



A Agente como ramal

RECOMENDADO

O SipPulse AI se registra num ramal comum do FusionPBX. Tudo pela interface, por domínio (multi-tenant).

- 1 Crie o agente no painel SipPulse AI · sippulse.ai
Configure prompt, voz e ferramentas. Anote o ID `agt_xxxx`.
- 2 Crie um ramal no FusionPBX

Accounts → Extensions → + Add

Extension	4000
Description	Agente IA
Password	senha forte

3 Cadastre o registro no painel SipPulse AI

Agente → Deployment → SIP → Registro externo

Servidor	domínio-do-tenant:5060
Usuário	4000
Senha	a mesma do ramal

No multi-tenant, o usuário SIP é `4000@domínio`; confira o registro em **Status → Registrations**.

4 Teste

Ligue de qualquer ramal para **4000** e converse com o agente.

B Via gateway

Sem registro: um gateway entrega a chamada direto em `sip.sippulse.ai` quando alguém disca 4001.

1 Crie o gateway

Accounts → Gateways → + Add

Gateway	sippulse
Proxy	sip.sippulse.ai
Register	False

2 Crie a entrada no dialplan

Dialplan → Dialplan Manager → + Add

Condition	destination_number = ^4001\$
Action · bridge	sofia/gateway/sippulse/agt_xxxx

Quem discar **4001** cai no agente pelo gateway.

3 Aponte o fluxo para o agente

Dialplan → Destinations: DID → ramal 4000. **Apps → IVR Menus:** opção "falar com a assistente" → 4000. **Ring Groups:** use o 4000 como destino de timeout — se ninguém atender, a IA assume.

- > **Cabeçalhos por chamada:** numa entrada do Dialplan Manager, ação **set** com `sip_h_X-Uid=...` antes do bridge (valores em Base64; use um Lua como no infográfico do FreeSWITCH)
- > **Áudio:** libere UDP 5060 e a faixa RTP do FreeSWITCH (16384–32768 por padrão) para sip.sippulse.ai

C Gravação estéreo e monitoria

O FusionPBX já grava em **estéreo por padrão** (RECORD_STEREO no dialplan `user_record`): diarização perfeita sem conversão.

1 Ative a gravação no ramal

Accounts → Extensions → 4000 → Advanced

Record	ALL
--------	-----

Os arquivos caem em

`/var/lib/freeswitch/recordings/<domínio>/archive/ano/mês/dia/`.

2 Transcreva em lote com Python no cron

```
# /usr/local/bin/transcrever.py · cron */5
import requests, pathlib, time
REC = pathlib.Path(
    "/var/lib/freeswitch/recordings")
for wav in REC.rglob("archive/**/*.*.wav"):
    saida = wav.with_suffix(".json")
    if saida.exists(): continue
    if time.time()-wav.stat().st_mtime < 60:
        continue
    r = requests.post(
        "https://api.sippulse.ai"
        "/v1/asr/transcribe",
        headers={"api-key": "SUA_API_KEY"},
        files={"file": open(wav, "rb")},
        data={"model": "pulse-telephony",
            "response_format":
                "stereo-diarization",
            "vad_preset": "telephony",
            "language": "pt"})
    saida.write_text(r.text)
```

- > **Monitoria de 100%:** passe cada JSON na análise estruturada — **POST** `/v1/structured-analyses/{id}/execute` — sentimento, objeções, score
- > **Multi-tenant:** o caminho por domínio já separa as gravações por cliente

● **Estéreo por padrão** · o FusionPBX já grava pronto para diarização

● **Multi-tenant** · um agente de IA por domínio ou compartilhado

● **Latência** inferior a 500 ms



Comece agora: crie um agente e registre-o no seu FusionPBX em minutos

sippulse.ai · docs.sippulse.ai · api.sippulse.ai · cobrança por consumo, tipicamente abaixo de R\$ 0,75/min