediftrue,disablediffalse.
      "console": false,-----呼叫队列,  enablediftrue,disablediffalse.
      "department": true, ----- Department,  enablediftrue,disablediffalse.
      "seizeAnalogLine": true----- Seize CO Line,
      enablediftrue,disablediffalse.
    }
  ]
}
```

The responses are given by arrays, it will respond with an empty array if no result found.

## Phonebook

### Query

URL:/pbx/phone-book/list

Method:GET

Param:

```
{
  "keyword":"string",-----
  Keywordquery,useemptystringtoqueryallcontacts "order":"string",
  "columnKey":"string",
  "page":0, "pageSize":0
}
```

Fuzzy matching the contact name, phone number, email address and company name by the given keywords

Response:

```
{
  "code":0, "message":"OK",
  "data":{
    "items"[
      {
        "id":14,----- RecordID
        "name":"22",----- Contactname
        "phoneNumber":"333444",----- Phonenumber
        "email":"","----- Emailaddress
        "company":"","----- Companyname
        "addByRole":"","----- Roleofthecreatorofthiscontact
        "addUser":"842",-----
        Extensionnumberofthecontactcreator,onlyiftheroleisextensionthisfieldisvalid
        "dialCode":"","----- SpeedDialNumber
      },
    ],
    "count":1,
    "page":1, "pageSize":20
  }
}
```

Possible contact creators:

- admin-----Admin user
- operator-----Operator user
- extension-----Extension user

Only if the contact creator is an extension user then there will be value in the addUser

field, and it would be the extension number.

## Add

URL:/pbx/phone-book/add

Method:POST

Param:

```
{
  "addByRole":"","
  "addUser":""," "id":0,
  "phoneNumber":"","
  "name":"jj",
  "company":"","
  "email":"","-----Emailaddress
  "dialCode":""
}
```

If the emailfield is not empty, it should comply with the standard Email format.Parameter comments have been described in the above query.

Response:

```
{
  "code":0, "message":"OK",
  "data":true
}
```

Results of the action.

## Modify

URL:/pbx/phone-book/update

Method:POST

Param:

Parameters same as add contacts

Response:

```
{
  "code":0, "message":"OK",
  "data":{}
}
```

Modify contact info will not change the “creator” of the contact.

## Delete

Multiple records can be deleted at once

URL:/pbx/phone-book/delete-contact

Method:DELETE

Param:

```
{
  "ids":[-----ArrayoftherecordID
  3
  ]
}
```

Response:

```
{
  "code":0, "message":"OK",
  "data":true
}
```

## Delete All

URL:/pbx/phone-book/delete-all Method:DELETE

Response:

```
{
  "code":0, "message":"OK",
  "data":{}
}
```

## Synchronization

Sync phonebook contacts to LDAP server

URL:/pbx/phone-book-ldap/sync

Method:GET

Response:

The response result is the same as the delete response.

## Time Conditions

### Query the time condition list

URL:/pbx/time-rule/list-time-rule-simple-info Method:GET

Response:

```
{
  "code": 0,
  "message": "OK", "data": [
    {
      "id": 1, ----- TimeconditionID
      "name": "上班" ----- Timeconitionname
    },
    {
      "id": 3,
      "name": "dghdfgh"
    }
  ]
}
```

The responses are given by arrays, it will respond with an empty array if no result found.

### Make and terminate calls

By using this API you may use a specific phone to dial a specific number or terminate an ongoing call.

#### Dial

Specify the caller number and the callee number to originate a call, the callee number should be given with correct dial prefix.

URL:/api/call-dial/dial

Method:POST

Param:

```
{
  "dest": "111", ----- Calleenumbr
  "src": "123" ----- Callernumber
}
```

Response:

```
{
  "code": 0, ----- Status Code
  "message": "OK", ----- Results of program execution
  "data": {
```

```
"msg":"ami dial"-----Dial success
}
}
```

The response is only about the dial command, for the call status please check call logs.

If the caller number does not exist, or the callee number is invalid, you'll get '400' error code.

## Terminate calls

URL:/api/call-dial/hangup

Method:POST

Param:

```
{
"extensions":["string"],----- thenumbertobeterminatedofaphonecall.
}
```

Response:

```
{
"code":0,-----Status Code
"message":"OK",-----Results of program execution
"data":{
"msg":"ami hangup"-----Hangup success
}
}
```

The response is only the results of the terminate command, it does not indicate the result of the call had been terminated or not.

If the extension does not exist, you'll receive '400' error code. The execution result shows the reason for the failure.