

Data	Versão	Nota
27/10/2022	1.0	Versão inicial
19/03/2023	1.2	Add cmd enable user
24/01/2024	1.3	Add campo "SN" para todos os comandos
11/08/2024	1.4	Add protocolo Webrtc para intercom
26/02/2025	1.5	Add detecção de veia palmar
04/07/2025	1.6	Add cmd adduser;
28/10/2025	1.7	Interface atualizada para Getuserinfo e Setuserinfo

Notas

- Algumas funcionalidades requerem hardware específico (intercom, veia palmar, biometria) - consulte se o seu modelo atende as necessidades.
- Utilize o protocolo **WebSocket** para comunicação (versão RFC645513)
- Porta de escuta padrão: (configurável, preferencialmente acima de 1024)
- Dados no formato **JSON. Regras de formatação:**
Todos os pares chave-valor utilizam caracteres minúsculos (**lower-case**)

Noções básicas iniciais

- O Evo facial atua como *Client*. Ao fazer a configuração e aponta-lo para um servidor, ele enviará o "cmd": "reg", ao receber o retorno o equipamento esta conectado. Caso a conexão seja interrompida o equipamento enviara "cmd": "reg" novamente.

Valores de `backupnum`

- `0 a 9` : Impressão digital
- `10` : Senha
- `11` : Cartão RFID
- `50` : Foto (Base64)

Nota: Um usuário pode ter:

- Até 10 impressões digitais
- Uma senha
- Um cartão RFID
- Uma face

1. Equipamento Ativo Enviando Dados para o Servidor

1.1 Registro

Mensagem enviada pelo equipamento:

```
{
  "cmd": "reg",
  "sn": "ABCD12345678",
  "devinfo": {
    "modelname": "AiFace",
    "usersize": 5000,
    "facesize": 5000,
    "fpsize": 0,
    "palmsize": 0,
    "cardsize": 5000,
    "pwdsize": 5000,
    "logsize": 500000,
    "useduser": 2,
    "usedface": 2,
    "usedfp": 0,
    "usedpalm": 0,
    "usedcard": 1,
    "usedpwd": 1,
    "usedlog": 22,
    "usednewlog": 16,
    "usedrtlog": 16,
    "netinuse": 1,
    "usb4g": 0,
    "fpalgo": "thbio3.0",
    "firmware": "VERSÃO DO FIRMWARE",
    "time": "2024-12-31 00:00:00",
    "intercom": 0,
    "floors": 48,
    "charid": 1,
    "useosdp": 0,
    "dislanguage": 524417,
    "curip": "IP DO EQUIPAMENTO",
    "mac": "ENDEREÇO MAC"
  }
}
```

Resposta do servidor:

- Sucesso

```
{
  "ret": "reg",
  "result": true,
  "cloudtime": "2023-12-23 13:49:30",
  "nosenduser": true
}
```

• Falha

```
{
  "ret": "reg",
  "result": false,
  "reason": "Mensagem" //A Mensagem aparecera na tela do equip.
}
```

1.2 Enviar Logs

Mensagem enviada pelo Equipamento:

```
{
  "cmd": "sendlog",
  "sn": "ABCD12345678",
  "count": 2,
  "logindex": 10,
  "record": [
    {
      "enrollid": 1,
      "time": "2023-12-23 13:49:30",
      "mode": 0, // 1 == biometria, 2 == senha, 8 == face
      "inout": 0, // No modo "mestre - escravo" mostra o sentido (1 == saida)
      "event": 0,
      "temp": 36.5,
      "verifymode": 13,
      "image": "gesg524hgd"
    },
    {
      "enrollid": 1,
      "time": "2023-12-23 13:51:40",
      "mode": 0,
      "inout": 1,
      "event": 0,
      "temp": 36.5,
      "verifymode": 13,
      "image": "gesg524hgd"
    }
  ]
}
```

O equipamento envia este comando quando há registros em sua memória interna que ainda não foram recebidos pelo servidor. Quando o equipamento recebe o retorno positivo do servidor, ele cessa de mandar aqueles registros.

Os parâmetros `access` e `message` são pertinentes aos equipamentos que trabalham de forma online. Ao acessar menu > servidor > servidor valida > sim/renúncia offline, o equipamento irá considerar estes campos. O campo `access` com valor 1 permite o registro, caso seja 0, não permite o registro. O campo `message` mostra a mensagem na tela no ato do reconhecimento.

Isso é ideal para controlar o acesso de forma mais flexível ao equipamento, criando a lógica para aquele usuário e alterando estes campos conforme necessidade.

Resposta do servidor:

- **Sucesso**

```
{
  "ret": "sendlog",
  "result": true,
  "count": 2,
  "logindex": 10,
  "cloudtime": "2023-12-23 13:49:30",
  "access": 1,
  "message": "message",
  "fontsize": 32, // tamanho da fonte, (10 12 14 16 18 20 22 24 28 32 40)
  "text_color": [255, 0, 0] // Cor da fonte, padrao RGB
}

// Os parametros "fontsize" e "text_color" precisam das seguintes configurações no
equipamento:

// Menu > Comunicação > Servidor
"Servidor valida": Sim // pode ser chamado também servidor ratify

// Menu > Sistema > aba: dispositivo
"Label_Style": "MESSAGE_BAR"
```

2. Servidor Ativo Enviando Dados para o Equipamento

2.1 Obter Lista de Usuários

Mensagem enviada pelo servidor:

```
{
  "cmd": "getuserlist",
  "stn": true // true == 1º pacote, false == demais pacotes
}
```

Resposta do equipamento:

- **Com usuários:**

```
{
  "ret": "getuserlist",
  "result": true,
  "count": 40,
  "from": 0,
  "to": 39,
  "record": [
    {"enrollid": 1, "admin": 0, "backupnum": 0},
    {"enrollid": 2, "admin": 1, "backupnum": 0},
    {"enrollid": 39, "admin": 0, "backupnum": 10}
  ]
}
```

• Sem usuários:

```
{
  "ret": "getuserlist",
  "result": true,
  "count": 0,
  "from": 0,
  "to": 0,
  "record": []
}
```

2.2 Obter Informações do Usuário

Mensagem enviada pelo servidor:

```
{
  "cmd": "getuserinfo",
  "enrollid": 1,
  "backupnum": 50 //Caso mande backupnum, retornará simplificado, com o conteúdo da
  face
}
```

Resposta do equipamento:

• Sucesso

```
{
  "ret": "getuserinfo", //retorno simplificado com face
  "result": true,
  "enrollid": 1,
  "name": "Fulano",
  "backupnum": 50,
  "admin": 0,
  "record": "aabbccddeeffggddssiifdjdkjfkjdsjlkjal"
}
```

```
{
    "ret":"getuserinfo", //retorno convencional, sem backupnum
    "result":true,
    "sn":"ABCD12345678",
    "enrollid":1,
    "name":"Fulano",
    "departmant":"","
    "admin":0,
    "pwd":0,
    "enable":1,
    "groupid":2
}
```

• Falha

```
{
    "ret": "getuserinfo",
    "result": false,
    "reason": 1
}
```

2.3 Definir/Enviar Informações de Usuário

Mensagem enviada pelo servidor:

```
{
    "cmd": "setuserinfo",
    "enrollid": 1,
    "name": "Fulano",
    "verifymode": 0, // Modo de verificação do usuário, lista abaixo
    "department": "RH",
    "card": 123456,
    "pwd": 123456,
    "enable":1, // 1 = usuário habilitado, 0 = desabilitado
    "shiftid": 0, // id do turno, 0 = padrão do equipamento
    "zoneid": 0, // id do perfil de horário
    "groupid":0, // id do grupo de acesso
    "access_times":0, //Quantidade de acesso, 0 = ilimitado
    "postid": 0, //id do departamento, ignorar caso use o campo "department"
    "starttime": "2023-03-02 00:00:00",
    "endtime": "2023-03-30 00:00:00",
    "backupnum": 0,
    "record": "aabbccddeeffggddssiifdjdkjfkjdsjlkjalflsgsadg"
}
```

verifymode:

0 = padrão do equipamento,

1 = biometria, // em dispositivos compatíveis

2 = senha,

3 = senha,

4 = biometria + senha
5 = biometria + cartão
8 = face
9 = face + senha
10 = face + cartão
11 = cartão + senha
12 = biometria + face
13 = qr code
14 = face + cartão OU senha
16 = palma da mão // em dispositivos compatíveis
19 = verificação 1:N
20 = qualquer método
21 = palma da mão + face
22 = biometria + palma da mão
23 = cartão + palma da mão
24 = palma da mão + senha

Resposta do equipamento:

• Sucesso

```
{  
  "ret": "setuserinfo",  
  "result": true  
}
```

• Falha

```
{  
  "ret": "setuserinfo",  
  "result": false,  
  "reason": 1  
}
```

Legenda do `setuserinfo` :

Reason: 1 -> Imagem muito grande;

Reason: 3 -> Equipamento ocupado;

Reason: 5 -> Face não identificada na imagem;

Reason: 6 -> Mais de uma face na imagem;

Reason: 7 -> Largura ou altura da imagem muito larga;

Reason: 8 -> Face muito grande;

Reason: 9 -> Face muito pequena;

Reason 10 -> Imagem com qualidade muito baixa;

Reason: 11 -> imagem parcial da face (angulada, parcialmente coberta etc);

Reason: 12 -> Face já cadastrada;

Outro: Decodificação da imagem falhou

2.4 Deletar Usuários

Para deletar o usuário do equipamento como um todo, utilize `"backupnum": 12`
Caso queira apagar algum dos campos, utilize o valor respectivo de `backupnum`

Mensagem enviada pelo servidor:

```
{
  "cmd": "deleteuser",
  "enrollid": 1,
  "backupnum": 12 //caso especificado algo diferente de 12, sera apagado o backupnum
respectivo, ex. 50 remove a face associada a esse ID
}
```

Resposta do equipamento:

• Sucesso

```
{
  "ret": "deleteuser",
  "result": true
}
```

• Falha

```
{
  "ret": "deleteuser",
  "result": false,
  "reason": 1
}
```

2.5 Obter Nome do Usuário

Mensagem enviada pelo servidor:

```
{
  "cmd": "getusername",
  "enrollid": 1
}
```

Resposta do equipamento:

• Sucesso

```
{
  "ret": "getusername",
  "result": true,
  "record": "Fulano"
}
```

- **Falha**

```
{
  "ret": "getusername",
  "result": false,
  "reason": 1
}
```

2.6 Definir Nome dos Usuários

Mensagem enviada pelo servidor:

```
{
  "cmd": "setusername",
  "count": 2,
  "record": [
    {"enrollid": 1, "name": "Fulano"},
    {"enrollid": 2, "name": "Ciclano"}
  ]
}
```

Resposta do equipamento:

- **Sucesso**

```
{
  "ret": "setusername",
  "result": true
}
```

- **Falha**

```
{
  "ret": "setusername",
  "result": false,
  "reason": 1
}
```

2.7 Habilitar Usuário

Mensagem enviada pelo servidor:

```
{
  "cmd": "enableuser",
  "enrollid": 1,
  "enflag": 1
}
```

Resposta do equipamento:

- Sucesso

```
{
  "ret": "enableuser",
  "result": true
}
```

- Falha

```
{
  "ret": "enableuser",
  "result": false,
  "reason": 1
}
```

2.8 Desabilitar Usuário

Mensagem enviada pelo servidor:

```
{
  "cmd": "enableuser",
  "enrollid": 1,
  "enflag": 0
}
```

Resposta do equipamento:

- Sucesso

```
{
  "ret": "enableuser",
  "result": true
}
```

- Falha

```
{
  "ret": "enableuser",
  "result": false,
  "reason": 1
}
```

2.9 Remover Todos os Usuários

Mensagem enviada pelo servidor:

```
{
  "cmd": "cleanuser"
}
```

Resposta do equipamento:

- Sucesso

```
{
  "ret": "cleanuser",
  "result": true
}
```

- Falha

```
{
  "ret": "cleanuser",
  "result": false,
  "reason": 1
}
```

2.10 Obter Logs Novos

Mensagem enviada pelo servidor:

```
{
  "cmd": "getnewlog",
  "stn": true
}
```

Resposta do equipamento:

- Com logs

```
{
  "ret": "getnewlog",
  "result": true,
  "count": 1000,
  "from": 0,
  "to": 49,
  "record": [
    {
      "enrollid": 1,
      "time": "2023-12-23 13:49:30",
      "mode": 0,
      "inout": 0,
      "event": 0
    }
  ]
}
```

- Sem logs

```
{
  "ret": "getnewlog",
  "result": true,
  "count": 0,
  "from": 0,
  "to": 0,
  "record": []
}
```

2.11 Obter Todos os Logs

Mensagem enviada pelo servidor:

```
{
  "cmd": "getalllog",
  "stn": true,
  "from": "2023-12-01",
  "to": "2023-12-24"
}
```

2.12 Limpar Todos os Logs

Mensagem enviada pelo servidor:

```
{
  "cmd": "cleanlog"
}
```

Resposta do equipamento:

- Sucesso

```
{
  "ret": "cleanlog",
  "result": true
}
```

- **Falha**

```
{
  "ret": "cleanlog",
  "result": false,
  "reason": 1
}
```

2.13 Limpar Aparelho

Mensagem enviada pelo servidor:

```
{
  "cmd": "initsys"
}
```

Resposta do equipamento:

- **Sucesso**

```
{
  "ret": "initsys",
  "result": true
}
```

- **Falha**

```
{
  "ret": "initsys",
  "result": false,
  "reason": 1
}
```

2.14 Reiniciar Aparelho

Mensagem enviada pelo servidor:

```
{
  "cmd": "reboot"
}
```

Nota: Não há retorno para este comando.

2.15 Remover Todos os Administradores

Mensagem enviada pelo servidor:

```
{  
  "cmd": "cleanadmin"  
}
```

2.16 Definir Data e Hora

Mensagem enviada pelo servidor:

```
{  
  "cmd": "settime",  
  "cloudtime": "2023-12-23 13:49:30"  
}
```

2.17 Definir Parâmetros do Equipamento

Mensagem enviada pelo servidor:

```
{  
  "cmd": "setdevinfo",  
  "deviceid": 1,  
  "language": 0,  
  "volume": 0,  
  "screensaver": 0, //1 - 255 segundos, delay para a protetor de tela  
  "verifymode": 0,  
  "sleep": 0, // 1 == apaga a tela mas o sensor permanece ativo  
  "usefpnum": 0, //máximo de digitais por pessoa (equipamentos c/ leitor de digital)  
  "loghint": 1000, //Aviso quando faltarem menos que X logs na memória, 0 sem aviso  
  "reverifytime": 0 //  
}
```

2.18 Ativar ou Desativar "Servidor Ratify"

Mensagem enviada pelo servidor:

```
{  
  "cmd": "setdevinfo",  
  "server_verify": 1  
}
```

2.19 Abertura remota de porta

Mensagem enviada pelo servidor:

```
{
  "cmd": "opendoor",
  "doornum": 1
}
```

2.20 Obter parâmetros do equipamento

```
{
  "cmd": "getdevinfo"
}
```

Sucesso

```
{
  "ret": "getdevinfo",
  "sn": "ABCD12345678",
  "result": true,
  "deviceid": 1,
  "language": 0,
  "volume": 0,
  "screensaver": 0,
  "verifymode": 0,
  "sleep": 0,
  "userfpnum": 3,
  "loghint": 1000,
  "reverifytime": 0
}
```

Falha

```
{
  "ret": "getdevinfo",
  "sn": "ABCD12345678",
  "result": false,
  "reason": 1
}
```

2.21 Definir parâmetros de acesso

```
{
  "cmd":"setdevlock",
  "opendelay":5, //Delay p/ abertura de porta
  "doorsensor":0, //Tipo do sensor: 0 == desabilitado, 1 == NC (Normal fechado), 2 ==
NO (Normal aberto)
  "alarmdelay":0, //Tempo para alarme de porta aberta, 1 - 255, 0 == sem alarme
  "threat":0, //Alarme: 0 == desligado, 1 == abrir porta com alarme, 2 == apenas soar
alarme, 3 == apenas abrir porta
  "InputAlarm":0, //0 == desabilitado, 1 == output alarme 1, 2 == output alarme 2
  "antpass": 0, //0 == desabilitado, 1 == mestre dentro, 2 == mestre para fora
  "interlock":0, //0 == desativado, 1 = ativado
  "mutiopen":0, //Requer abertura simultanea, 0 == deslig.; 1 - 4:requer que N pessoas
acessem simultaneamente para abrir a porta
  "tryalarm":0, //0 == desativado, 1 - 10 == Quando alguém falhar o reconhecimento N
vezes, o alarme sera acionado
  "tamper":0, //0 == desativado, 1 == ativado
  "wgformat":0, //Formato do wiegand, 0 == 26 1 == 34
  "wgoutput": 0, //dado transmitido via wiegand: 0 == enrollid, 1 == 1+enroll id, 2 ==
deviceid + enroll id
  "cardoutput":0, //Transmite o valor do cartao, biometria etc via wiegand 0 ==
disable, 1 == enable;
  "access_times":0, //Quantidade de acessos permitidos, 0 == desabilitado
  "dayzone":[ //Max 8 perfis de dia.
    { // cada dia pode conter no max 5 sessoes
      "day": [ //a este dia ficará atribuido "day":1 - ordem sequencial
        {"section":"07:00~08:30"},
        {"section":"12:45~13:15"},
        {"section":"14:00~15:00"},
        {"section":"16:01~16:59"},
        {"section":"18:00~18:30"}]
    },
    {
      "day": [ // "day":2
        {"section":"00:01~23:59"}] //nao sao necessarios todos
os dias nem todas as sessoes
    }
  ],
  "weekzone":[ //Max 8 perfis de semana. A semana e vinculado ao enrollID.
    {
      "week":[ //week 1, e assim por diante
        {"day":2}, //domingo
        {"day":1}, //segunda
        {"day":1}, //terça
        {"day":1}, //quarta
        {"day":1}, //quinta
        {"day":1}, //sexta
        {"day":2} //sabado
      ]
    },
    {
      ...
    }
  ]
}
```

```
{
  "ret": "setdevlock",
  "sn": "ABCD12345678",
  "result": true
}
```

Falha

```
{
  "ret": "setdevlock",
  "sn": "ABCD12345678",
  "result": false,
  "reason": 1
}
```

Dicas

Apenas as partes que importam do comando pode ser enviadas, por exemplo, caso só haja uma dia e semana:

```
{"cmd": "setdevlock", "dayzone": [{"day": [{"section": "07:00~18:00"}]}, {"weekzone": [{"week": [{"day": 1}]}]}
```

Para o acesso permitido o dia todo dentro de uma semana, pode se utilizar "00:0123:59".

Para o bloqueio total no dia, utilize "00:0000:00"

2.22 Obter parâmetros de acesso do equipamento

```
{
  "cmd": "getdevlock"
}
```

Sucesso

```
{
  "ret":"getdevlock",
  "sn": "ABCD12345678",
  "result":true,
  "opendelay":5,
  "doorsensor":0,
  "alarmdelay":0,
  "threat":0,
  "inputalarm":0,
  "antpass":0,
  "interlock":0,
  "multiopen":0,
  "tryalarm":0,
  "tamper":0,
  "wgformat":26,
  "wgoutput":0,
  "cardoutput":0,
  "dayzone":[
    {
      "day":[{"section":"00:01~23:59"}]
    },
    {
      "day":[{"section":"00:00~00:00"}]
    }
  ],
  "weekzone":[
    {
      "week":[
        {"day":2},
        {"day":1},
        {"day":1},
        {"day":1},
        {"day":1},
        {"day":1},
        {"day":2}
      ]
    }
  ]
}]}}
```

Falha

```
{
  "ret":"setdevlock",
  "sn":"ABCD12345678",
  "result":false,
  "reason":1
}
```

2.23 Obter parâmetros de acesso do usuário

```
{
  "cmd":"getuserlock",
  "enrollid":1
}
```

Sucesso

```
{
  "cmd":"getuserlock",
  "sn":"ABCD12345678",
  "result":true,
  "enrollid":1,
  "weekzone":1, // perfil da semana vinculado ao enrollId
  "group":0,
  "starttime":"2025-10-10 01:00:00", //data e hora, formato AAAA-MM-DD HH:MM:SS
  "endtime":"2099-03-25 23:00:00"
}
```

Falha

```
{
  "cmd":"getuserlock",
  "sn":"ABCD12345678",
  "result":false,
  "reason":1
}
```

2.24 Definir parâmetros de acesso do usuário

```
{
  "cmd":"setuserlock",
  "count":40,
  "record":[
    {
      "enrollid":1,
      "weekzone":1,
      "group":1
    },
    {
      "enrollid":2,
      "weekzone":1,
      "group":1
    },
    {...}
  ]
}
```

Sucesso

```
{
  "ret": "setuserlock",
  "sn": "ABCD12345678",
  "result": true
}
```

Falha

```
{
  "ret": "setuserlock",
  "sn": "ABCD12345678",
  "result": false,
  "reason": 1
}
```

2.24 Deletar parâmetros de acesso do usuário

```
{
  "cmd": "deleteuserlock",
  "enrollid": 1
}
```

Sucesso

```
{
  "ret": "deleteuserlock",
  "sn": "ABCD12345678",
  "result": true
}
```

Falha

```
{
  "ret": "deleteuserlock",
  "sn": "ABCD12345678",
  "result": false,
  "reason": 1
}
```

2.25 Limpar parâmetros de acesso do dispositivo

```
{
  "cmd": "cleanuserlock",
}
```

Sucesso

```
{
  "ret": "cleanuserlock",
  "sn": "ABCD12345678",
  "result": true
}
```

Falha

```
{
  "ret": "cleanuserlock",
  "sn": "ABCD12345678",
  "result": false,
  "reason": 1
}
```

2.26 Obter data e hora do equipamento

```
{
  "cmd": "gettime",
}
```

Sucesso

```
{
  "ret": "gettime",
  "sn": "ABCD12345678",
  "time": "2025-10-10 08:00:00"
}
```

2.27 Equipamento envia QR code ao servidor

```
{
  "cmd":"sendqrcode",
  "sn":"ABCD12345678",
  "record":"valor_qr_code"
}

//Resposta do servidor
{
  "ret":"sendqrcode",
  "sn":"ABCD12345678",
  "result":true,
  "access": 1, //pode retornar 0 caso nao esteja autorizado, e a porta nao abra
  "enrollid":10, //para mostrar na tela
  "username":"Fulano", //para mostrar na tela
  "message": "Acesso OK", //para mostrar na tela
  "fontsize":32, // (10 12 14 16 18 20 22 24 28 32 40)
  "text_color":[255,0,0] //RGB
}
```

2.28 Definir parâmetros de enquete

```
{
  "cmd":"setquestionnaire",
  "sn":"ABCD12345678",
  "result":true,
  "tittle":"Proteina", //titulo no equipamento
  "errmsg":"Favor selecionar", //quando uma op. e obrigatoria
  "radio":true, //true == multiplas escolhas, false == apenas umas
  "optionflag":0, // em caso de multiplas escolhas, forca a obrigatoria
  "usequestion":true,
  "useschedule":false,
  "card":0, //passar o cartao para iniciar a enquete
  "items": ["Bife", "Frango", "Bisteca Suina", "Vegetariano"], //multipla escolhas, ate
8 opcoes. Escolha unica, ate 16 opcoes
  "Schedules": ["05:30-07:30*1", "16:30-19:00*2"]
}
```

2.29 Obter parâmetros de enquete

```
{
  "cmd":"getquestionnaire",
  "stn":"true"
}
```

Sucesso

```
{
  "ret": "getquestionnaire",
  "sn": "ABCD12345678",
  "result": true,
  "tittle": "Proteina", //titulo no equipamento
  "errmsg": "Favor selecionar", //quando uma op. e obrigatoria
  "radio": true,
  "optionflag": 0,
  "usequestion": true,
  "useschedule": false,
  "card": 0,
  "items": ["Bife", "Frango", "Bisteca Suina", "Vegetariano"],
  "Schedules": ["05:30-07:30*1", "16:30-19:00*2"]
}
```

2.30 Adicionar usuário diretamente no equipamento

```
{
  "cmd": "adduser",
  "enrollid": 1,
  "backupnum": 50, //face
  "admin": 0,
  "name": "Fulano",
  "flag": 10 // registro automatico
}
//utilize backupnum: 11 para cadastrar o cartão, 40 & 41 para a veia palmar, 0 á
9 para biometria etc
```

```
{
  "ret": "adduser",
  "result": false,
  "reason": 3,
  "mesg": "device is working"
}

//caso receba esse retorno, trate da seguinte forma
{
  "cmd": "adduser",
  "cancel": true
}
```

Firmwares especiais:

TurnstilePass

Altera o valor "Inout" para contemplar a se houve e giro e qual a direção
Valores de inout: 7 = Não girou; 6 = girou na direção A, 5 = Girou na direção B;

Exemplo de log:

```
{
  "cmd":"sendlog",
  "sn":"ZYSA11003779",
  "count":1,
  "logindex":0,
  "record":[
    {
      "enrollid":1,
      "name":"",
      "time":"2024-09-18 17:24:25",
      "mode":3, //forma de validação
      "inout":7, //Não girou
      "event":0
    }
  ]
}
```

Valores de mode: 3 == cartão, 8 == face, 16 == veia palmar, x == senha

Nada é alterado comparado ao log convencional, deve-se apenas observar o valor de `inout`

Configuração do hardware

CM -> Comum
NA -> Normalmente Aberto
NF -> Normalmente Fechado
GND -> Terra (Ground)

Ligação do acionamento do facial para a placa Semi-eletrônica:

- Fio CM e NA do relé do facial na botoeira da Semi-eletrônica
- Quando o facial acionar o rele precisa liberar ambos na Semi-eletrônica

Configuração na placa Semi-eletrônica:

Na Semi-eletrônica configurar o retorno de giro nos relés 2 e 3

Ligação do retorno de giro na Semi-eletrônica:

Quando efetuar o giro, de acordo com a configuração realizada acima, a Semi-eletrônica irá enviar um pulso negativo no FIRE IN e no EXIT BUTTON para confirmação do giro.

Relé 2 Semi-eletrônica: CM gnd - NA fio branco FIRE IN

Relé 3 Semi-eletrônica: CM gnd - NA fio amarelo BUTTON

ANTES DE LIGAR CONFERIR SE OS JUMPERS DOS RELES NÃO ESTÃO FECHADOS COM O 12V. SE ESTIVER, HAVERÁ CURTO E DANIFICARA OS EQUIPAMENTOS

Pinagem

GND: Preto - Qualquer chicote

FIRE_IN: Branco - Chicote COM (Vermelho)

BUTTON: Amarelo - Chicote ACESSO (Branco)

Configurações no Evo Facial:

Atualizar o firmware da personalização *TurnstilePass*

Acesso > Acesso > {"Tempo de acionamento": "1seg"}

Acesso > Acesso > {"Nova Verif.": "10seg"}

Turno > Aba: Regras > {"Nova Verif.": "1seg"}

3. Intercom com WebRTC

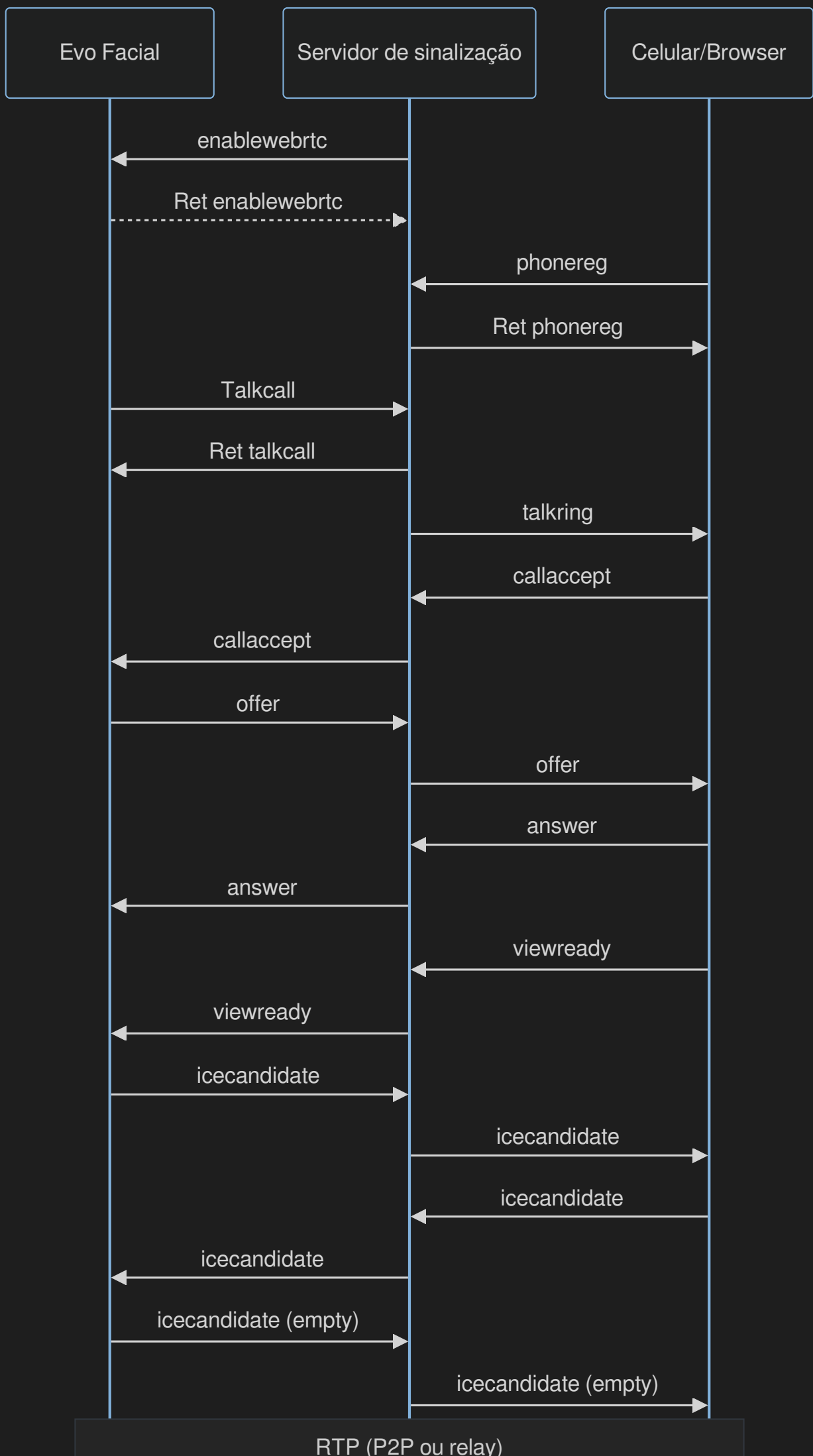
Comunicação e Preparação Prévia

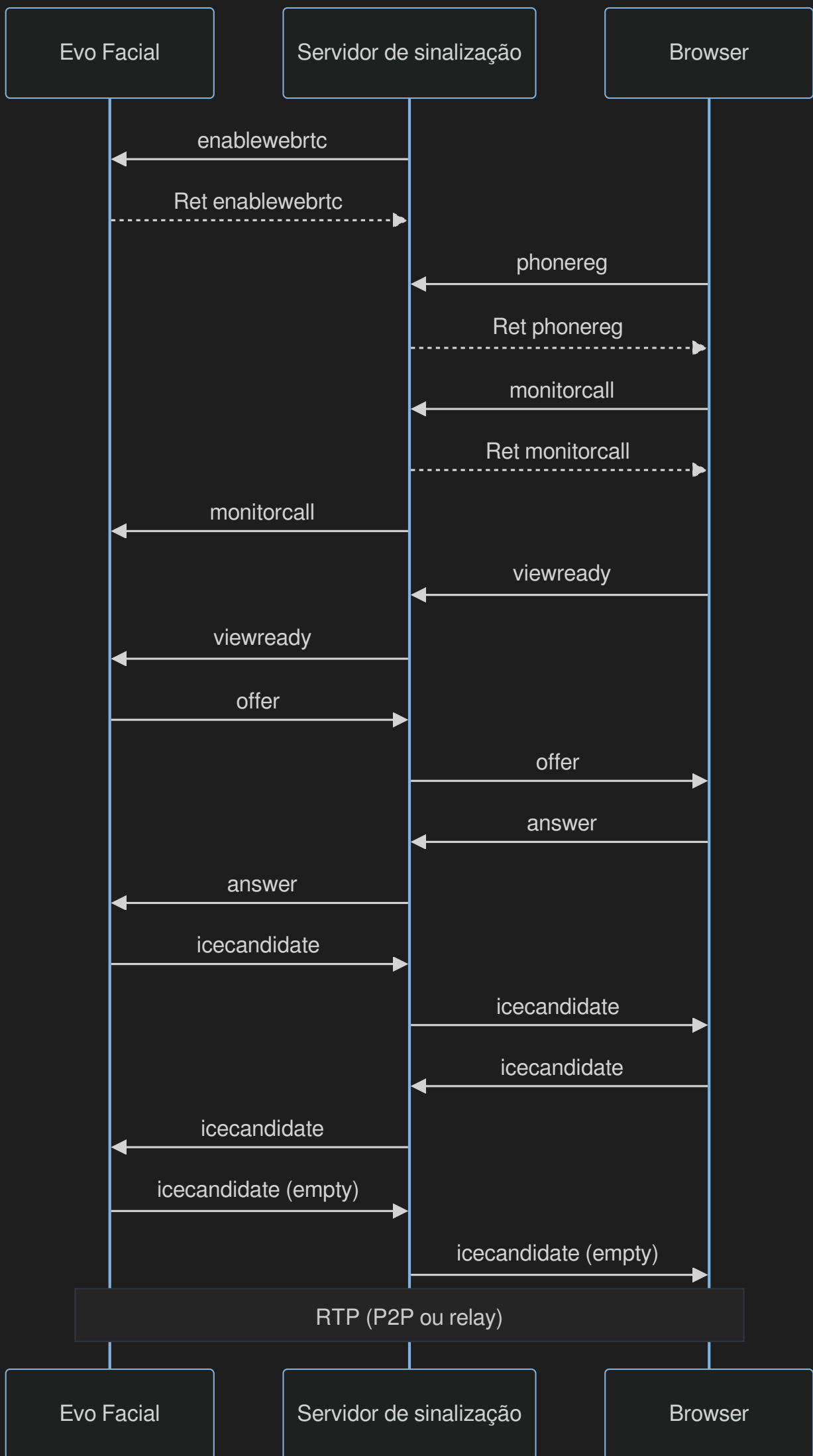
1. Complete a comunicação WebSocket+JSON para assiduidade na nuvem
2. Na interface de comunicação WebSocket original, adicione uma porta de escuta para comunicação WebSocket com o dispositivo móvel
3. Configure seu próprio servidor STUN/TURN (especialmente o servidor TURN, pois é usado para armazenar em cache o fluxo de vídeo do interfone visual)

Nota importante: No WebRTC, aproximadamente 20% das conexões não conseguem estabelecer conexão P2P diretamente e precisam usar o servidor TURN como intermediário. Como os servidores TURN requerem grande largura de banda, não existem servidores TURN gratuitos disponíveis.

Servidor STUN gratuito para testes: `stun:stun.1.google.com:19302`

3.1 Fluxo de Comunicação para Interfone Visual





3.1.1 Habilitar WebRTC

Mensagem enviada pelo servidor:

```
{
  "cmd": "enablewebrtc",
  "intercom": true,
  "monitor": true,
  "inputmode": 1,
  "stunserver": "stun:stun.l.google.com:19302",
  "turnserver": "turn:192.168.0.1:3478",
  "turnusername": "usuário",
  "turnpassword": "senha"
}
```

Resposta do equipamento:

• Sucesso

```
{
  "ret": "enablewebrtc",
  "result": true,
  "sn": "ZX0006827500"
}
```

• Falha

```
{
  "ret": "enablewebrtc",
  "result": false,
  "sn": "ZX0006827500",
  "reason": 1
}
```

Valores de inputmode:

0: directly - chamada direta para número padrão
1: enrollnumber - número de registro do usuário
2: phone number - número de telefone
3: room number - número do apartamento

3.1.2 Requisição de Chamada (TalkCall)

Mensagem enviada pelo Evo Facial:

```
{
  "cmd": "talkcall",
  "sn": "ZX0006827500",
  "inputnum": 1234,
  "inputmode": 1,
  "landscape": 0,
  "calltype": 0
}
```

Resposta do servidor:

• Sucesso

```
{
  "ret": "talkcall",
  "result": true,
  "sessionid": "2323523523",
  "enrollid": 1234,
  "username": "Fulano",
  "calltype": 0
}
```

• Falha

```
{
  "ret": "talkcall",
  "result": false,
  "reason": 1,
  "calltype": 0
}
```

3.1.3 Notificação de Toque (TalkRing)

Mensagem enviada pelo servidor para o dispositivo móvel:

```
{
  "cmd": "talkring",
  "sn": "ZX0006827500",
  "msg": "Alguém te ligou",
  "sessionid": "SESSION_ID",
  "calltype": 0
}
```

3.1.4 Aceitação de Chamada (CallAccept)

Mensagem enviada pelo servidor para o Evo Facial:

```
{
  "cmd": "callaccept",
  "sessionId": "SESSION_ID",
  "result": true,
  "calltype": 0
}
```

Mensagem de recusa:

```
{
  "cmd": "callaccept",
  "sessionId": "SESSION_ID",
  "result": false,
  "calltype": 0
}
```

3.1.5 Troca de Ofertas (Offer)

Mensagem enviada pelo Evo Facial:

```
{
  "cmd": "offer",
  "sn": "DEVICE_SERIAL_NUMBER",
  "sessionId": "SESSION_ID",
  "type": "offer",
  "sdp": "v=0\r\no=...",
  "calltype": 0
}
```

3.1.6 Resposta à Oferta (Answer)

Mensagem enviada pelo servidor:

```
{
  "cmd": "answer",
  "sessionId": "SESSION_ID",
  "type": "answer",
  "sdp": "v=0\r\no=...",
  "calltype": 0
}
```

3.1.7 Troca de Candidatos ICE (IceCandidate)

Mensagem enviada com candidato:

```
{
  "cmd": "icecandidate",
  "sessionId": "SESSION_ID",
  "candidate": "{\"candidate\":\"candidate:842163049 1 udp...\", \"sdpMid\":\"0\", \"sdpMLineIndex\":0}\",
  "calltype": 0
}
```

Mensagem de finalização de candidatos:

```
{
  "cmd": "icecandidate",
  "sn": "DEVICE_SERIAL_NUMBER",
  "sessionId": "SESSION_ID",
  "candidate": ""
}
```

3.1.8 Cancelamento de Chamada (CallCancel)

Estrutura da mensagem:

```
{
  "cmd": "callcancel",
  "sessionId": "SESSION_ID",
  "calltype": 0
}
```

3.2 Fluxo de Comunicação para Monitoramento

3.2.1 Requisição de Monitoramento (MonitorCall)

Mensagem enviada pelo servidor para o Evo Facial:

```
{
  "cmd": "monitorcall",
  "sessionId": "SESSION_ID",
  "calltype": 0
}
```

Mensagem enviada pelo navegador para o servidor:

```
{
  "cmd": "monitorcall",
  "sn": "ZX0006827500",
  "calltype": 0
}
```

Resposta do servidor:

- Sucesso

```
{
  "ret": "monitorcall",
  "result": true,
  "sessionId": "SESSION_ID",
  "calltype": 0
}
```

• Falha

```
{
  "ret": "monitorcall",
  "result": false,
  "reason": 1,
  "calltype": 0
}
```

3.2.2 Notificação de Visualização Pronta (ViewReady)

Mensagem enviada pelo navegador para o servidor:

```
{
  "cmd": "viewready",
  "sessionId": "SESSION_ID",
  "calltype": 0
}
```

Nota importante: Para o fluxo de interfone visual, a mensagem ViewReady deve ser enviada pelo navegador após receber e processar a oferta SDP e enviar a resposta SDP. Para o fluxo de monitoramento, a mensagem ViewReady deve ser enviada imediatamente após receber a resposta positiva do monitor call.

3.3 Diferenças entre Interfone Visual e Monitoramento

1. O interfone visual é iniciado pelo Evo Facial e requer que o telefone atenda antes de conectar. O som do microfone do telefone pode ser transmitido para o Evo Facial.
2. O monitoramento é iniciado pelo navegador ou telefone, não requer aceitação do Evo Facial e pode conectar diretamente. O som do microfone do telefone não pode ser ouvido no Evo Facial, que permanece em modo silencioso.

3.4 Registro de Dispositivo Móvel (PhoneReg)

Mensagem enviada pelo dispositivo móvel:

```
{
  "cmd": "phonereg",
  "phonenum": 13824406089,
  "enrollid": 23456
}
```

Resposta do servidor:

• Sucesso

```
{
  "ret": "phonereg",
  "stunserver": "stun:123.255.189.213:3478",
  "turnserver": "turn:123.255.189.213:3478",
  "turnusername": "usuário",
  "turnpassword": "senha",
  "result": true
}
```

• Falha

```
{
  "ret": "phonereg",
  "result": false,
  "reson": 1
}
```

Notas Importantes:

1. Recomenda-se usar uma porta de escuta diferente para dispositivos móveis e Evo Facial.
2. O mecanismo de sinalização WebRTC utiliza o servidor WebSocket para trocar informações de mídia (SDP) e candidatos ICE antes de estabelecer a conexão P2P.

3.5 Implementação de ID de Sessão

Por convenção, o número de série (SN) do dispositivo Evo Facial é utilizado como o ID de sessão (sessionid) em todas as comunicações WebRTC. Isso simplifica o rastreamento de sessões e garante identificadores únicos para cada dispositivo.