

SBC-AutoOps

Design Partner Briefing

The independent truth layer for real-time voice. We read your SBC configs before deploy and tell you exactly what will break. Your configs never leave your environment.

METAVENTIONS AI

Dico Angelo and Philip Drammeh, co-founders

sbcvalidator.metaventionsai.com | June 2026 | v0.16.4



Calls fail silently. You find out from users.

Every business call crosses a Session Border Controller. One misconfiguration: one-way audio, dead trunks, one to four-plus days of repair. Engineers call the discovery mechanism the scream test. Three named incidents from the 2026 Microsoft CA migration (MC1235747):

Voipcloud

P1 across thousands of users. Untrusted root presented at the mutual-TLS handshake; Microsoft stopped accepting calls.

CallTower

Silent blackhole. Unpatched SBCs stopped OPTIONS pings; Microsoft auto-deactivated the domain. No SIP 503 warned anyone.

FusionConnect

One-way audio on an SRTP mismatch. Honest caveat: this one originated on Microsoft's media-relay side; a config check would not have caught it.

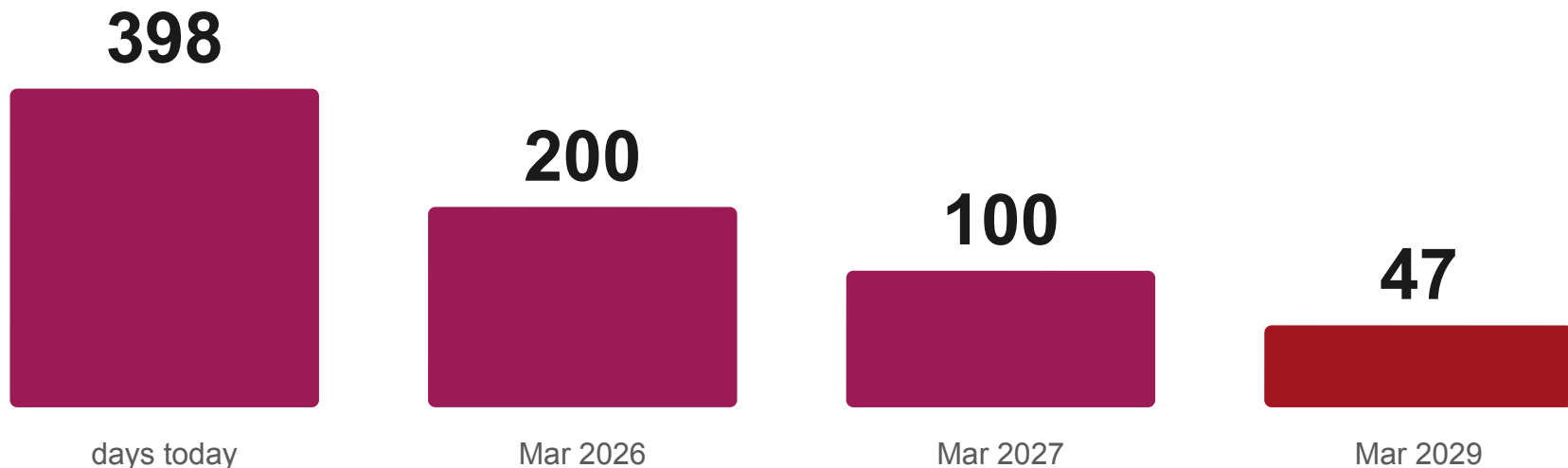
You cannot monitor a call that is cryptographically blocked from ever starting. The check has to happen before deploy.





The cadence, not the climax

June 2026: the serverAuth-EKU enforcement for Teams Direct Routing lands this month. Behind it, per CA/Browser Forum SC-081v3, public TLS certificate lifetimes compress on a fixed schedule:



Max certificate lifetime, in days. By 2029 that is roughly eight renewals a year, per SBC, across every vendor you run. Manual validation stops scaling well before 47 days. Each new wave ships as a signed rule bundle on the same engine.

**TRUST NOW ROTATES
FASTER THAN TELECOM**
**SBC FORCING EVENTS
2026-2029**

TIMELINE START

WAVE 1: MICROSOFT CA/EKU

MAR 2026
7 ROOTS REQUIRED



APR 2026
EDGE CERT ROTATION
HANDSHAKE HARD-FAIL



JUN 2026
SERVERAUTH EKU ONLY

DAYS AWAY

POLICY PROGRESSION

2026

2027

2028

2029

WAVE 2: CERT LIFETIME COMPRESSION

398 DAYS

200 DAYS

100 DAYS

LIFESPAN FLOW

47 DAYS

**8 RENEWALS A YEAR
BY 2029**

WAVE 3: CHROME ROOT POLICY

RECURRING, UNDATED

WAVE 4: POST-QUANTUM TLS

HORIZON

**EACH WAVE = A NEW SIGNED
RULE BUNDLE. SAME ENGINE.**



Ed25519



Five vendors. One model. A deterministic verdict.

AudioCodes, Cisco CUBE, Ribbon, Oracle Acme, and Metaswitch Perimeta configs are normalized to one vendor-neutral model and validated across eight domains. Same engine, by construction. Where a format cannot prove a fact, the engine stays silent and says verify out-of-band. It refuses to guess.

5

vendor parsers

8

validation domains

172

tests in CI

1

runtime dependency

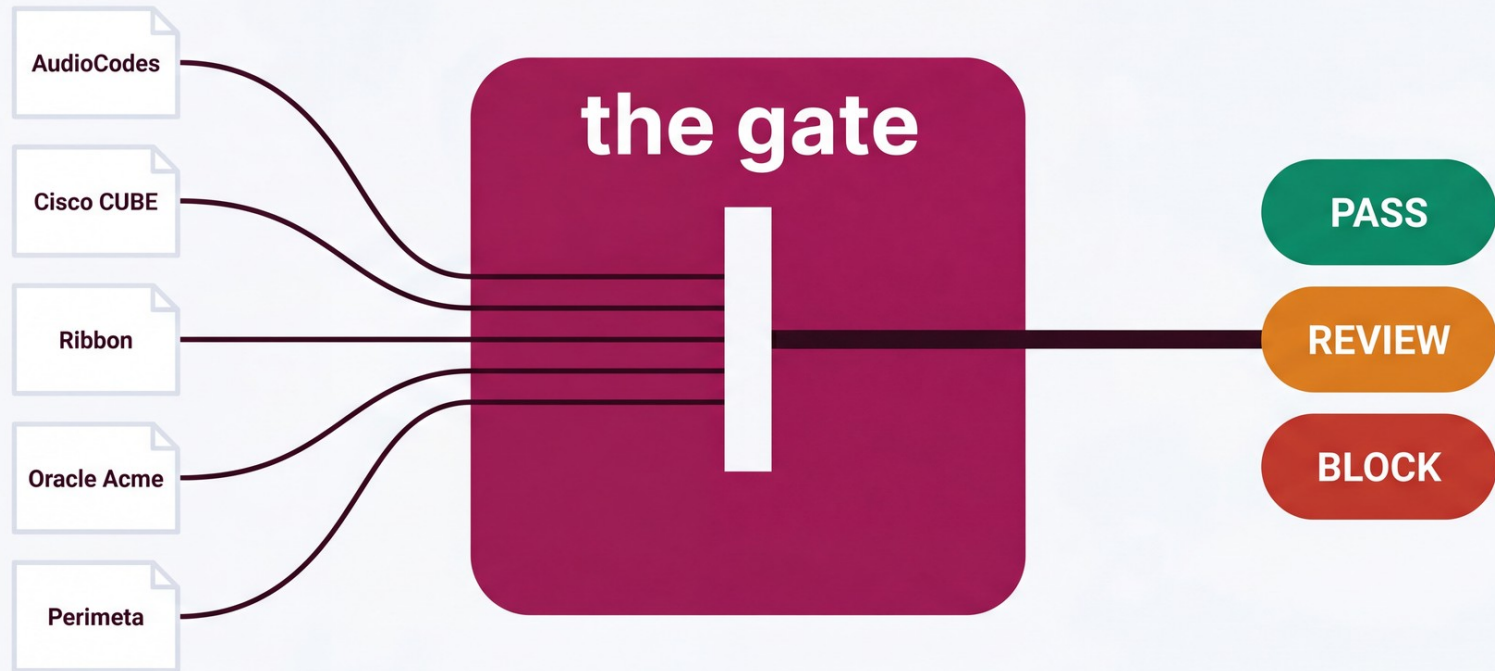
Every config gets one of three verdicts, and the same audit artifact from every run path: results/<sbcs>/<timestamp>.json, stamped with the ruleset it ran against.

PASS

REVIEW

BLOCK

validate · simulate · explain · diff · fleet · serve · report
· probe · demo



before the change window



Eight validation domains

A

Syntax / semantic

Malformed config, dangling references

B

Interop

TLS transport, OPTIONS keep-alive, normalization

C

TLS / CA (the 2026 wedge)

Missing Microsoft roots, mTLS, SRTP, ECU, chain validity

D

NAT / media

Private IP in SDP, missing symmetric RTP: one-way audio

E

Codec

Cross-leg overlap, forced transcode, DTMF

F

Topology leak

Private-IP leakage on the signaling plane

G

Routing / classification

404 and unclassified Teams routing

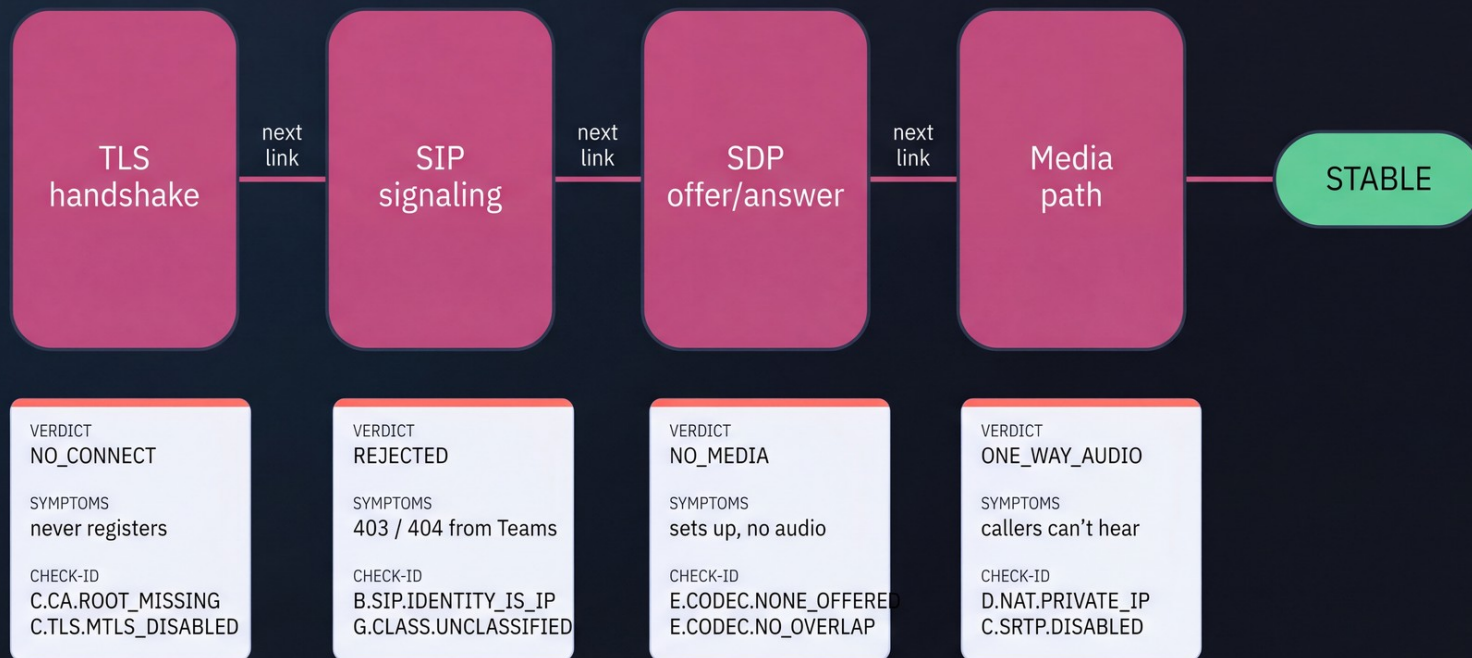
S

Security

ACL default-deny, broad CIDR, rule shadowing

Domain C is grounded against Microsoft Learn: the seven roots, the TLS floor, the cipher allowlist, and the SRTP suite reconciled as exact matches.

 first broken link





Where does your data go? Nowhere.

INBOUND

One channel: a signed rule bundle. Nothing else. Ed25519-signed, verified before use against a key pinned in source. Freshness floor refuses rollback. 8 MB cap. Configs never travel this channel.

OUTBOUND, default

Nothing. No telemetry, no call-home, no update check. Documented production mode: docker run with network none.

OUTBOUND, opt-in

Anonymized findings, only behind an explicit flag. Check IDs, severities, vendor family, ruleset version, salted org token. Never config text, FQDNs, CN/SAN, IPs, or paths. The exclusion is a 32-line module your reviewer can read live.

RUNTIME

Air-gapped and unprivileged. Non-root (uid 10001). Never touches the SBC: reads config exports only. No CLI sessions, no SNMP, no probes.

DATA-FLOW CONTRACT · SBC-AUTOOPS VALIDATOR

INBOUND one channel	A signed rule bundle. Nothing else. Versioned, Ed25519-signed, verified before use against a publisher key pinned in source (<code>rules/client.py</code>) and deliberately not environment-overridable. A compiled-in freshness floor refuses rollback to stale bundles; inbound reads are capped at 8 MB. Configs never travel this channel. Verify a bundle yourself, out-of-band: <code>python -m sbc_validator.tools.fetch_ruleset</code> checks the signature before writing a byte; a tampered bundle is rejected, not cached.
OUTBOUND default	Nothing. No telemetry, no call-home, no update check. The documented production mode is <code>docker run --network none</code> ; zero outbound network is required to run.
OUTBOUND opt-in only	Anonymized findings, only if you pass <code>--share-anon</code>. The payload (<code>schema sbc-anon/1</code>) carries check IDs, severities, the vendor family, the ruleset version, and a salted non-reversible org token. Excluded by construction: config text, locators, FQDNs, CN/SAN, IP addresses, file paths, free-text details.
RUNTIME	Air-gapped and unprivileged. The container drops to a non-root user (uid 10001). The engine is ~4,700 lines across 29 modules with one runtime dependency (<code>cryptography</code>); the pcap reader, rule client and dashboard server are hand-rolled on the standard library.

contract as of 2026-06-11 · engine v0.16.4 · every row maps to source: `rules/client.py` · `report/anonymize.py` · `Dockerfile`



Drive it from an agent

The same air-gapped engine, now exposed to an AI agent through the SBC Inspector MCP - a zero-dependency JSON-RPC-over-stdio server. Same deterministic verdicts, same air gap, now agent-native.

11

tools

3

prompts

3

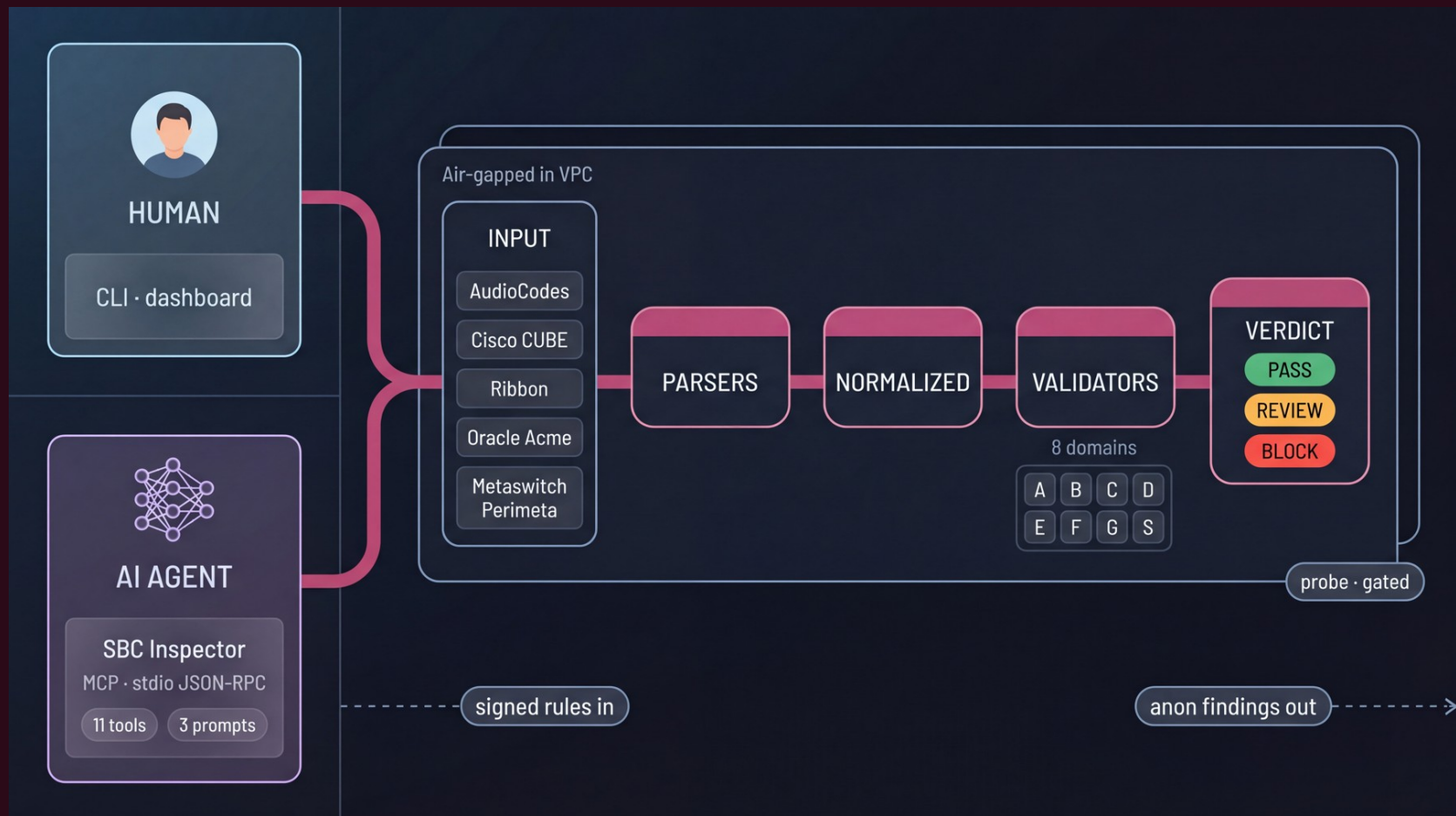
resources

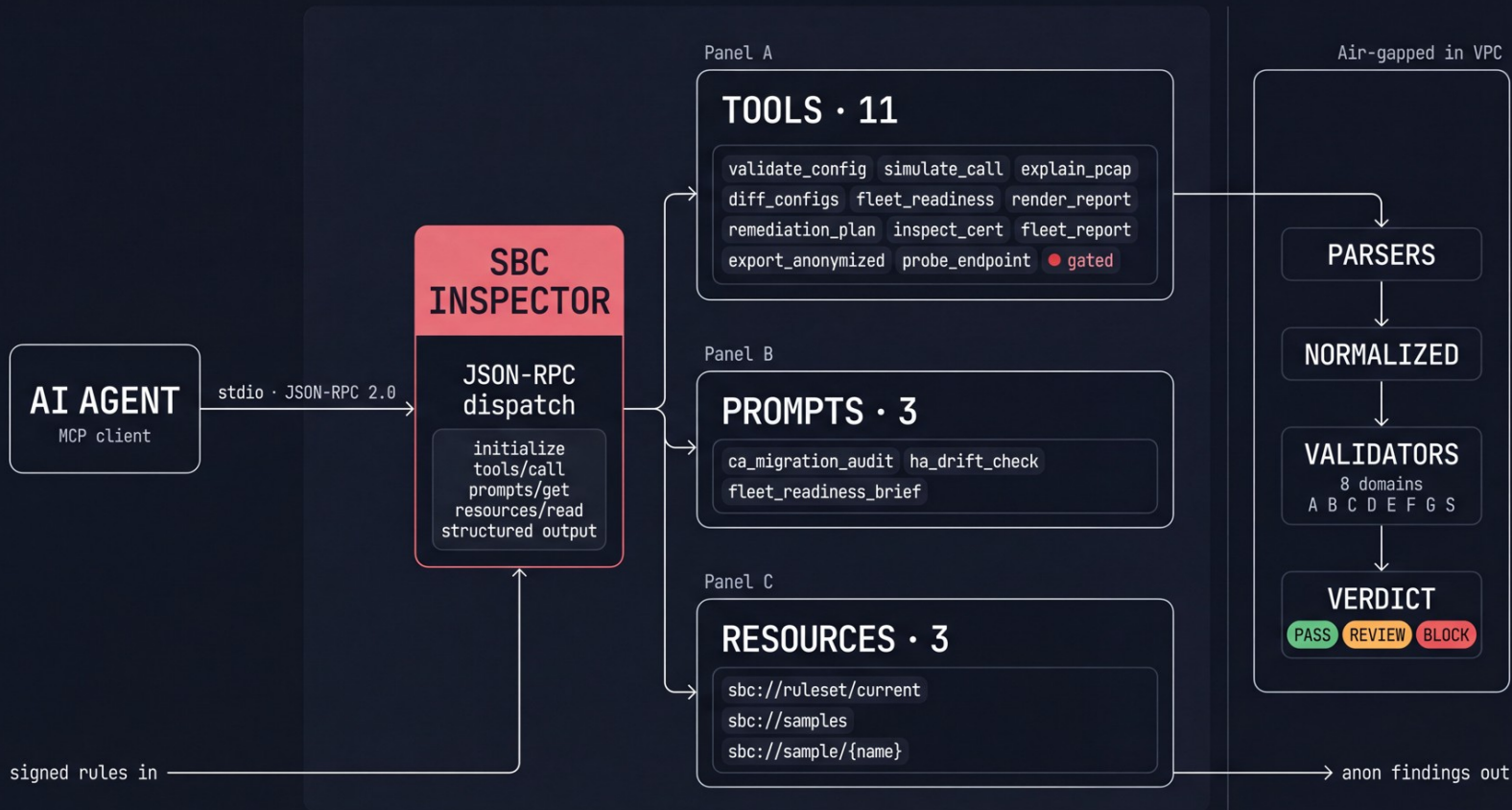
0

new dependencies

validate · simulate · explain · diff · fleet · report · remediate · inspect-cert · fleet-report · anonymize · probe

Every tool read-only and deterministic · offline by default (probe gated) · typed structured output · MCP 2025-06-18, conformance checked in CI against the official reference inspector.







Everything else is single-vendor, or watches the call after it fails

Approach	What it does	Against the 2026 problem
Post-deployment AIOps	Inspects active SIP and RTP: jitter, latency, MOS	Blind. The failure drops TLS before signaling exists.
Vendor tools (OVOC, RAMP, EM)	Manage their own hardware only	Siloed. Real estates are multi-vendor; no single pane exists.
Ribbon LEAP (closest comparable)	AI that learns live call flows, generates test scripts	Ribbon-only. Our pitch: LEAP, but cross-vendor.
SBC-AutoOps	Validates any vendor's config before deploy, then predicts the call	Catches the missing root, the wrong ECU, the SRTP gap. From the config, before traffic, across the whole fleet.

Structural argument: a vendor is biased to its own hardware and cannot credibly build the layer that reads everyone's.

Why pre-deployment, why independent



Post-deployment AIOps

blind: cannot watch a call that is blocked before it starts

tools (OVOC, RAMP)

Single-vendor tools

siloed: one make only, blind across a mixed fleet

SBC-AutoOps

catches it from the config, before traffic, on any vendor



Verify, don't trust

The meeting we want with you is your security review. Every claim below is checkable in the source.



Build from source

SHA-256 integrity manifest, CycloneDX SBOM, pip-audit in CI on every push. Nothing pre-built to trust.



Signed everything

Releases cosign-signed (keyless Sigstore). Rule bundles Ed25519-signed, verified against a source-pinned key.



One dependency

The cryptography library, guarded by a ratchet test. pcap reader, rule client, and dashboard are stdlib.



Deterministic

Same config plus same ruleset always equals the same verdict. No model temperature in the verdict path.



Runs your way

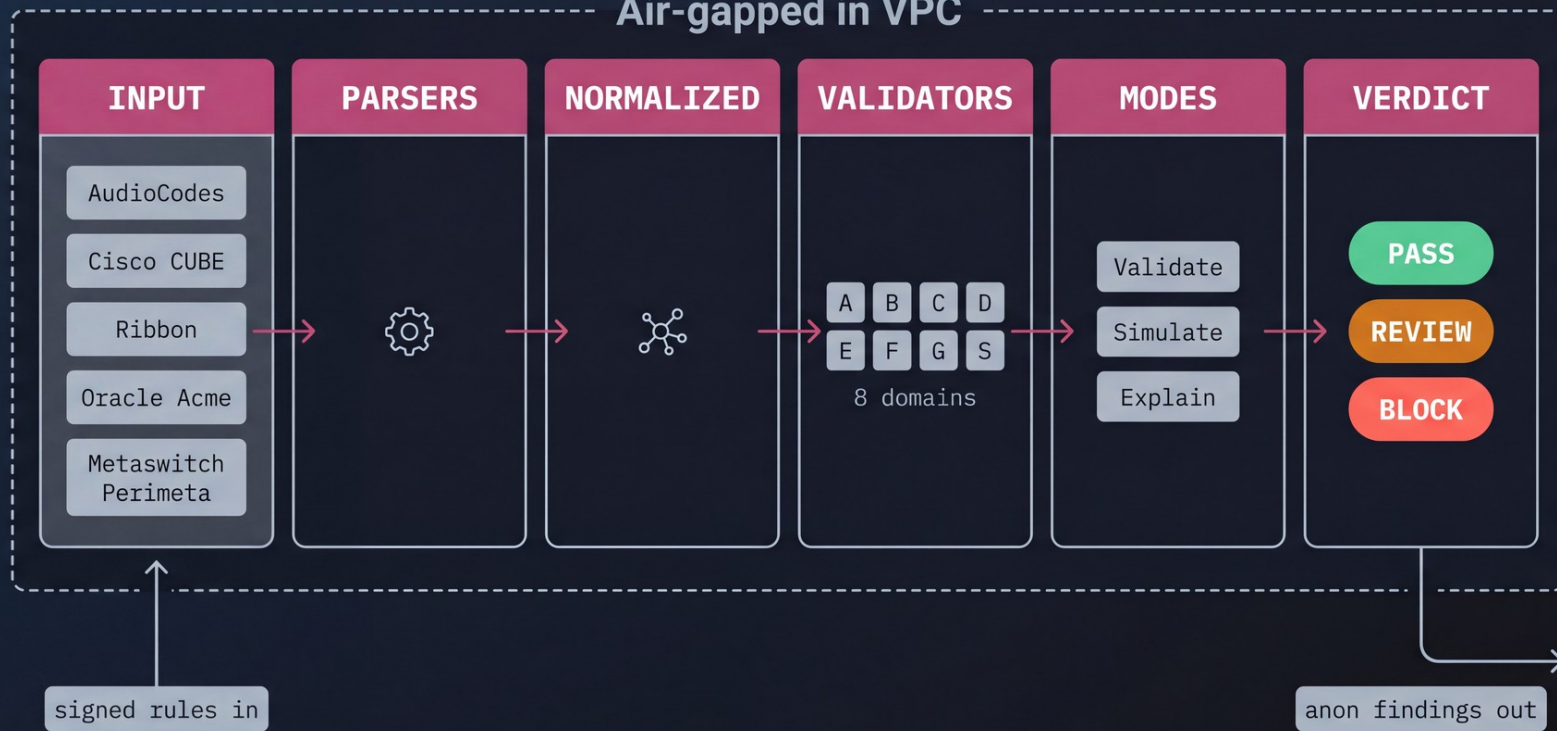
Docker is packaging, not a requirement: pip install on Python 3.10+, offline wheels, rootless Podman.



Disclosed before asked

An early dev-only signing key was rotated June 7 and scrubbed from history. We tell you because the posture must survive your diff audit.

Air-gapped in VPC





The design partnership

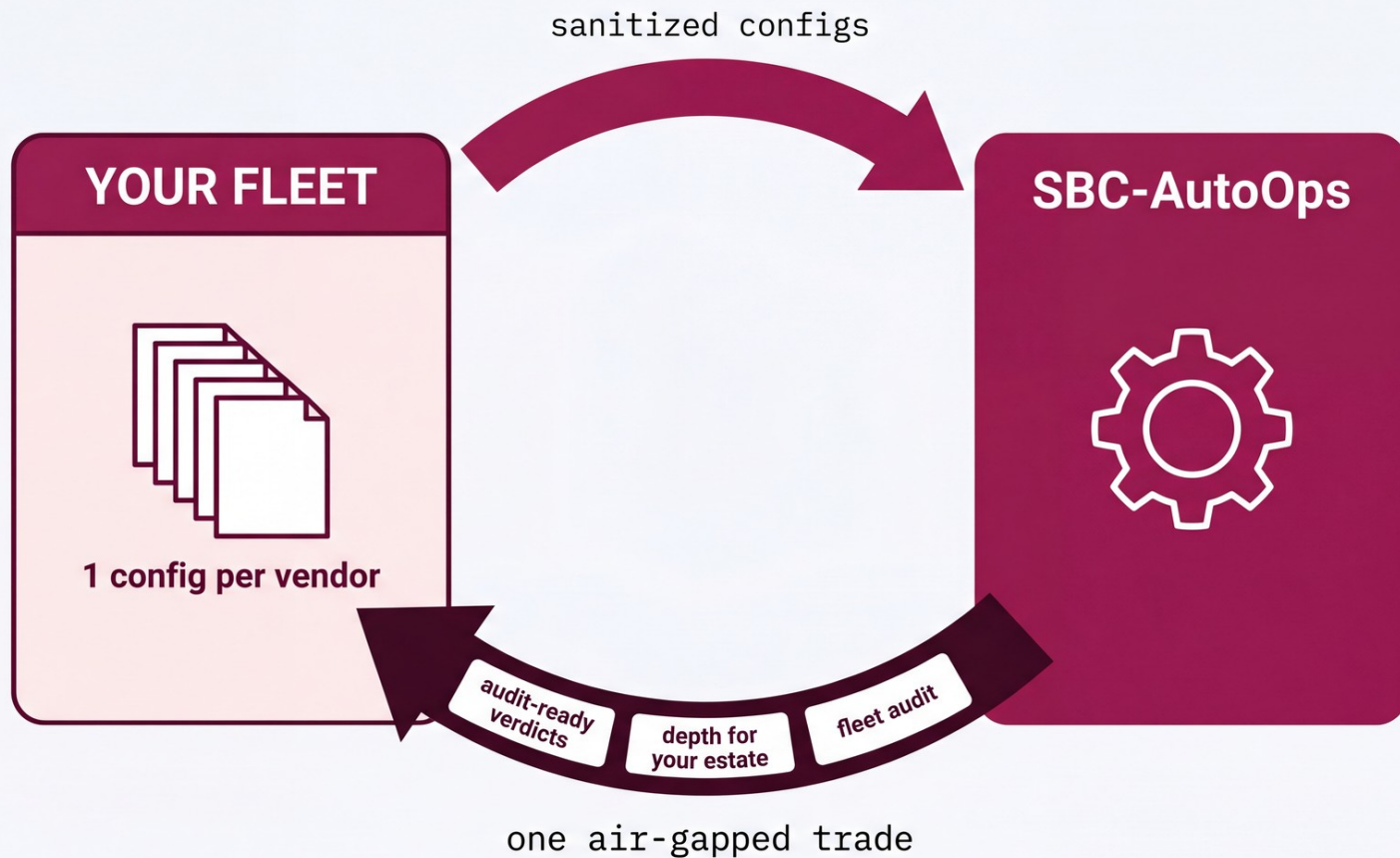
Two partners, 50 or more SBCs, mixed-vendor estate. The trade is one sentence: one sanitized config per vendor you run, for a 2026 readiness audit across your whole fleet, air-gapped in your environment.

What you get

- The full-fleet readiness audit, run by your team, on your hosts; reports stay on your disk
- Depth built against your platforms first: each config converts gated domains to firing, your fleet view first
- A deterministic pre-deploy gate for your CI and change process, with version-stamped evidence your auditors can cite
- Both founders, direct, through the pilot; first input on the roadmap

What we ask, and how it runs

- One real, sanitized config per vendor. A per-vendor checklist defines exactly what to export and strip; your security team approves it
- Mutual NDA first, then the checklist, then the pilot, air-gapped
- Candid feedback on verdicts against your ground truth
- Honest answer to the obvious question: you would be in the first cohort. That is the deal, stated plainly



The next step is your security review.

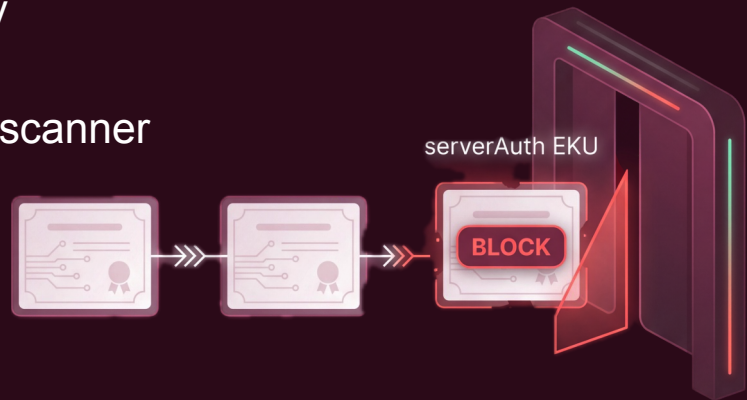
A 30-minute working session: we walk the data-flow contract first, run the live demo second, and leave you the source to verify. If the review holds, the pilot starts with an NDA and a sanitization checklist, not a contract.

Pre-read sbcvalidator.metaventionsai.com/#security

Free edge check sbcvalidator.metaventionsai.com/scanner

Contact dicoangelo@metaventionsai.com

METAVENTIONS AI · Dico Angelo and Philip Drammeh



1



Mutual NDA



2



**Sanitization
checklist**

your security reviews



3



**Air-gapped
pilot**

in your environment



4



**Depth
conversion**

your vendors first

days, not quarters