API INTEGRAÇÃO

Versão 5.0.3.1

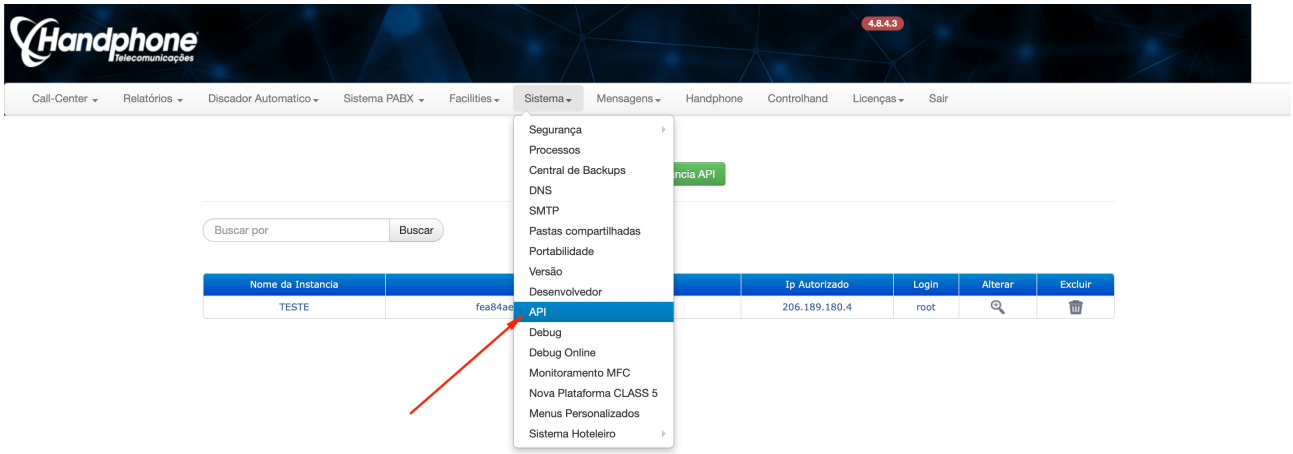
API: v1.2

SUMÁRIO

Configurações Iniciais e Ativação da API

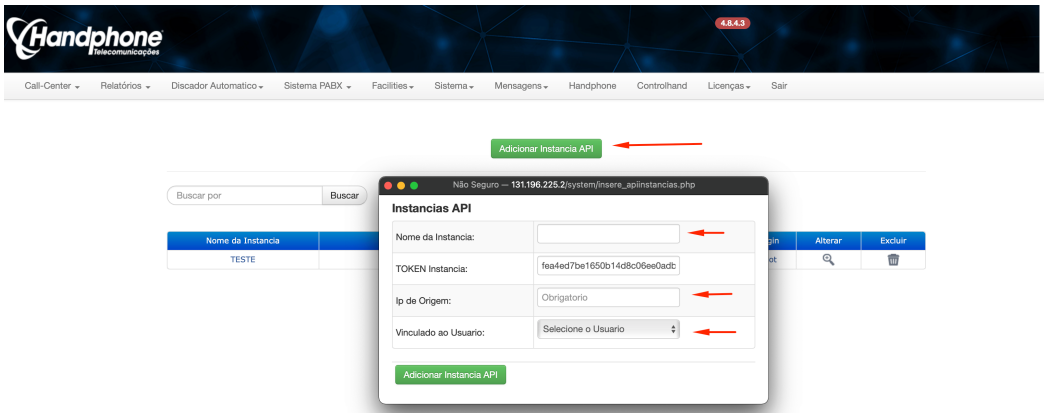
As APIS são ativadas por instancias e possuem licenciamento próprio, sendo assim é necessário possuir licença para ativação de uma nova instancia API.
Essa instância deverá ser vinculada a um usuário da central e cadastrado apenas a **01 ÚNICO** ip de origem, caso necessite de outro IP de origem será necessário criar outra instancia.

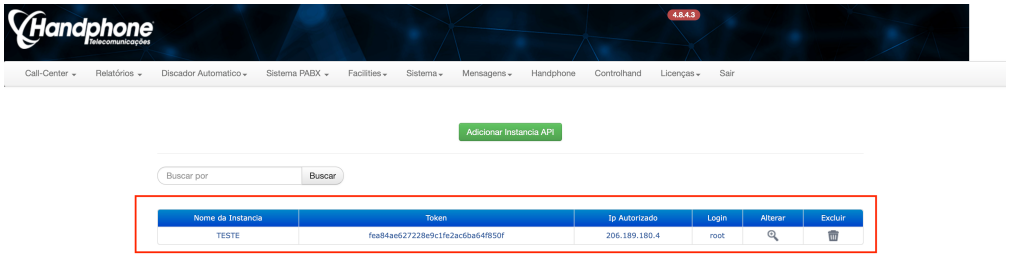
1) Acesse o menu Sistema -> API



2) Insira uma Nova Instancia de API

O próprio sistema ira gerar uma HASH com criptografia que será utilizada posteriormente.





Utilizando API - Login, Senha, Token, Ip Origem

Essa API por questões de segurança possui autenticação de 4 níveis.

Nesse caso quaisquer acesso a API serão sempre validados os 04 campos

- Login
- Senha
- Token (hash criada no momento da liberação da API)
- IP de Origem

Se um desses parâmetros estiverem errados o acesso não será concedido, como essa API fornece informações confidenciais como Gravações de Chamadas, informações de detalhadas de cada registro de ligação, comandos para efetuar chamadas, capturar informações de mailings etc, o acesso não será concedido pelo sistema se um dos campos não estiverem corretos.

Somente 01 único IP pode ser cadastrado como origem para cada token, se for necessário cadastrar mais de 01 IP o Administrador deverá fornecer outro token.

Acessando API - https

{“acao":"XXXX","login":"XXXXXX","senha":"XXXXXXXX","token":"XXXX"}

As informações acima serão sempre validadas em quaisquer requisição, conferindo sempre a origem do IP que está solicitando ou executando os comandos na *API.*

A API sempre deverá ser acessada com *https* caso contrario irá retornar o seguinte erro:

["Status","Erro de Protocolo"]

Registro de Ramais - Método TCP/UDP

Registrando dispositivos SIP

É possível registrar quaisquer dispositivo com tecnologia SIP 2.0.

Protocolos Aceitos: TCP ou UDP

Porta de Registro: 7048

Endereço de Registro: https://Endereco_da_central

Codecs disponíveis: G729.A, G711a, G711u, GSM

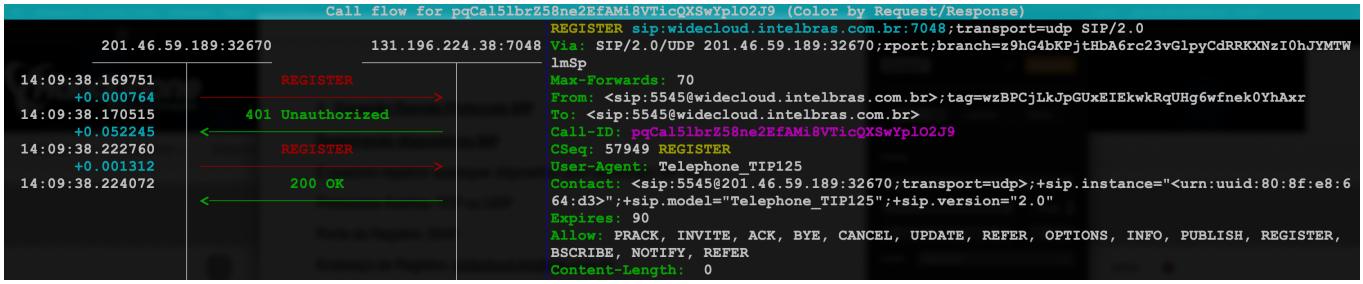
Faixa de Portas RTP: 10000 - 20000

Deverá ser configurado no dispositivo SIP client o login (número do ramal) e senha (senha de autenticação do ramal configurada previamente na nuvem) para efetuar o registro.

O dispositivo sip client deverá enviar um pacote do tipo ***REGISTER*** com protocolo TCP ou UDP, no primeiro envio ele ira receber uma resposta SIP 401, e após a resposta o dispositivo deverá enviar novamente o mesmo pacote para receber a resposta 200 OK da nuvem.

OBS: Entende-se por dispositivo sip client quaisquer equipamento ou software que possua tecnologia SIP 2.0

Exemplo:

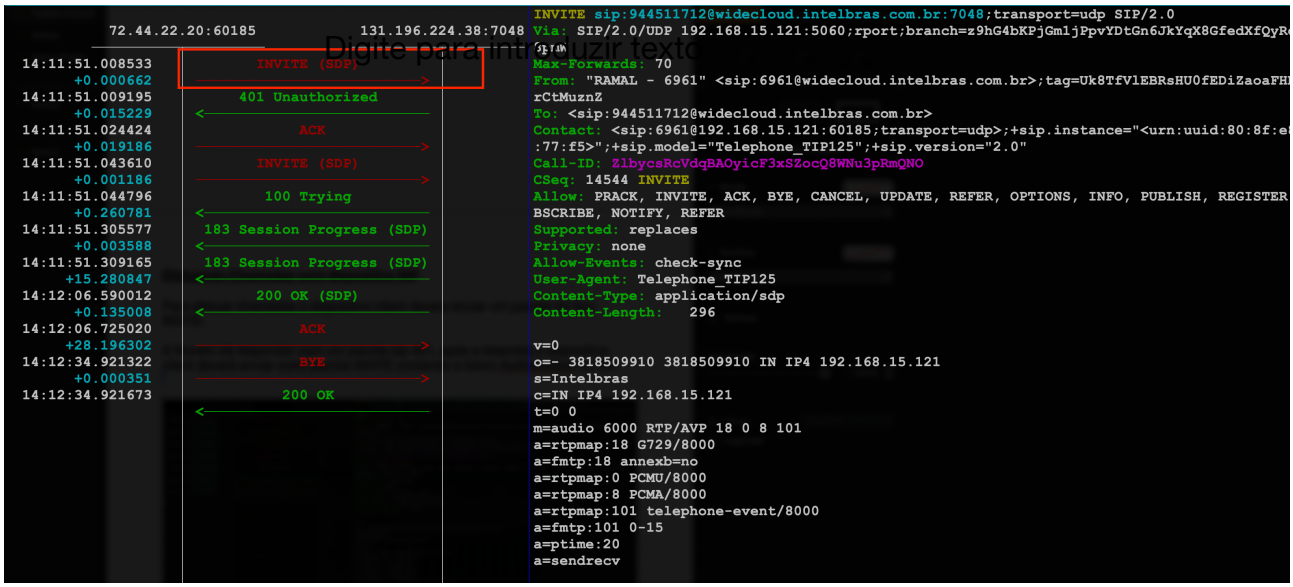


No momento do envio do pacote o dispositivo client envia uma porta de origem, essa porta normalmente é escolhida de forma aleatória pelo próprio dispositivo client. Se outro dispositivo client enviar a mesma porta de origem haverá um conflito e o dispositivo não ira receber chamadas corretamente salvo se o ip de internet for diferente.

Efetuando Chamadas com dispositivo SIP

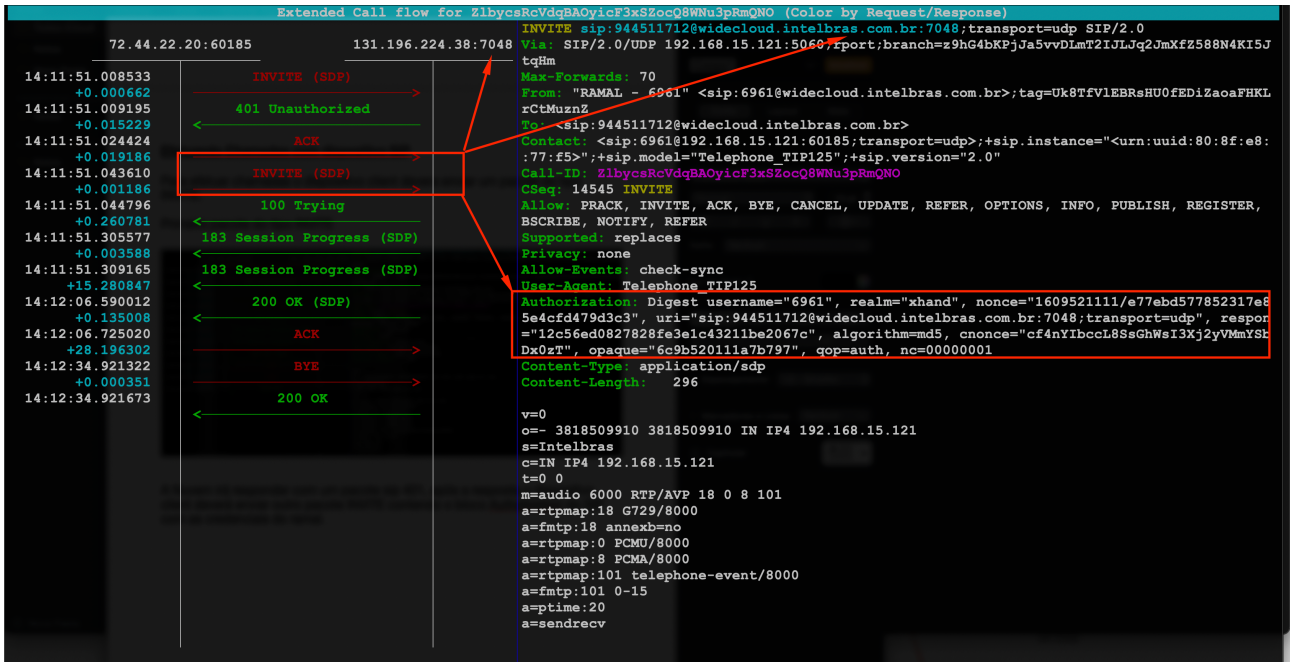
Para efetuar chamadas o dispositivo client devera enviar um pacote do tipo ***INVITE*** no endereço da nuvem e na mesma porta que foi enviado o registro. Esse pacote é um pacote padrão SIP 2.0 com os campos From, To, Conta, Call-Id, CSeq, Allow, Supported, User-Agent

Primeiro pacote do tipo INVITE



A nuvem irá responder com um pacote sip 401, após a resposta o dispositivo client deverá enviar outro pacote ***INVITE*** contendo o bloco Authorization do SIP com as credenciais do ramal.

Segundo pacote SIP contendo as credenciais



Nesse momento a nuvem irá aceitar a chamada e processar de acordo com as regras de roteamento.

Registro de Endpoints WebRTC

É possível utilizar o método WebRTC com TLS e SSL para registro de dispositivos SIP client.

O Registro deverá ser feito na porta 8089 sobre a camada de conexão *https* (443) Ou conexão web socket :”ws"

O endereço de registro e quaisquer outra transação será o mesmo utilizado no método de registro de ramais UDP/TCP

Antes de fazer o registro é necessário ativar a função ramal-web nas configurações do ramal.

Como se trata de uma conexão no formato web socket toda a transação SIP será criptografada entre o dispositivo ou software até a nuvem.

Abaixo exemplo log do software utilizando método https e wss

```
Have WebRTC = yes
Have GUM = yes
Engine initialized
s_websocket_server_url=(null)
s_sip_outboundproxy_url=(null)
b_rtcweb_breaker_enabled=yes
b_click2call_enabled=no
b_early_ims=yes
b_enable_media_stream_cache=yes
o_bandwidth={}
o_video_size={}
SIP stack start: proxy='widecloud.intelbras.com.br:8089/ws', realm='<sip:widecloud.intelbras.com.br>', impi='7777', impu='"7777"
<sip:7777@widecloud.intelbras.com.br>'
Connecting to 'wss://widecloud.intelbras.com.br:8089/ws'
__tsip_transport_ws_onopen
==stack event = starting
==stack event = started
State machine: tsip_dialog_register_Started_2_InProgress_X_oRegister
SEND: REGISTER sip:widecloud.intelbras.com.br SIP/2.0
Via: SIP/2.0/WSS df7jal23ls0d.invalid;branch=z9hG4bKSvVpU7eMPfo5cGy3ZkB1Dry68HilZ9Tx;rport
From: "7777"<sip:7777@widecloud.intelbras.com.br>;tag=rzwvccxKfbnJ8AqDdoxi
To: "7777"<sip:7777@widecloud.intelbras.com.br>
Contact: "7777"<sips:7777@df7jal23ls0d.invalid;rtcweb-breaker=yes;transport=wss>;expires=50;click2call=no;+g.oma.sip-im;+audio;language="en,f
Call-ID: 4d0cf4f1-c9d8-3ab7-aa66-36fb55895776
CSeq: 5729 REGISTER
```

Exemplos de bibliotecas:

https://sipjs.com

https://www.doubango.org/sipml5/

Discador - Inserindo contatos em Campanhas (preditivo ou preview)

Com essa função é possível inserir contatos em uma campanha pré configurada na nuvem. Essa facilidade possibilita realizar integrações para que os mailings do discadores sejam alimentados de forma automática.

O método consiste em realizar um *POST* no formato *JSON* com as seguintes informações

acao; -> INSERT,STATUS (Insert para inserir contatos, status para resultado dos contatos)
Idcampanha; -> Deverá ser preenchido com o valor do ID da campanha criada na nuvem.
idcrm; -> Utilizado para Integrações de sincronismo de tela
cliente; -> Nome do cliente (contato)
cpf; -> CPF do contato
ddd1; -> DDD do contato com dois dígitos
tel1; -> Telefone do Contato
ddd2; -> DDD do contato com dois dígitos
tel2; -> Telefone do Contato
ddd3; -> DDD do contato com dois dígitos
tel3; -> Telefone do Contato
ddd4; -> DDD do contato com dois dígitos
tel4; -> Telefone do Contato
ddd5; -> DDD do contato com dois dígitos
tel5; -> Telefone do Contato
endereco; -> Campo Aberto 254 Caracteres
bairro; -> Campo Aberto 254 Caracteres
cidade; -> Campo Aberto 254 Caracteres
estado; -> Campo Aberto 254 Caracteres
cep; -> Campo Aberto 254 Caracteres
contato; -> Campo Aberto 254 Caracteres
agente; -> Campo Aberto 254 Caracteres
dpto; -> Campo Aberto 254 Caracteres
conta; -> Campo Aberto 254 Caracteres
gen1; -> Campo Aberto 254 Caracteres
gen2; -> Campo Aberto 254 Caracteres
gen3; -> Campo Aberto 254 Caracteres
gen4; -> Campo Aberto 254 Caracteres
gen5; -> Campo Aberto 254 Caracteres
gen6; -> Campo Aberto 254 Caracteres
gen7; -> Campo Aberto 254 Caracteres
gen8; -> Campo Aberto 254 Caracteres
gen9; -> Campo Aberto 254 Caracteres
gen10; -> Campo Aberto 254 Caracteres
gen11; -> Campo Aberto 254 Caracteres
gen12; -> Campo Aberto 254 Caracteres
gen13; -> Campo Aberto 254 Caracteres
gen14; -> Campo Aberto 254 Caracteres
gen15; -> Campo Aberto 254 Caracteres
gen16; -> Campo Aberto 254 Caracteres

gen17; -> Campo Aberto 254 Caracteres
gen18; -> Campo Aberto 254 Caracteres
gen19; -> Campo Aberto 254 Caracteres
gen20; -> Campo Aberto 254 Caracteres
gen21; -> Campo Aberto 254 Caracteres
gen22; -> Campo Aberto 254 Caracteres

Cada contato poderá ter até 5 telefones, o campo DDD deverá ser sempre separado do campo numero de telefone.

Exemplo:

{“acao”:"insert",idcampanha":"235","login":"XXXXXX","senha":"XXXXXXXX","token":"XXXX","idcrm":"1234","cliente":"Leandro","cpf":"346150521852","ddd1":"11","tel1":"41187777","ddd2":"21","tel2":"12345678","gen15":"gen15"}

Depois de realizado o POST a nuvem irá responder com:

["Status","OK"]

Discador - Possíveis Respostas POST JSON

["Status","ID CAMPANHA NAO CADASTRADO"]

Indica que o ID da CAMPANHA é invalido para o login e senha informado.

["Status","Login/Senha Invalido”]

Indica que as credenciais estão invalidas.

Discador - Exemplo de código em Python

```
import requests
from requests.structures import CaseInsensitiveDict

url = “https://endereco_da_central/api.php”

headers = CaseInsensitiveDict()
headers["Content-Type"] = "application/json"

data =
‘{"acao":"insert","idcampanha":"583","login":"XXXXXXX","senha":"XXXXXXXX","token":"X
XXX","idcrm":"1234","cliente":"Leandro","cpf":"346150521852","ddd1":"11","tel1":"4118
7777","ddd2":"21","tel2":"12345678"}'


resp = requests.post(url, headers=headers, data=data)

print(resp.status_code)
```

Discador - Exemplo de Código em JavaScript

```
var url = “https://endereco_da_central/api.php”;

var xhr = new XMLHttpRequest();
xhr.open("POST", url);

xhr.setRequestHeader("Content-Type", "application/json");

xhr.onreadystatechange = function () {
   if (xhr.readyState === 4) {
      console.log(xhr.status);
      console.log(xhr.responseText);
   }};

var data =
‘{"acao":"insert","idcampanha":"583","login":"XXXXXX","senha":"XXXXXXXX","token":"XX
XX","idcrm":"1234","cliente":"Leandro","cpf":"346150521852","ddd1":"11","tel1":"41187
777","ddd2":"21","tel2":"12345678"}';

xhr.send(data);
```

Discador - Exemplo de Código em PHP

```
<?php

$url = "https://endereco_da_central/api.php";

$curl = curl_init($url);
curl_setopt($curl, CURLOPT_URL, $url);
curl_setopt($curl, CURLOPT_POST, true);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);

$headers = array(
   "Content-Type: application/json",
);
curl_setopt($curl, CURLOPT_HTTPHEADER, $headers);

$data =
‘{"acao":"insert","idcampanha":"583","login":"XXXXX","senha":"XXXXXXX","token":"XXXX
","idcrm":"1234","cliente":"Leandro","cpf":"346150521852","ddd1":"11","tel1":"4118777
7","ddd2":"21","tel2":"12345678"}';

curl_setopt($curl, CURLOPT_POSTFIELDS, $data);

//for debug only!
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);

$resp = curl_exec($curl);
curl_close($curl);
var_dump($resp);

?>
```

Discador - Recebendo Informações do Mailing

Com essa função é possível receber atualização dos contatos em uma campanha pré configurada na nuvem ou central física.

Essa facilidade possibilita receber o retorno da discagem ou contatos que ainda não foram efetivados.

Metodo *POST*

Formato: *JSON*

acao; -> statusdiscador (Para resultado dos contatos)
Idcampanha; -> Deverá ser preenchido com o valor do ID da campanha criada na nuvem.

Discador - Exemplo de Requisição - Resposta Contato

```
#!/bin/bash

curl -X POST https://endereco_da_central/api.php -H "Content-Type: application/json" -d
“{\”acao\”:\”statusdiscador\”,\”idcampanha\":\"583\",\"login\":\"XXXXXXX\",\"senha\":
\"XXXXXXXX\",\"token\":\"XXXX\"}"
```

Discador - Exemplo de Resposta do Contato

[{"0":"1435743","id":"1435743","1":"NOME DO RESPONSAVEL","cliente":"NOME DO RESPONSAVEL","2":"1","ddd":"1","3":"1","numero":"1","4":"","endereco":"","5":"","bairro":"","6":"FILIAL DO CADASTRO","cidade":"FILIAL DO CADASTRO","7":"","estado":"","8":"","cep":"","9":"NOME DO RESPONSAVEL","contato":"NOME DO RESPONSAVEL","10":"Descartado","status":"Descartado","11":"0000-00-00 00:00:00","agendamento":"0000-00-00 00:00:00","12":"09:09:40","iniciodialing":"09:09:40","13":"","ocorrencia":"","14":"0000-00-00 00:00:00","datendimento":"0000-00-00 00:00:00","15":"","obs":"","16":"","agente":"","17":"","dpto":"","18":"EP_TRAD_30.12","campanha":"EP_TRAD_30.12","19":"2","tentativas":"2","20":"CONTA","conta":"CONTA","21":"CONTA_IDENTIFICADOR","cuje":"CONTA_IDENTIFICADOR","22":"","bina":"","23":"","status_ligacao":"","24":"CPF","gen1":"CPF","25":"LIMITE DISPONIVEL","gen2":"LIMITE DISPONIVEL","26":"FILIAL DO CADASTRO","gen3":"FILIAL DO CADASTRO","27":"ACAO","gen4":"ACAO","28":"CAMPO1","gen5":"CAMPO1","29":"CAMPO2","gen6":"CAMPO2","30":"ORDEM","gen7":"ORDEM","31":"TURNO","gen8":"TURNO","32":"","gen9":"","33":"","gen10":"","34":"","gen11":"","35":"","gen12":"","36":"","gen13":"","37":"","gen14":"","38":"","gen15":"","39":"","gen16":"","40":"","gen17":"","41":"","gen18":"","42":"","gen19":"","43":"","gen20":"","44":"","gen21":"","45":"","gen22":"","46":"","tab1":"","47":"","tab2":"","48":"","tab3":"","49":"","cob":"","50":"1435743","id_registro":"1435743","51":"LOCAL","tipo_de_chamada":"LOCAL","52":"2020-12-30 08:12:43","data_insert":"2020-12-30 08:12:43"}]

Discador - Campo STATUS

Na resposta de cada contato teremos o campo STATUS, esse campo poderá ter os seguintes valores:

- AGUARDANDO (Ainda não foi discado para esse numero do contato)
- CONCLUIDO (Ja houve conversação superior a 10 segundos)
- Descartado (Tentativa de contato porem não houve atendimento)
- Dialing (Em discagem ou tratamento pelo Discador)
- EXCEDIDO (Ja tentou 5x o contato nesse numero).

Discador - Informações da Chamada Concluída

É possível receber as informações detalhadas da chamada do discador caso o status do retorno seja CONCLUIDO.

*Status “CONCLUIDO" são todas as chamadas completadas pelo discador automático que foram transferidas para o Operador e tiveram conversação.

Nesse caso necessário fazer uma requisição do tipo *POST* em formato *JSON*

{“acao”:"statusdetalhado","idcampanha":"583","idmailing":"1435749","login":"XXXXX","senha":"XXXXXX","token":"XXXX"}

O campo idmailing deverá ser preenchido com o valor do campo ID do contato, esse campo pode ser obtido pelo método STATUS.

Discador - Exemplo de Resposta:

{"id":"9050292","conta":"0191","nomeoperador":"DANIELE","ramal":"0191","tipo":"DISCADOR","chamada":"S","datahora":"2020-12-30 09:58:12”,"duracao":"335","numero":"123456789","tempofila":null,"abandonoramal":null,"abandonofila":null,"naoatendidas":"0","statuschamada":"ANSWER","grupo":"9991","aguardouatendimento":"1","uniqueid":"1609333069.28159","gravacao":"2020\/12\/30\/9991-E-123123123123123","ip":"","backup":"","caminho":"","time":"","dac_aut":"","nota_atendimento":"","t_chamando":"","dialed_time":"1","tempo_pausa":"0","tempo_logado":"0","setor_nome":"","sidehangup":"CLIENTE","cliente":"Trial-QQ","campanha":"583","id_mailing":"1435749","tronco":"0191","codigo_conta":"--SEM CONTA ","tipo_chamada":"DDI","operadora":"","call_entrada":"123123123123","ip_origem":"","operadora_cadup":"","log":"1","tarifa":"","modulacao":"","nota_atendimento2":"","valor_custo":"","valor_cobrado":"","protocolo":"","tag":"","tamanho_gravacao":"553278","v5":"1","ura":"","ftp":"2020-12-31 01:03:32","enlace":"EXTERNA"}

Discador - Exemplo de Requisição Status Detalhado

Exemplo Código JavaScript

```
var url = "https://endereco_da_central/api.php";
var xhr = new XMLHttpRequest();
xhr.open("POST", url);
xhr.setRequestHeader("Content-Type", "application/json");
xhr.onreadystatechange = function () {
   if (xhr.readyState === 4) {
      console.log(xhr.status);
      console.log(xhr.responseText);
   }};
var data =
‘{“acao":"statusdetalhado","idcampanha":"583","idmailing":"1435749","login":"XXXXXX",
"senha":"XXXXXX","token":"XXXX"}';

xhr.send(data);
```

Chamadas - Informações de Chamadas e Gravações Report

Com esse método é possível obter informações detalhadas das chamadas realizadas e recebidas de sua central na nuvem.

Deverá ser realizado um POST com formato JSON.

{"acao":"statusreport","login":"XXXXX","senha":"XXXXX","token":"XXXX","datainicial":"2020-01-02 00:00:00”}

O campo data deverá ser preenchido no formato Americano, sendo ano com 4 dígitos mes (dois dígitos) e dia (dois dígitos) YYYY-MM-DD HH:ii:ss, caso venha vazio será considerada a data do dia atual.

A quantidade máxima de registros retornados será sempre de 500 ou menos.

Chamadas - Exemplo de Requisição

Exemplo no formato Python

```
import requests

from requests.structures import CaseInsensitiveDict

url = "https://endereco_da_central/api.php"

headers = CaseInsensitiveDict()

headers["Content-Type"] = "application/json"

data = ‘{“acao":"statusreport","login":"XXXXX","senha":"XXXXX",
"token":"XXXX","inicio":"2020-02-01 00:00:00"}'

resp = requests.post(url, headers=headers, data=data)

print(resp.status_code)
```

Callcenter -Informações de Login, Logout, Pausa

Com esse método é possível obter informações detalhadas de Login, Logout e pausas dos operadores.

Deverá ser realizado um POST com formato JSON.

{"acao":"statusoperacoes","login":"XXXXX","senha":"XXXXX","token":"XXXX","datainicial":"2020-01-02 00:00:00"}

Discador - Deletando Dados da Campanha

Com esse método é possível apagar os registros de uma campanha do discador.

Deverá ser realizado um POST com formato JSON.

{“acao":"deletediscador","login":"XXXXX","senha":"XXXXX","token":"XXXX","idcampanha":"XXXX"}

A reposta será um JSON com a quantidade de registros deletados.

Exemplo de resposta
[“Total Deletados","1528"]

Gravações - Download de Gravações

É possível fazer o download dos arquivos de gravações, basta fazer uma requisição do tipo GET HTTP.

https://endereco_da_central/gravador28/$ARQUIVO_DE_GRAVACAO.gsm

O valor do campo $ARQUIVO_DE_GRAVACAO pode ser obtido com o método de Informações de chamadas

Click to Call

Com essa função é possível fazer integrações com outros sistemas, CRM's e/ou ERP's para efetuar chamadas clicando no nome do cliente, numero ou botão personalizado na aplicação do cliente.

O sistema ou aplicação deverá enviar um ***POST*** no formato *JSON*.

acao: clicktocall

login -> Login Habilitado

senha -> Senha do usuario

token -> Chave hash fornecida

origem -> Ramal que irá originar a chamada

destino -> Numero de Telefone destino

Exemplo:

{"acao":"clicktocall","login":"root","senha":"root1234","token":"XXXX","origem":"2002","destino":"56453377"}

Retorno:

["Status","CHAMADA OK”,"ID","2002_20210108175724_357675"]

A plataforma irá retornar o ID da chamada, que poderá ser consultado posteriormente para ter todas as informações da chamada, como gravação, duração, status, ramal, numero destino e etc.

Click to Call - Exemplo de Código

```
#!/bin/bash

curl -X POST https://131.196.225.2/discador.php -H "Authorization: Bearer
{"acao":"clicktocall","login":"root","senha":"root1234","token":"XXXX","origem":"2002","destino":"41187777"}" -H "Content-Type: application/json" -d "{\"acao\":\"clicktocall\",
\"login\":\"root\",\"senha\":\"root1234\",\"origem\":\"2002\",\"destino\":\"41187777\"}"
```

O ramal de origem irá tocar e quando esse retirar do gancho a chamada será completada automaticamente.

Sincronismo de Tela

Com esse método é possível obter o status de cada ramal, a aplicação externa de CRM ou ERP conseguirá identificar a origem ou destino e executar uma ação em seu próprio software conforme o retorno da Nuvem.

acao: statusramais

{"acao":"statusramais","login":"root","senha":"root1234","token":"XXXX"}

No retorno também irão aparecer 3 campos *JSON1, JSON2 e JSON3,* esses são variáveis dinâmicas que podem ser setados em quaisquer parte da chamada como por exemplo URAS, Rotas, Ramais etc.

Sendo assim os campos *JSON1 JSON2* e *JSON3*, podem conter por exemplo o CPF do cliente, código de cliente, ou quaisquer outras informações necessárias para que a aplicação do cliente (CRM's ERP's) possam executar determinadas ações com essas informações.

Um exemplo clássico é quando o cliente digita o seu CPF na URA, a Nuvem ou central física armazena essa informação na variável JSON1.

A aplicação externa de CRM ou ERP consulta essa informação com o método statusramais e obtém o que foi digitado na URA através do campo JSON1.

Dessa forma a aplicação consegue abrir ou carregar a ficha do cliente fazendo o sincronismo de tela.

Sincronismo de Tela - Ramal 2001 conversando com 2002

```
[0] => stdClass Object
    (
        [Ramal] => 2001
        [Status] => In use
        [Direcao] => ENTRADA
        [Duracao] => 109
        [StatusCanal] => Up
        [Callerid] => 2002
        [Extension] => 2001
        [JSON1] => 12345678
        [JSON2] =>
        [JSON3] =>
        [GRAVACAO] => 2021/01/10/972064460-S-2001-1610289043.371
        [IDCHAMADA] => 1610289043.371
    )

[1] => stdClass Object
    (
        [Ramal] => 2002
        [Status] => In use
        [Direcao] => SAIDA
        [Duracao] => 109
        [StatusCanal] => Up
        [Callerid] => 2001
        [Extension] => 2001
        [JSON1] =>
        [JSON2] =>
        [JSON3] =>
        [GRAVACAO] => 2021/01/10/972064460-S-2001-1610289043.371
        [IDCHAMADA] => 1610289043.371
```

O campo gravação contem o nome do arquivo de audio e deverá ser acrescido a extensão do arquivo como por exemplo “.gsm" dessa forma é possível ja armazenar o nome do arquivo mesmo com a chamada em curso.

Sincronismo de Tela - Exemplo de Captura Status Ramais

```
<?php
$url = “https://ip_da_central/api.php”;
$curl = curl_init($url);
curl_setopt($curl, CURLOPT_URL, $url);
curl_setopt($curl, CURLOPT_POST, true);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
$data = '{"acao":"statusramais","login":"root","senha":"root1234","token":"XXXX"}';
curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
//for debug only!
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
$resp = curl_exec($curl);
curl_close($curl);
$b=0;
$a = json_decode($resp);
        while ($b < count($a)){

                $RAMAL =  $a[$b]->Ramal;
        $b++;
        }
print_r($a);
?>
```

Bloqueio/Desbloqueio Cliente Classe5

Com esse método é possível bloquear ou desbloquear uma central inteira Classe5.

Utilizado para em caso do cliente ficar inadimplente.

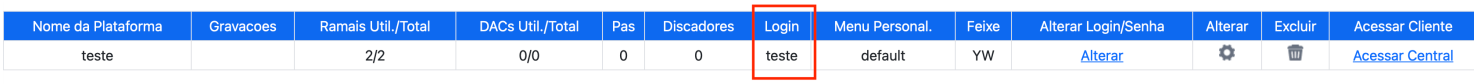
Bloqueando um Cliente

acao: bloqueioclasse5

Exemplo de Json Enviado via POST

{“login":"root","senha":"root1234","token":"XXXXXXX","acao":"bloqueioclasse5","logincliente":"teste","classe5acao":"BLOQUEAR"}

O campo logincliente é o login do classe5.



Nome da Plataforma	Gravacoes	Ramais Util./Total	DACs Util./Total	Pas	Discadores	Login	Menu Personal.	Feixe	Alterar Login/Senha	Alterar	Excluir	Acessar Cliente
teste		2/2	0/0	0	0	teste	default	YW	Alterar			Acessar Central

Desbloqueando um Cliente

acao: bloqueioclasse5

Exemplo de Json Enviado via POST

{“login":"root","senha":"root1234","token":"XXXXXXX","acao":"bloqueioclasse5","logincliente":"teste","classe5acao":"DESBLOQUEAR"}

O campo logincliente é o login do classe5.

OBS: Apenas o campo classe5acao é modificado nesse metodo.