Sipulse

DATASHEET

SIP VAULT

Troubleshooting de SIP como um profissional. Deixe a IA fazer os 80%.

Cada chamada SIP capturada, correlacionada e pronta pra ser lida — pelos seus engenheiros e pela IA. Um cofre tamper-evident para sinalização SIP, estatísticas RTP/RTCP, logs do OpenSIPS e CDRs, indexado por Call-ID. Um clique — ou um prompt — e a história inteira da chamada está na sua frente.

Call-ID	< 30s	eBPF	MCP
UMA CHAVE. TODA A EVIDÊNCIA.	/TROUBLESHOOT-CALL RESOLVE	AGENTE SEM DEPENDÊNCIAS	DASHBOARD + LLM, MESMA API

POR QUE ISSO IMPORTA

Suporte é o imposto silencioso sobre toda plataforma SIP.

DO CUSTO OPERACIONAL

50%

vai pra suporte técnico e troubleshooting de chamadas.

REALIDADE DE CONTRATAÇÃO

Escasso.

Engenheiros SIP qualificados são difíceis de achar, mais difíceis ainda de segurar.

CHURN DE CLIENTES

#1

Suporte ruim é o principal motivo pelo qual o cliente vai embora.

PRA ONDE O TEMPO REALMENTE VAI

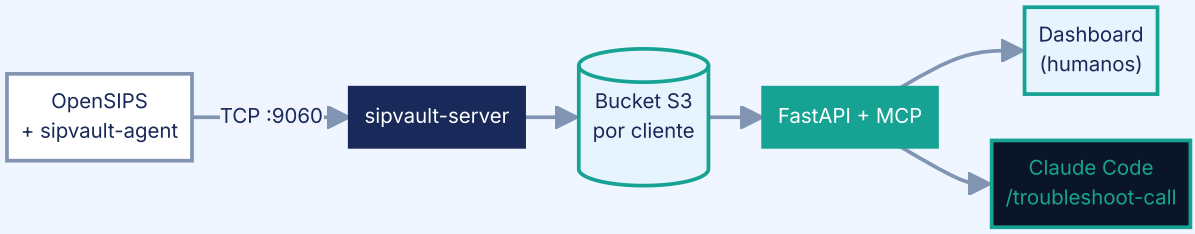
Quatro passos. Duas fases. Uma delas é enorme.

PASSO 1	PASSO 2	PASSO 3	PASSO 4
Definir e reproduzir o problema	**Coletar traces e logs**	**Analisar as informações coletadas**	**Implementar e documentar a solução**
80% do tempo		20% do tempo	

A aposta: colapsar os 80% pra perto de zero, e os 20% viram um trabalho completamente diferente. É exatamente o que o SIP VAULT faz.

ARQUITETURA

Da borda do cliente até a análise por IA.



Um contrato, dois consumidores — humanos batem no dashboard, IA bate no MCP. Mesmos dados, superfícies diferentes.

CAPACIDADES

O que cada chamada carrega

EVIDÊNCIA CAPTURADA

- Diálogo SIP INVITE completo (pcap)
- Relatórios sender + receiver de RTCP
- Estatísticas RTP (MOS, jitter, perda, latência)
- Logs do OpenSIPS, etiquetados por Call-ID
- CDR enriquecido — endereçável por Call-ID

DASHBOARD

- CDR com link VAULT por linha
- Busca e filtro (data, método, código, tags)
- Qualidade de chamada por leg (UAC vs UAS)
- Flag automática da leg ruim (MOS / jitter / perda)
- PCAP, RTCP e logs brutos a um clique

SUPERFÍCIE IA (MCP)

- FastAPI + MCP, contrato único
- Skill /troubleshoot-call Call-ID via Claude Code
- Fetch → correlate → reason → report
- Post-mortem em Markdown com evidência linkada
- Mesma fronteira de auth do dashboard

DIFERENCIAIS

Por que SIP VAULT

O CDR virou a porta de entrada

Cada chamada, cada linha, endereçável por Call-ID. Clica no VAULT e o pcap, logs e estatísticas RTP daquela chamada específica estão ali. Sem SSH, sem grep. O CDR deixou de ser só artefato de cobrança — virou o índice de tudo que aconteceu.

LLM-ready desde o desenho

O servidor MCP não é puxadinho — é o mesmo contrato que serve o dashboard. A IA tem acesso de primeira classe a cada byte de evidência.

Aberto na borda

sipvault-agent é Apache 2.0. Protocolo de fio documentado. Rode contra o servidor SipPulse ou construa o seu — você decide.

Store tamper-evident

Bucket S3 por cliente, evidência com hash de conteúdo. Quando uma chamada cai no cofre, dá pra provar que ninguém editou depois.

Respeitoso por padrão

Apenas diálogos INVITE. Nunca decodifica payload de RTP. Agente phones-home só pro servidor que você configura. Saída pela :9060, nada novo exposto.

IA POR CIMA

Um Call-ID. Um comando. Toda a história.

```
$ /troubleshoot-call <call-id>
```

Aperta enter, o Claude entra em ação: **busca** pacotes, logs e estatísticas RTP via MCP → **correlaciona** as duas pernas e as linhas do tempo → **raciocina** sobre o modo de falha real (mismatch de codec, NAT, timing, roteamento) → **reporta** um post-mortem em Markdown com evidência e próximos passos. Caso real do campo: RTP assimétrico, NAT simétrico do lado UAC, sem RTCP voltando, pico de jitter casando com um reroute na borda do cliente. Era uma hora pra um engenheiro sênior. **O Claude resolve em menos de 30 segundos.**

ESPECIFICAÇÕES TÉCNICAS

O agente — open source na borda

ESPECIFICAÇÃO	DETALHE
Modos de captura	eBPF (kernel ≥ 4.18) · libpcap fallback (CentOS 6/7)
Arquitetura	amd64, arm64 (auto-detect no install)
Dependências	Zero no modo eBPF. Binário Go estático.
Transporte	Uma conexão TCP outbound na :9060
Resiliência de rede	Buffer local de 100 MB; sem perda em queda de rede

ESPECIFICAÇÃO	DETALHE
Armazenamento	S3 por cliente, indexado por Call-ID, tamper-evident
API	FastAPI + MCP (mesmo contrato pra humanos e IA)
Skill IA	/troubleshoot-call <Call-ID> via Claude Code
Escopo	Apenas diálogos INVITE. Nunca decodifica payload RTP.
Fonte	github.com/sippulse/sipvault — Apache 2.0

COMERCIAL

Modelos de licenciamento

MODELO	DESCRIÇÃO
SaaS	Hospedado pela SipPulse. Bucket S3 por cliente. Dashboard + MCP inclusos. Mensalidade escala por volume de chamadas capturadas.
Agente open source	sipvault-agent é Apache 2.0. Use contra o servidor SipPulse ou construa o seu receiver — o protocolo de fio é documentado.
OEM / on-prem servidor	Pra operadoras que precisam do cofre inteiro dentro da própria infraestrutura. Fale com a gente.

ECOSSISTEMA

Produtos relacionados

SipPulse SoftSwitch
Plataforma Class 4/5

SipPulse SBC
Controle de borda

SipPulse BSS
Faturamento (BCORE)

SipPulse BCW
Communication Workspace

SipPulse AI
Voice agents (NIVA)

GLOSSÁRIO

Acrônimos

SIP — Session Initiation Protocol
RTP / RTCP — Real-time Transport / Control Protocol
eBPF — Extended Berkeley Packet Filter
OpenSIPS — Servidor SIP open source
Call-ID — Identificador único de diálogo SIP (RFC 3261)
MOS — Mean Opinion Score (qualidade)
MCP — Model Context Protocol
NAT — Network Address Translation
UAC / UAS — User Agent Client / Server
pcap — Arquivo de captura de pacotes
CDR — Call Detail Record
SRE — Site Reliability Engineering

SIP VAULT — cada chamada tem sua evidência. Cada evidência tem sua IA.

Veja rodando no seu stack

info@sippulse.com | sippulse.com | github.com/sippulse/sipvault

 |