SBC-AutoOps

Design Partner Briefing

The independent truth layer for real-time voice.

Pre-deployment validation for Session Border Controllers, across five vendors, run air-gapped in your environment.

METAVENTIONS AI

Dico Angelo and Philip Drammeh, co-founders

June 2026 | Product version 0.16.4 | Partner edition

sbcvalidator.metaventionsai.com

Contents

1. Executive Summary

SBC-AutoOps is a pre-deployment validator for Session Border Controllers, the fragile, multi-vendor gateways every business call crosses on its way between an enterprise network and the carrier. The product reads any of five vendors' configuration exports, normalizes them to one model, validates eight domains, and returns a deterministic PASS, REVIEW, or BLOCK verdict before a change window, not after an outage. It runs local-first and fully air-gapped: zero outbound network is required, and raw configurations never leave your environment.

The trigger is a dated, cross-vendor forcing event. Microsoft's 2026 certificate-authority migration for Teams Direct Routing has already passed its hard root-CA deadline (March 31), and the serverAuth-EKU enforcement lands this month, June 2026. An SBC that misses it does not degrade gracefully: the TLS handshake fails, calls stop in both directions, and nothing in Teams points at the certificate. The deeper point is that this is the cadence, not the climax. Public TLS certificate lifetimes compress on a fixed schedule from 398 days today to 200 in March 2026, 100 in March 2027, and 47 by 2029, roughly eight renewals a year per SBC. Manual validation of those change windows stops scaling well before 47 days.

The product is built, tested, and live: version 0.16.4, 172 automated tests in CI across three Python versions, cosign-signed releases, one runtime dependency, Ed25519-signed rule bundles verified against a source-pinned key. The first design-partner cohort is forming now: two partners, and the trade is stated plainly in Section 11 and Section 14. As a design partner you would be among the first organizations to run this on a real fleet, and the document says so honestly wherever it matters.

At a glance

Company	METAVENTIONS AI. Co-founders Dico Angelo (product, engineering, security architecture) and Philip Drammeh (telecom architecture, ex-Microsoft Teams Direct Routing)
Product	Pre-deployment SBC config validator: five vendor parsers on one normalized model, eight validation domains, deterministic PASS / REVIEW / BLOCK verdicts
Status	v0.16.4, 172 tests in CI, cosign-signed releases, live at sbcvalidator.metaventionsai.com. Air-gapped: zero outbound network required
Why now	June 2026 serverAuth-EKU enforcement, then cert lifetimes compressing 398 to 200 to 100 to 47 days by 2029. The cadence, not the climax
The offer	Two design partners (50+ SBCs). One sanitized config per vendor, traded for a full-fleet 2026 readiness audit run in your environment

2. Company and Team

METAVENTIONS AI is the company behind SBC-AutoOps. Two co-founders, complementary lanes:

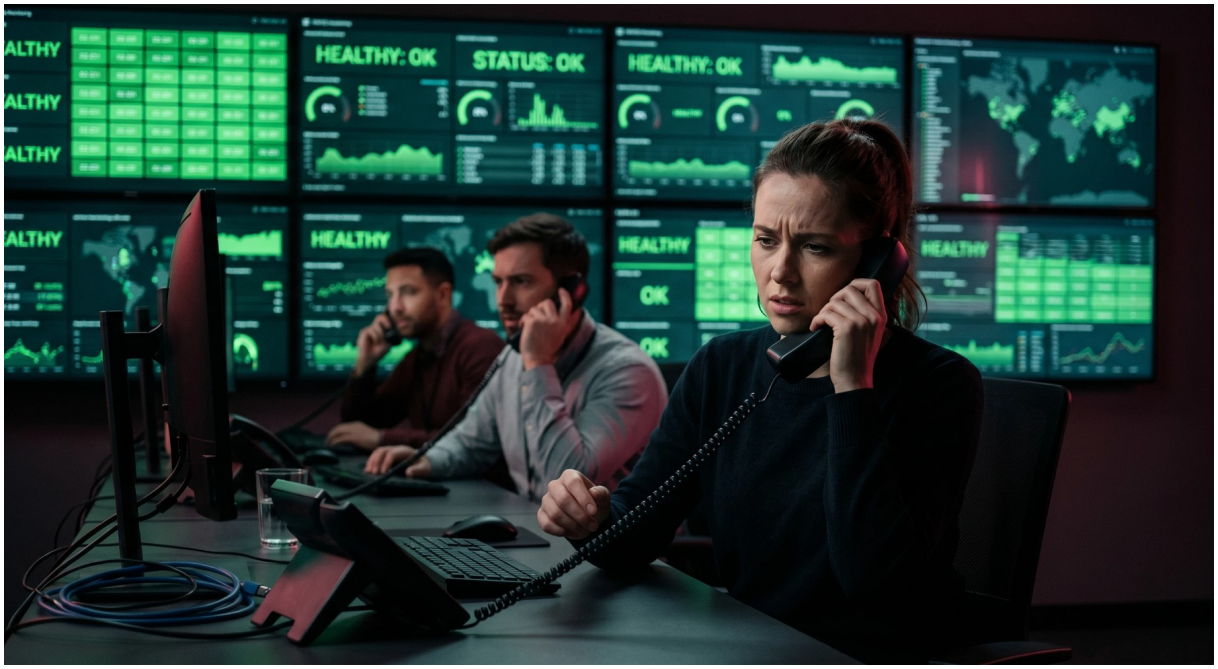
- **Dico Angelo.** Co-founder. AI builder and systems architect. Owns the product engineering, the validation engine, the security architecture, and the live surfaces.
- **Philip Drammeh.** Co-founder. Telecom architecture, formerly Microsoft Teams Direct Routing specialist lead. Owns telecom domain expertise, vendor semantics review, and practitioner relationships.

Operating posture

The company runs deliberately lean and builds only what a partner needs. The engine ships with one runtime dependency (the cryptography library), guarded by a ratchet test in CI. There is no cloud-hosted dashboard, no telemetry aggregator, and no multi-tenant control plane: results stay on your disk, and the architecture is structured so that has to remain true.

3. The Problem

Every business call crosses a Session Border Controller. One misconfiguration and calls fail silently: one-way audio, dead trunks, hours of repair, and the operator hears about it from users, not from a dashboard. Engineers call the discovery mechanism the scream test.



Why the problem is structural

- **The Layer 5 blind spot.** Layer 3 and 4 validation is stateless formatting. Layer 5 is a live, bidirectional, negotiated session. A mismatched SDP payload profile lets a call connect but destroys the audio; a one-character error in a header-rewrite rule kills a trunk silently. Generic network tools cannot reason about this.
- **Multi-vendor reality.** Enterprises accumulate AudioCodes, Cisco, Oracle, Ribbon, and Metaswitch estates through mergers and history. No tool spans them; engineers translate logic across vendor silos by hand.
- **Slow, expensive failure.** Mean time to repair runs one to four-plus days per misconfiguration, and the majority of network outages trace back to configuration and change management rather than hardware.
- **No pre-deployment check exists.** Vendor tools manage their own hardware only. AIOps products watch live calls after deployment. Nothing independently validates the configuration before it ships.

The failure is already happening in production

Three named incidents from the 2026 Microsoft CA migration, tracked under Microsoft Message Center MC1235747:

- **Voipcloud.** P1 outage across thousands of users. Microsoft stopped accepting their calls because a temporally valid but untrusted root was presented at the mutual-TLS handshake.

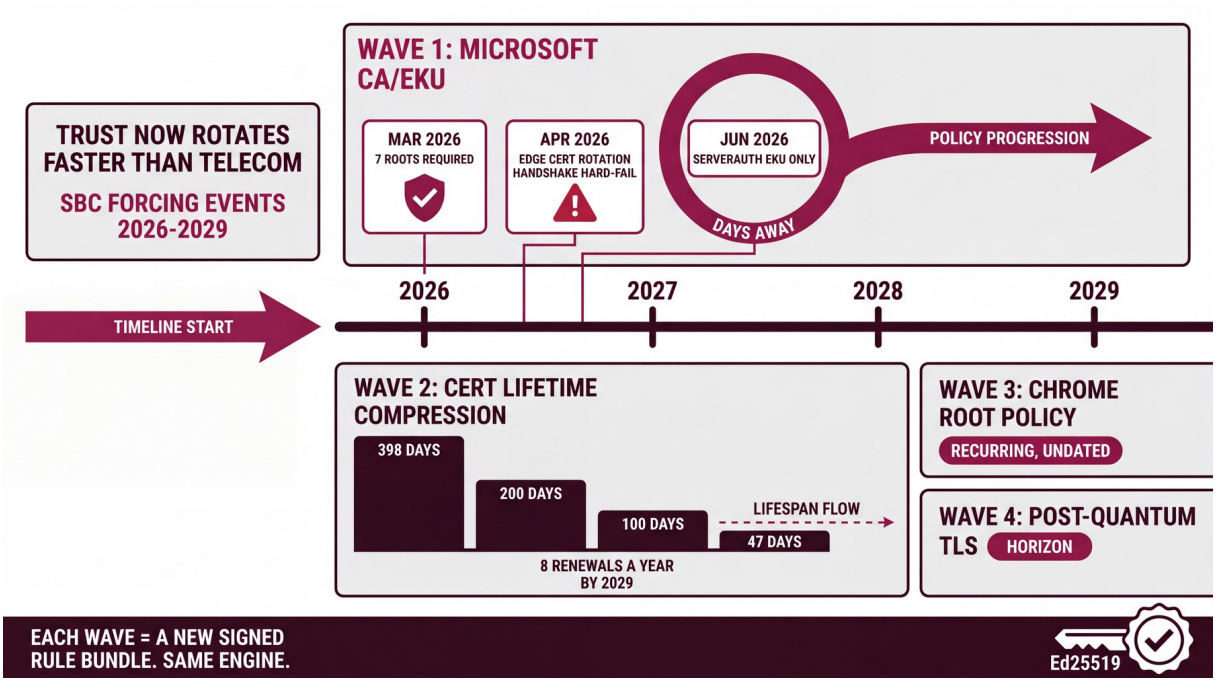
- **CallTower.** Silent blackhole. Unpatched SBCs stopped sending SIP OPTIONS pings; Microsoft auto-deactivated the Direct Routing domain and blackholed every voice route with no SIP 503 to warn anyone.
- **FusionConnect.** One-way audio. Signaling succeeded but media relays presented new DigiCert certificates and outbound audio silently dropped on an SRTP mismatch. Honest caveat: this particular failure originated on Microsoft's media-relay side and would not have been catchable from the customer's configuration. It is included because it illustrates the failure class, not because the product would have prevented it.

One sentence

You cannot monitor a call that is cryptographically blocked from ever starting. That is why the category must be pre-deployment, and why post-deployment observability cannot fill it.

4. The Forcing Events: a Cadence, Not a Climax

Teams Direct Routing runs entirely on mutual TLS. In 2026 Microsoft retired the legacy roots, moved to new DigiCert and Microsoft 2017 root CAs, and required that the SBC certificate carry the Server Authentication EKU. The 2026 enforcement timeline:



Date	Event
Feb 16, 2026	Microsoft test endpoint live for pre-deployment validation
Feb 28, 2026	Industry remediation deadline: firmware and root imports done
Mar 31, 2026	Hard deadline: legacy roots rejected outright
April 2026	Microsoft rolls new certificates onto the primary SIP proxy; unpatched peers drop
June 2026	Public CAs issue serverAuth-only certificates; the SBC leaf must carry the Server Authentication EKU. Enforcement lands this month.

Behind the deadline sits the durable schedule. Per the CA/Browser Forum ballot SC-081v3, public TLS certificate maximum lifetimes compress on a fixed, third-party-verifiable calendar:

Effective	Max lifetime	Operational meaning per SBC
Today	398 days	Roughly annual renewal
March 2026	200 days	Two renewals a year
March 2027	100 days	Quarterly-plus renewal
March 2029	47 days	Roughly eight renewals a year

Direct Routing SBC certificates are public-CA certificates, so every fleet is hit, every cycle, across whatever mix of vendors it runs. Behind the lifetime compression sit further waves on the same trajectory: recurring Chrome root-policy changes, and the post-quantum TLS migration (NIST ML-KEM and ML-DSA are finalized; SIP infrastructure has not started). Each wave becomes a new signed rule bundle on the same engine. 2026 is episode one; the trust regime keeps moving, and this is the layer that keeps a voice fleet in sync with it.

5. Market

Size and context

Figure	What it measures
$1.1B to $2.4B	SBC market, 2025 to 2035, roughly 8 percent CAGR
$100B+	UCaaS market, growing 15 to 20 percent a year
20M+ users	On Microsoft Teams Direct Routing for PSTN calling
50 to 75 percent	Telephony savings of Direct Routing versus Microsoft Calling Plans; the entire saving rides on the SBC being right

Who this is built for

- **Tier-1 carriers and SBCaaS.** Multitenant cloud SBC fleets where one certificate mismatch cascades into simultaneous outages across hundreds of enterprise tenants.
- **Cloud-voice MSPs and operator-connect providers.** API-driven virtual-SBC fleets behind white-label voice; a failure severs downstream channel revenue.
- **Systems integrators.** Own the multi-vendor PBX-to-Teams migration and the responsibility that hybrid estates survive the cryptographic shift.
- **Enterprises with 50+ mixed-vendor SBCs.** Direct Routing or BYOC estates with no cross-vendor, pre-deployment check before each change window.

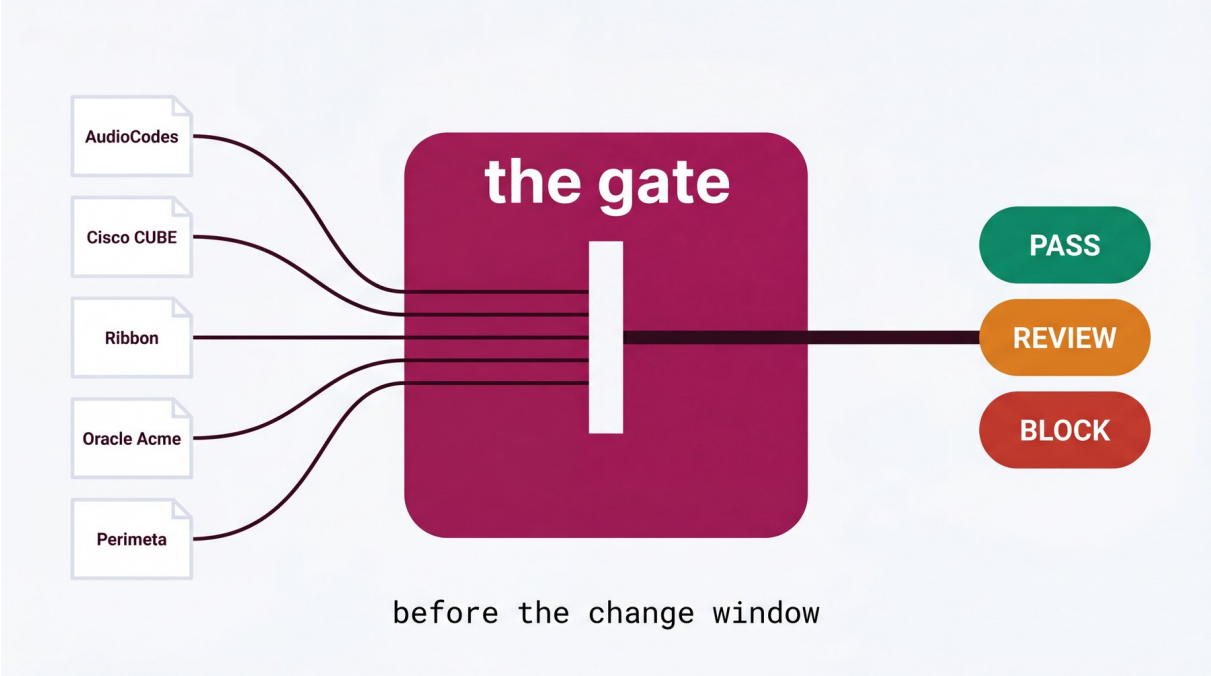
The regulated segment: who cannot afford to get this wrong

Organizations that run their own SBCs usually do so because compliance keeps them off managed cloud calling, which means they own the certificate problem directly and downtime is regulatory or life-safety, not inconvenience:

- **Financial (MiFID II, FINRA):** trader-voice estates where an unrecorded call is a fine, on HA Oracle clusters and mixed AudioCodes plus Cisco CUBE migrations.
- **Health and public safety (HIPAA, CJIS):** police routing, clinics, and emergency triage, where survivable branch appliances must trust the new roots.
- **Defense and manufacturing:** JITC-certified Ribbon with FIPS-140-2; HA AudioCodes pairs running Local Media Optimization across global factory floors.

6. The Product: What Ships Today

Version 0.16.4. 172 automated tests in CI on Python 3.10 through 3.12. Approximately 4,700 lines of code across 29 modules at the last architecture audit, with one runtime dependency (cryptography), enforced by a ratchet test. The pcap reader, HTTPS rule client, and dashboard server are hand-rolled on the standard library, so the tool audits clean and runs fully air-gapped: docker run with network none is the documented production mode.



Five vendors, one model

Every layer speaks one vendor-neutral shape, the NormalizedConfig. Parsers produce it, validators consume it, and they never import each other, so five vendors on one engine is true by construction, not by claim.

Vendor	Format	Depth today
AudioCodes	.ini (table and simple)	Full: TLS, certificate, SRTP, codec, NAT, routing, security. Hardened against a real Mediant export.
Cisco CUBE	IOS-XE	TLS, certificate, SRTP, codec, NAT. Routing and security activate once validated against a real config (the design-partner path).
Ribbon	SBC Core set-config	TLS, certificate, SRTP, codec, NAT. Routing and security activate once validated against a real config.
Oracle Acme	Acme Packet ACLI	TLS, certificate, SRTP, codec, NAT. Routing and security activate once validated against a real config.

Metaswitch Perimeta	Adjacency CLI export	Interop and transport posture. Trust store, certificate, and codec live outside the export, so those domains stay silent and the report says verify out-of-band.

The engine refuses to guess. Where a configuration format cannot prove a fact, the verdict says verify out-of-band rather than inventing one. The gating is enforced in the type system (an unknown stays silent), not in documentation. A wrong verdict is the one thing a pre-deployment control cannot afford; silence beats it.

Eight validation domains

	Domain	Catches
A	Syntax / semantic	Malformed configuration, dangling references
B	Interop	TLS transport, OPTIONS keep-alive, header normalization, IP-as-identity (which Teams rejects with 403)
C	TLS / CA (the 2026 wedge)	Missing Microsoft roots, mTLS off, SRTP off, EKU, chain validity, wildcard-aware CN/SAN matching
D	NAT / media	Private IP in SDP, missing symmetric RTP, the one-way-audio class
E	Codec	Cross-leg overlap, forced transcode and DSP exhaustion, DTMF, wideband downgrade
F	Topology leak	Private-IP leakage on the signaling plane
G	Routing / classification	404 and unclassified Teams routing, both-direction routes
S	Security / access control	ACL default-deny, broad CIDR, rule shadowing, media-plane ACL omission

The Domain C ruleset is authoritatively grounded against Microsoft Learn: the seven Microsoft roots, the TLS floor, the cipher allowlist, the SRTP suite, and the test endpoint were reconciled as exact matches against Microsoft's published requirements.

Modes and commands

Three diagnostic modes that speak the call flow, not config lint: validate (static audit producing a risk score and a PASS, REVIEW, or BLOCK verdict), simulate (deterministic offline prediction of how far a call gets, naming the user-visible symptom and rendering a SIP ladder truncated at the failure), and explain (packet-capture post-mortem reconstructing the SIP ladder and detecting one-way audio from RTP direction).

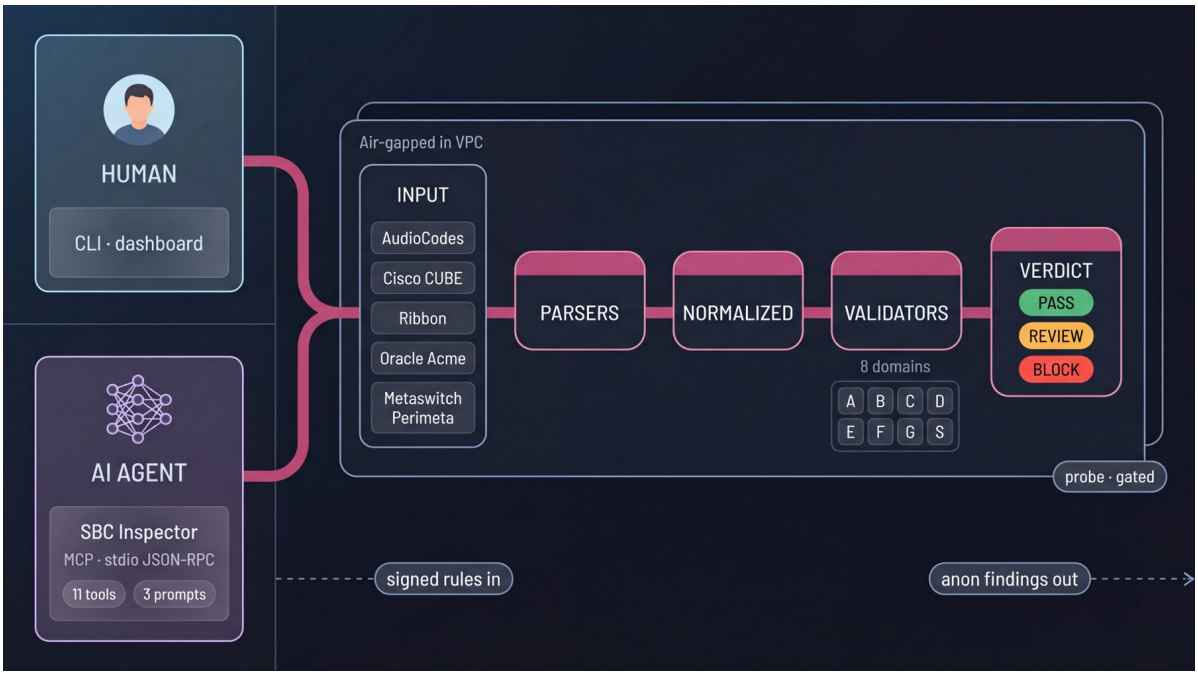
Nine CLI commands: validate, simulate, explain, diff (HA drift between active and standby), fleet (readiness rollup), serve (local dashboard), report (executive and compliance mapping, including MiFID II), probe (outside-in live TLS handshake graded against the ruleset), and demo (the whole showcase in one packaged command). The validate command returns a non-zero

exit code, so it drops straight into CI and fails a build before a non-compliant configuration reaches the change window.

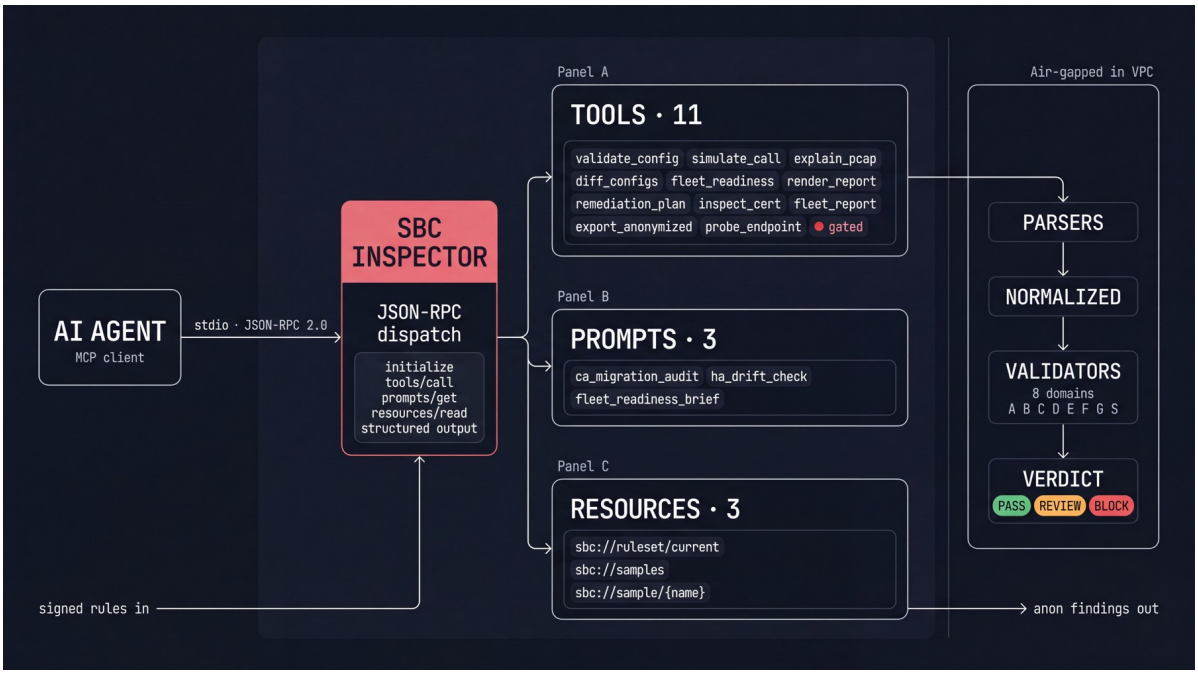
Every run path produces the same audit artifact, results/<sbc>/<timestamp>.json, version-stamped with the ruleset it was validated against, which is what makes the verdict usable as change-management evidence.

The agent surface: SBC Inspector MCP

The same engine is also an MCP server (sbc-validator mcp): a zero-dependency JSON-RPC-over-stdio interface that lets an AI agent or a CI pipeline drive the validator directly. It exposes eleven read-only tools (validate, simulate, explain, diff, fleet readiness, render report, remediation plan, cert inspect, fleet report, anonymized export, and a gated network probe), three guided prompts (CA-migration audit, HA-drift check, fleet brief), and read-only resources. Every tool is deterministic and offline by default; probe_endpoint is the only tool that touches the network and stays off unless explicitly enabled. There is no new SDK and no new dependency, so the one-runtime-dependency promise holds, and MCP conformance is verified in CI against the official reference inspector. A human runs the CLI; an agent calls the MCP; both reach the same validators behind the same air gap.



Two surfaces, one engine: a human CLI and dashboard, and an AI agent over the SBC Inspector MCP, both driving the same air-gapped engine.



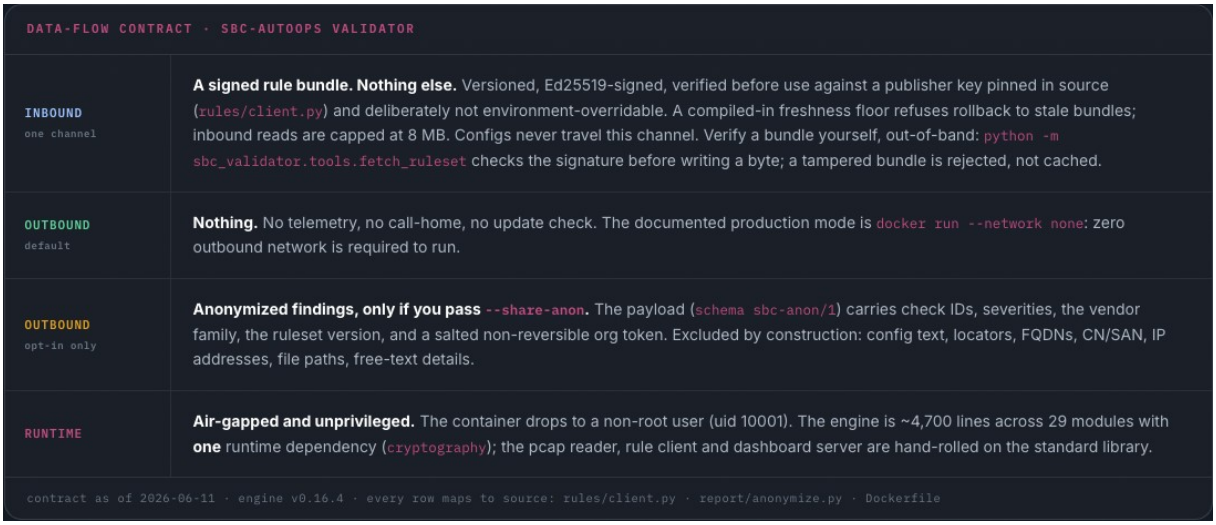
The MCP backend in detail: eleven tools, three prompts and three resources over stdio JSON-RPC, all calling the air-gapped validation core.

Live surfaces you can verify right now

- **sbcvalidator.metaventionsai.com:** the business case. The #security section is the data-flow contract panel, written to be lifted into a security review. The page itself is a demo of the trust model: zero external runtime requests, self-hosted fonts, verifiable in any browser's network tab.
- **/scanner:** the free outside-in edge grader. Enter an SBC FQDN; it runs a real TLS handshake against the public edge and grades it against Microsoft's 2026 requirements. No login, no config upload, edge-only, SSRF-guarded, rate-limited; stores grades and check IDs in aggregate only, never hostnames.
- **/dashboard/:** the seven-view local console demo: fleet, scanner, walkthrough, findings, rule bundles, reports, and setup guide.

7. Security and Trust Architecture

The security model is built for your security review, and the review is welcome. The data-flow contract, stated precisely:



Direction	Contract
Inbound (one channel)	A signed rule bundle. Nothing else. Versioned, Ed25519-signed, verified before use against a publisher key pinned in source and deliberately not environment-overridable. A compiled-in freshness floor refuses rollback to stale bundles; inbound reads are capped at 8 MB. Configurations never travel this channel. A bundle can be verified out-of-band before a byte is written; a tampered bundle is rejected, not cached.
Outbound (default)	Nothing. No telemetry, no call-home, no update check. The documented production mode is docker run with network none: zero outbound network is required to run.
Outbound (opt-in only)	Anonymized findings, only behind an explicit flag. The payload (schema sbc-anon/1) carries check IDs, severities, vendor family, ruleset version, and a salted non-reversible org token. Excluded by construction: configuration text, locators, FQDNs, CN/SAN, IP addresses, file paths, free-text details. The exclusion lives in a 32-line module your reviewer can read live.
Runtime	Air-gapped and unprivileged. The container drops to a non-root user (uid 10001). The engine never needs privileges and never touches the SBC itself: it reads configuration exports, with no CLI sessions to the device, no SNMP, no probes.

Verify, don't trust

- Build from source on your side: a SHA-256 integrity manifest verifies the tree, and a runtime CycloneDX SBOM ships in the repository. pip-audit runs in CI on every push.
- Published images are cosign-signed (keyless Sigstore). SLSA provenance and an HSM for the publisher key are committed pre-GA roadmap.

- Docker is packaging, not a requirement: plain pip install on any Python 3.10+ host, fully-offline wheel installs, and rootless Podman all run identically.
- Deterministic by design: the same configuration and the same ruleset always produce the same verdict. There is no model temperature in the verdict path. AI features, where they arrive, live in the explanation layer, never the verdict path.

Honest disclosure, stated before you ask

A development-only throwaway signing key was committed early in the repository's life. It was rotated to an offline keypair on June 7, 2026, and the git history was rewritten so the dead key is gone from every commit. The pinned verifier key never had its private half in the repository. We disclose this proactively because the trust posture only works if it survives a diff audit, and yours is welcome.

8. Competition

Approach	What it does	Against the 2026 problem
Post-deployment AIOps (e.g. Ribbon Analytics)	Inspects active SIP and RTP for jitter, latency, MOS	Blind. The failure drops TLS before signaling exists; you cannot monitor a call that is cryptographically blocked from ever starting.
Vendor tools (AudioCodes OVOC/ARM, Ribbon RAMP, Oracle EM)	Manage their own hardware only; the AudioCodes wizard resets unmanaged parameters to factory defaults	Siloed. Reality is multi-vendor: LoopUp runs Ribbon plus AudioCodes; Societe Generale runs Oracle plus AudioCodes plus Ribbon. No single pane exists.
Ribbon LEAP	AI that learns live call flows and generates test scripts; the closest comparable	Ribbon-only. Our pitch is LEAP but cross-vendor.
SBC-AutoOps	Validates any vendor's configuration before deploy, then predicts the call	Catches it: the missing root, the wrong EKU, the SRTP gap, from the configuration, before traffic, across the whole mixed fleet.

The whitespace is an independent, multi-vendor, pre-deployment Layer 5 validator. A structural argument reinforces the gap: a vendor is biased to its own hardware and cannot credibly build the layer that reads everyone's.

9. Why This Stays Hard to Copy

- **Vendor-agnostic Layer 5.** True SIP and SDP normalization across mixed estates, hard to build and harder to keep correct. The deepest version of it, what a missing configuration line means per vendor and per firmware, can only be learned from real production configurations, which is exactly what the design partnership builds, starting with your platforms.
- **Security-first, local-first.** Air-gapped local execution removes the raw-config-exfiltration objection that blocks comparable tools in security review, converting the review from an adoption blocker into an adoption driver. The correctness discipline, refusing to guess, is part of the same trust posture.
- **A compounding benchmark, without your data.** To be unambiguous for your security team: the raw configuration never leaves. What can leave, only if you opt in, is anonymized findings (check IDs and severities, never configuration text or IPs). Those aggregate into a cross-vendor readiness benchmark no single-vendor tool can assemble. The pattern compounds; your configuration never does.

10. Business Model

The commercial model is a per-SBC subscription with an explicit data-consent framework. The security-first architecture doubles as the pricing argument: you are paying for a control that runs inside your trust boundary, on a fleet whose downtime is measured in regulatory exposure and severed channel revenue, against a renewal cadence that is contractually certain to accelerate.

Why this is a recurring control, not a one-off project

- The cert-lifetime schedule (Section 4) makes validation a permanent operating activity, roughly eight change windows a year per SBC by 2029, not a one-time migration.
- Each new trust-regime wave (root changes, EKU policies, post-quantum) ships as a new signed rule bundle on the same engine.
- For regulated estates, the gate can be named as the mandated pre-deploy step in an audited change-management procedure, with the retained, ruleset-stamped JSON verdict as the evidence record.

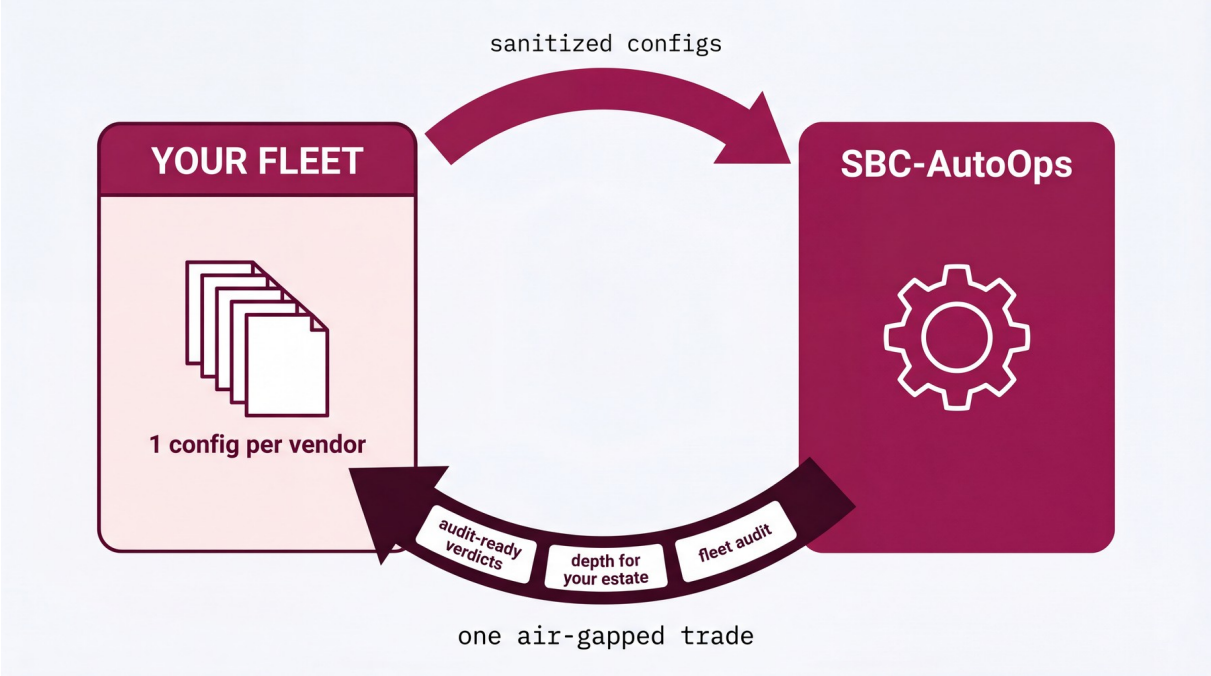
Compliance mapping (indicative, not legal advice)

Requirement family	Where the control lands
MiFID II call-recording continuity	Domain C blockers stop a change that would silently break recorded trader voice
Change-management evidence (SOX-style, FINRA)	The retained JSON verdict per change, ruleset-version-stamped
CJIS / HIPAA encrypted-media posture	Domains C and D findings: SRTP, media trust path, NAT exposure
Minimize-human-error mandates	The gate itself: scripted, deterministic, fails closed

Pricing is deliberately unset in this document. Design partnerships are traded for configurations, not cash, and paid pricing will be set against real fleet economics rather than published as a speculative rate card. We think that is the honest order of operations.

11. The Design Partnership

Two design partners: MSPs or enterprises running 50 or more SBCs across a mixed-vendor estate. The trade, verbatim from the live site: one real, sanitized configuration per vendor you run, in exchange for a 2026-migration readiness audit across your entire mixed-vendor fleet, run air-gapped inside your environment, before the next failure finds you.



What you get

- The full-fleet readiness audit, run by your team, on your hosts, with every report staying in a local results directory you own.
- Validation depth built against your exact platforms first: each sanitized configuration converts that vendor's routing and security domains from gated to firing, visible in your own fleet view before anywhere else.
- A deterministic pre-deploy gate that drops into your CI and change process, and a control your auditors can cite, with version-stamped evidence per change.
- Direct access to both founders during the pilot, and first input on the roadmap (Section 12).

What we ask

- One real, sanitized configuration per vendor in your fleet. A per-vendor sanitization checklist specifies exactly what to export and what to strip, so your security team controls precisely what is shared.
- Candid feedback on verdicts against your operational ground truth.

How onboarding runs

- Mutual NDA first.
- Then the configuration-sanitization checklist, reviewed by your security team before anything is exported.
- Pilot scope: the fleet readiness audit in your environment, air-gapped; the documented production mode requires zero outbound network.
- Depth conversion: gated domains activate for your vendors as each configuration is modeled, your fleet first.

Honest answer to the obvious question

Who else runs this? You would be a design partner in the first cohort. That is the deal, stated plainly: your configurations buy depth on your exact platforms, and you get the fleet audit and a control built around your estate.

12. Roadmap

Three horizons. Everything in the first ships today; everything after earns its way in the same manner the rest of the tool did, validated against real configurations, never guessed. Next and Vision are roadmap, not product, and are labeled that way everywhere they appear.

Shipped, running today

SBC Inspector MCP: the engine as an agent surface (11 tools, 3 prompts, read-only resources), reference-inspector-conformant, zero new dependencies.

- Local-first validator: five vendors normalized to one model; eight validation domains including the 2026 CA and EKU checks.
- Validate, simulate, and explain modes; HA-drift diff; fleet readiness reports; a CI/CD gate; the local dashboard; the executive and compliance report; the outside-in probe.
- Signed rule bundles, air-gapped Docker image, cosign-signed releases, 172 automated tests in CI.

Next, with design partners

- Routing and security checks live on Cisco, Ribbon, and Oracle; each needs exactly one real configuration.
- Per-config cipher-suite matching; the cross-tenant readiness benchmark; deeper CI/CD and change-management integration.
- Trust backbone: the signed-ruleset distribution service, the re-sign workflow for when Microsoft changes the CA list, and an HSM for the publisher key.

Vision, the lifecycle

- Continuous drift detection across the fleet; post-quantum readiness scoring.
- An LLM explainer for SIP ladders from packet captures, explicitly in the explanation layer and never the verdict path.
- Assisted remediation, applied by engineers, never auto-pushed to production.
- STIR/SHAKEN and Rich Call Data conformance auditing.

13. Honest Limits

This product's voice rule is simple: where the engine cannot prove a fact, it stays silent, and so does this document. The limits, stated plainly:

- The gate validates what a configuration can prove. It cannot verify that an arbitrary SIP header-rewrite regex is semantically correct; no static tool can. This boundary is stated, not hidden.
- Routing and security depth for Cisco, Ribbon, and Oracle stays silent until validated against a real configuration for that vendor. That is the design-partner ask, not a hidden gap.
- Metaswitch Perimeta is read from an adjacency-CLI export that does not carry the trust store, certificate, or codec policy inline, so those domains stay silent for it and the report says verify out-of-band.
- Validation depth is currently skewed to the Teams leg; carrier-leg coverage and SRTP-to-RTP interworking are thinner and named as such.
- It does not monitor a running SBC. Pre-deployment validation, call-flow simulation, capture post-mortems, HA and golden drift, and fleet rollups are what ship today.

Capabilities described as available in this document are implemented and covered by the 172-test CI suite. Anything that is roadmap is labeled roadmap.

14. The Ask

Two design partners: MSPs or enterprises running 50 or more SBCs across a mixed-vendor estate.

The trade: one real, sanitized configuration per vendor in your fleet. In exchange, a 2026-migration readiness audit across the entire fleet, run air-gapped inside your own environment, with every report staying on your own disk.

The next step is a 30-minute working session that doubles as your security review: we walk the data-flow contract first, run the live demo second, and leave you the source to verify. Contact: dicoangelo@metaventionsai.com, or the pre-read at sbcvalidator.metaventionsai.com/#security.

Every capability claim in this document is verifiable in the repository and the live surfaces. Figures reflect repository state at v0.16.4, June 11, 2026. Market figures are third-party estimates cited on the live business case.