CrossConnect
CONNECT. MONITOR. OPTIMIZE.
BY Cybr IQ

Platform Overview

An honest orientation to CrossConnect: what it is, how it is built, what it can do, and the trust posture under every answer it gives. Written for the technical leaders deciding whether it belongs in their network.

Audience: engineering & IT leadership, architecture, evaluation owners

Scope: the platform thesis, architecture, capability spread, and trust posture, not an install or security deep-dive

Posture: read-only collection · self-hosted · grounded advisory AI · confidence-scored facts

Document: platform overview, 21 June 2026

Contact: contact_us@cybriq.io

0 How to read this document

Read this first. It gives a technical leader enough to know what CrossConnect is, where it fits, and what it is honest about. It opens with the problem, then the shape of the solution, then the architecture and the range of what the platform does. The control-by-control detail a security review needs lives in the companion *Security & Architecture Reference*.

GA shipped & on by default CONFIGURABLE shipped, operator-enabled

DEPLOYMENT OPTION supported integration / source you turn on

1 The problem we solve

The network you documented is not the network you actually have. Switches get added. Firmware drifts. A contractor patches in a device nobody writes down. By audit time the spreadsheet is fiction, and the team that runs the network cannot prove what it runs, let alone explain it to the auditor, the project manager, or the executive who is paying for it.

The usual tools for this job take whatever an operator typed and trust it. They store intent and call it reality. They do not check the written record against the live network, they cannot answer a plain question and show the evidence behind the answer, and they leave a non-specialist waiting on a network engineer for a straight answer. CrossConnect starts from the opposite idea: **what you documented and what is actually out there are two separate facts. We keep both, compare them when you ask, and every answer can show its evidence.**

2 What CrossConnect is

CrossConnect is a network source-of-truth and operational-intelligence platform. It keeps one accurate, current picture of your network and lets anyone ask questions of it. It reads the live network read-only over SNMP and LLDP, the protocols switches already use to report inventory and neighbors, and, if you opt in, device configurations over SSH. From those reads it builds one model of devices, interfaces, IP space, VLANs, cabling, circuits, and routing, then keeps that model honest by holding it side by side with what the network actually broadcasts. On top of that model sit the operational lenses: lifecycle, compliance, capacity, segmentation, audio-visual (AV) assurance, and a conversational assistant that answers from the records and shows which ones it used. The whole team can use it, not only the network engineer.

It builds the inventory for you

Point it at the gear. It pulls models, interfaces, neighbors, serials, and running software, and tags every record with **when it was last seen**, so drift surfaces instead of hiding.

It tells you what is aging out

Hardware and software past their support dates, plus known security flaws (CVEs), across the fleet. It also checks whether a recommended upgrade will fit the box's memory and flash before you schedule it.

It makes the audit boring

Scores the network against common security frameworks (CIS, NIST CSF, and SOC 2) using evidence it already holds, is upfront about what it cannot prove, and exports a clean evidence pack.

It answers in plain language

Ask a question; it answers from your records, shows the records it used, and never executes a change. The non-specialist gets a straight answer.

> **What sets it apart, in one line.** Most tools store what you typed and trust it. CrossConnect keeps your records and the live network side by side, tells you the moment they stop matching, and backs every AI answer with a record it can point to, or says "I don't know."

3 The platform at a glance

The whole flow fits in one picture. It starts with read-only collection and ends in advisory output that cites its sources. The model in the middle is the single PostgreSQL system of record. Between raw observation and trusted truth sits a trust gate: the checkpoint that decides what is allowed to count as fact, which is what makes an answer trustworthy.

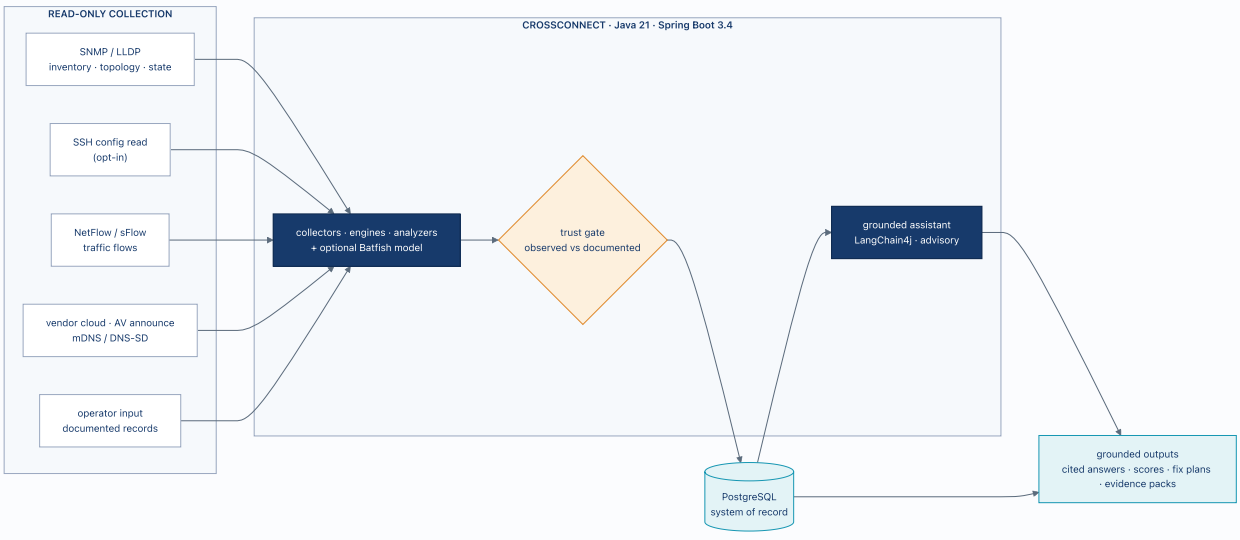

Figure 1. The platform at a glance. Read-only collection feeds real engines. Observations must pass a trust gate before they become truth in the single PostgreSQL system of record. Every output, whether a score, a fix plan, an evidence pack, or an assistant answer, traces back to that record. Nothing in this path writes to a managed device.

4 The architecture

The architecture is deliberately small. CrossConnect is a single deployable Java application backed by one PostgreSQL database, plus an optional formal-analysis helper that runs alongside it (a sidecar). One language from collector to user interface, one store, one thing to deploy. The stack is chosen so the whole system stays small enough to hold in one head and ship correctly.

Layer	Choice	Why it is here
Runtime	Java 21 (LTS)	Virtual threads, records, pattern matching; one language end to end
Framework	Spring Boot 3.4	DI, REST, filters, scheduling; mature ecosystem
System of record	PostgreSQL	Single store. JSONB, range types, recursive CTEs fit the network domain; no second database
Schema	Flyway	Versioned, production-safe SQL migrations
Operator UI	Vaadin Flow (LTS)	The interface is rendered in Java on the server, so there is no separate JavaScript frontend in the primary path
Formal analysis	Batfish sidecar OPTIONAL	A vendor-neutral model of device configs that answers reachability, access-list, and forwarding questions
AI orchestration	LangChain4j (provider-pluggable)	Runs in the same JVM; the AI model provider can be swapped out or turned off entirely
Collection	snmp4j · sshj · JDK HttpClient · JDK NIO multicast	Real protocol implementations, read-only, fast-timeout
Observability	OpenTelemetry	Traces and metrics export to any OTLP-compatible monitoring backend you already run

> **Real-first, never modeled.** Every capability tries the genuine source first: a real protocol, a real API, a real engine. It falls back to demo or seed data only when the live source cannot be reached, will not authenticate, or is not licensed. A value that came from the live source is marked *Confirmed*; a fallback or a best guess is marked *Inferred* or *Unconfirmed*. A modeled value is never dressed up as a measured one.

5 The observe-then-commit model

This is the design decision the whole product rests on: a freshly discovered fact is never quietly treated as truth. New observations land in a holding area that is only ever added to, never overwritten. There they are confidence-scored by how many sources agree, and they cross into the official source of truth only once they earn enough confidence to be committed. That is why an answer can be trusted, and why the AI labels how sure it is rather than presenting a guess as a fact.

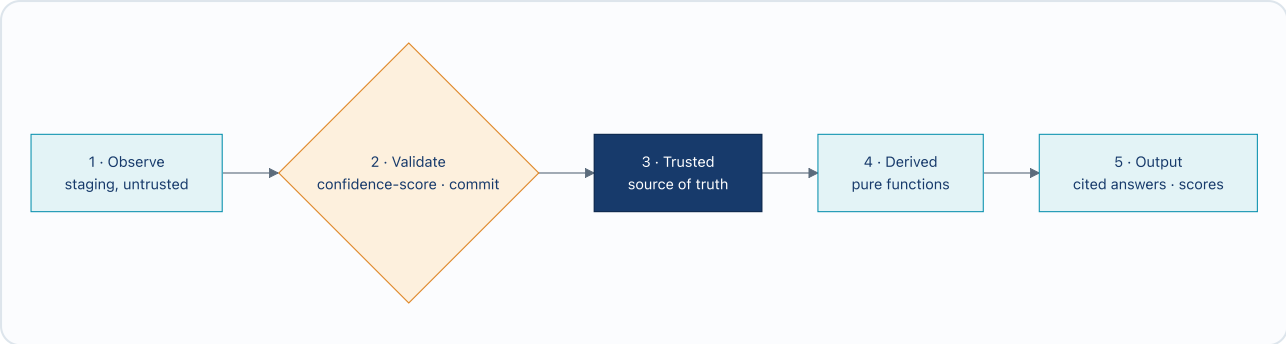

Figure 2. From observed to trusted. Two sources that agree score *Confirmed*; a single source scores *Inferred*; an observation that matches no known device is *Unconfirmed* and flagged as a possible rogue. A commit is the only way an observation becomes truth, and every one is audited. Derived results such as compliance, data quality, and reachability are calculated fresh from a snapshot on demand, not stored as new facts.

The same gate guards anything claimed from outside, whether a bulk import, an inbound event API, or an external feed. Nothing coming in becomes truth without passing through review, so the model stays trustworthy no matter where a claim started.

6 The capability spread

The platform is broad. It is one model viewed through many lenses, organized by the job you came to do. The table below shows the shape of it. For the full, current catalog, the in-product *Sitemap* and *Feature list* stay up to date.

Area	What it covers
Overview & triage	A ranked “what should I act on now” hotspot queue across every risk signal, with a causal “what changed, likely cause” trace; health, changes, network map, and L2 topology
Fleet & inventory	Devices, virtualization, services; vendors, models, modules, racks, locations, power feeds and power chain; the named service / intent layer above the boxes
Connections	Interfaces, cables, patch panels, wireless and Wi-Fi coverage; endpoint and MAC history; circuits, tunnels, console/out-of-band, L2VPN; path, VLAN, and IP-path tracing
Addressing / IPAM	Aggregates, IP spaces, prefixes, addresses, ranges; VLANs and groups, VRFs, FHRP, IPAM roles, DNS; trust-plane zones and a segmentation matrix
Assurance	Compliance scoring (CIS / NIST CSF / SOC 2), config hardening and grade, golden-config drift, CVE security, threat detection, lifecycle, data quality, operational maturity, and a single ranked per-device fix plan
Formal analysis BATFISH	Config-parse correctness, ACL and reachability checks, duplicate-IP detection, forwarding health, failure-impact blast radius, change preview, and the “network black box” that finds the exact config change that broke a flow
Routing & fabric	BGP and OSPF read from live MIBs and config intent, a clickable routing map, and spine-leaf (Clos) fabric validation

Area	What it covers
Operations	Service readiness, change safety, performance, capacity-to-full, storage, rack power, automation and event rules, jobs, upgrades, maintenance windows, reports, runbooks, search
Occupancy & space	Wi-Fi-derived space occupancy, scheduled-vs-actual room use, and occupancy source setup, passive, no cameras, confidence-banded
AI	The grounded assistant, bring-your-own-model setup, a write-intent confirmation queue, and an AI quality loop
Admin & integrations	Tenants, users and roles, discovery settings, secrets, encryption keys, webhooks and outbound sinks, API tokens, custom fields/links/tags, and the inbound/outbound connector catalog with its trust-gate review

Research previews, labelled as such. A small set of forward-looking lenses ship as clearly-marked experimental previews: *Gravitational Wobble* (spot an unmanaged device from the effect it has on managed gear), *Happy Auditor* (a one-click, control-mapped evidence pack), *Sense* (use the network itself as a building sensor), *Red Twin* (a risk-free simulated attacker that runs inside the formal model), and *Peek-a-Boo* (a view of the network-attached cameras, displays, codecs, and DSPs). Each one is read-only, advisory, and honest about how confident it is.

7 The AV lens

CrossConnect carries a first-class audio-visual (AV) lens that a plain device inventory does not offer. It works out the AV fleet from each device's vendor with no extra data entry, and labels discovered AV endpoints by role, codec, display, camera, microphone, DSP, or AV-over-IP encoder/decoder. It does this by combining signals it already collects: the hardware vendor prefix of a MAC address (the MAC OUI), mDNS service types, the media flows it sees on the wire, and model strings reported over LLDP and SNMP. Each label carries a Confirmed, Inferred, or Unconfirmed chip.

Is my AV safe

An AV segmentation score rates how well AV gear is walled off, for example cameras and mics that can be reached from the guest network, or stray AV gear sitting on a sensitive segment. It does this by combining the AV classifier, the zone model, and Batfish reachability.

Is my AV healthy

Timing (PTP / clock) health flags domains with only one clock source and clocks that have drifted out of lock. mDNS discovery health catches AV endpoints split across VLANs with nothing relaying discovery between them.

Does the network protect it

Quality-of-service (QoS) visibility reads running-configs into a per-interface policy inventory and flags AV-bearing devices with no policy, priority classes with no ceiling, or a missing trust boundary, each with a proposed fix written in that vendor's syntax.

Multicast, end to end

A weighted multicast health score across snooping, querier, routing, live delivery, and interop, with a flow map of every group and a ranked, vendor-aware troubleshooting hub.

8 The grounded assistant

The conversational layer reads from the source of truth; it is not an autonomous agent acting on its own. It answers questions about the model in plain language, and it is held to rules the software enforces, not prompt instructions it could be talked out of.

Grounded & cited

Every answer cites the record it used or says "I don't know." A citation validator rejects any answer that references a record the tools did not actually return, so the model cannot invent devices, IPs, or relationships.

Tenant- & role-scoped

The assistant sees only what the user asking is allowed to see. It honors role-based access controls and keeps each tenant's data separate, exactly as the rest of the API does.

Advisory only

It explains, highlights, and tells you how to fix. It never executes a change on a device. Any proposed change is queued as a write intent for a human to confirm.

Fully audited

Every prompt, the records retrieved, and the output are logged to a tenant-scoped AI audit entry, so any answer can be reconstructed and reviewed.

Provider and data flow. You choose the AI provider. Point it at an AI model endpoint of your choosing, under your own key, or leave it switched off, in which case the platform falls back to fixed, non-AI responses. Only the question and the exact records needed to answer it are sent to the model. Secrets are never included. CONFIGURABLE

9 The trust posture

This is the same posture a security review will probe, stated plainly here so leadership knows its shape before the deep-dive. The full control-by-control answer lives in the *Security & Architecture Reference*.

Read-only by construction

Discovery uses read-only credentials and read-only protocol operations. There is no code path that writes configuration to a managed device. No packet capture, no payload inspection, no SPAN/mirror feeds, no endpoint agents.

Runs on your infrastructure

Self-hosted in your data center, private cloud, or a managed instance you control. Your network data stays in your PostgreSQL. There is no mandatory vendor cloud in the data path.

Encrypted, with managed keys

Secrets are encrypted with AES-256-GCM under a layered key scheme (a master key protects per-record keys). The master key never ships in code, is read from your environment or secret manager, and can be rotated without re-encrypting your data.

Tamper-evident audit

Every change is written to an audit trail where each entry is cryptographically linked to the one before it and signed. Altering any past entry breaks the chain and is caught on verification. That chain is the integrity evidence behind every answer.

Users sign in through your own identity provider (IdP) using OIDC, and access is granted by role and scoped to the tenant on every request. The tenant is the dividing line between customers: every row carries a `tenant_id`, every query filters on it, and one tenant reading another's data is prevented by the structure itself, not just by policy.

10 Where it runs

Installation is a single set of containers, not a sprawl of separate processes to wire together. The same artifact runs from a quick evaluation to a hardened production deployment.

Mode	Shape	Fits
Demo	One container, bundled sample data, resets on restart	Evaluation, screenshots, a walkthrough on seeded data
Small	Single container, one data volume	Smaller estates; one-command install
Standard	App container + dedicated PostgreSQL (Docker Compose)	Production up to large estates
Managed / hosted	Container against a managed PostgreSQL instance	Cloud-hosted deployments you control

The optional Batfish helper joins the set when you want formal analysis. Apart from the application port itself, all of its traffic to the network is outbound and read-only; the platform opens no inbound listeners facing the customer network.

11 What we deliberately do not do

Knowing what not to build is part of the product. CrossConnect stays a source of truth and an intelligence layer. It does not try to become the system that makes your changes for you.

- **It never executes a change.** Fixing is advice for a person to act on: it highlights, explains, and tells you how to fix. It will not push a change to a device on its own authority, not even with a confirm-before-commit step.
- **It is not an automation framework.** It offers a clean way to emit events (signed webhooks and outbound sinks) and plugs into the schedulers and orchestration you already run, rather than building its own.
- **It does not inspect traffic content.** It reads switch-derived signals and the control-plane facts gear already exposes. It never looks at packet contents and does no deep packet inspection.

- **It does not overstate compliance.** Posture is stated in precise, defensible language; controls are designed to map onto recognized frameworks, and any certification not actually held is not claimed.
- **It does not bluff.** A modeled value is never shown as a measured one, and the assistant says “I don't know” rather than make up an answer.

> **Built for** the network teams, AV integrators, managed service providers (MSPs), and IT leaders who have to stand behind the network they run, and who want to walk into an audit or a budget review knowing exactly what they run and why. One system of record, kept honest by the network itself.

CrossConnect by CybrIQ · Platform Overview · High-altitude orientation · 21 June 2026 · Capabilities described reflect the shipped operator-preview build; items marked *experimental preview* are forward-looking and labelled in-product. · contact_us@cybriq.io