SOLUTION BRIEF

QRadar® + AiStrike

IBM QRadar + AiStrike

Keep QRadar. Add an AI layer that investigates every offense, explains its reasoning, and tunes your detections over time.

> AiStrike connects to IBM QRadar over its REST API and operates as an AI-native control plane above it. QRadar keeps collecting, normalizing, correlating, and generating offenses. AiStrike investigates those offenses with identity, asset, vulnerability, and threat-intel context, produces an evidence-backed conclusion with a full audit trail, and feeds what it learns back into rule tuning. No log migration. No rule migration. No infrastructure change.

Detection was never the bottleneck

QRadar is a mature, high-throughput correlation engine, and it does that job well. The constraint sits downstream. Offenses land in a queue faster than analysts can work them, and each one costs the same manual routine: pivot to identity, pivot to the asset inventory, pivot to the scanner, pivot to threat intel, reconstruct the timeline, write it up.

Three things follow from that. Offenses get triaged by whoever is on shift, so investigation quality tracks analyst experience rather than a standard. Low-priority offenses get closed on pattern recognition instead of evidence, because there isn't time to do otherwise. And detection rules never improve — nobody has the hours to go back and fix the rule that generated 400 benign offenses last month.

The result is a SOC that is fully staffed and still behind.

How AiStrike works with QRadar

QRadar

QRADAR CONTINUES TO OWN

- Log collection and ingest
- Event normalization
- Correlation rules
- Offense generation

AiStrike

AISTRIKE OPERATES ABOVE IT

- Investigates qualifying offenses as they are created
- Enriches with identity, asset, vulnerability, and threat-intel context
- Builds attack timelines, root cause, and blast radius
- Determines business impact against asset criticality
- Recommends or executes response, approval-gated
- Returns findings to detection engineering as tuning input

Because the integration is API-based, the AI layer is not coupled to the SIEM underneath it. If the SIEM changes later, the SOC workflow doesn't.

What teams run on it

AI Investigation of QRadar Offenses

Automated investigation, offense grouping, prioritization, root-cause analysis, MITRE ATT&CK mapping, and a written investigation report per case.

Detection Engineering

Identifies noisy rules and coverage gaps, recommends tuning, and reports on detection effectiveness so rule quality improves instead of drifting.

Exposure Analytics

Attack-path analysis, asset criticality weighting, vulnerability prioritization, and identity-risk correlation to separate exploitable exposure from inventory noise.

Threat Intelligence Operationalization

IOC enrichment, actor attribution, and campaign correlation applied to live offenses rather than sitting in a feed nobody reads.

Response Automation

Host isolation, user disablement, firewall updates, ticket creation, and notifications — with human approval gates on anything that touches production.

Executive & Regulatory Reporting

Investigation summaries, incident records, and KPI reporting built from the same audit trail the analyst worked from.

Explainable by construction, not by report

Every conclusion carries the evidence it was built from and the reasoning that connected it. Analysts can open any step, see which data source answered which question, and disagree with it. Response actions on critical systems wait for a human. Regulated environments can pin specific workflows to deterministic playbooks rather than model judgment.

The learning loop is the part that compounds. Analyst feedback and investigation outcomes are inputs to detection tuning, so the rules that generate the noise get fixed instead of filtered. That is a different objective than suppressing alerts, and it produces a different result twelve months in.

Composite AI. Machine learning, knowledge graphs, and LLMs working together — not an LLM wrapper around a SIEM API.

What changes

TRADITIONAL QRADAR SOC	QRADAR + AISTRIKE
Manual offense triage	Every qualifying offense investigated
Analysts manually gather evidence	Automated evidence collection
Multiple console pivots per case	Single AI-guided investigation workflow
Reactive investigation	Context-aware investigation with business impact
Static detection rules	Continuous detection optimization
Manually written investigation reports	AI-generated reports with linked evidence
Quality varies by analyst	Consistent, explainable, fully audited

Deployment

01 — CONNECT

AiStrike authenticates to QRadar over REST API. Read access to offenses and the telemetry needed to investigate them.

02 — SCOPE

Define which offense types are investigated automatically. Start narrow, expand as confidence builds.

03 — ADD CONTEXT

Connect identity, asset inventory, vulnerability, threat intel, and ticketing sources. Context is what separates an investigation from an enrichment.

04 — TUNE

Analysts review findings and correct them. That feedback shapes both future investigations and QRadar rule recommendations.

No log migration. No rule migration. No agents to deploy.

Prove it on your own offenses

A structured proof of concept runs in four to six weeks against live QRadar data, with success criteria agreed before ingestion begins.

WEEKS 1–2

Connect QRadar and context sources. Establish baseline: offense volume, investigation time, false-positive rate, analyst effort per case.

WEEKS 3–4

AiStrike investigates offenses end to end. Timelines, root cause, blast radius, recommended response, generated reports.

WEEKS 5–6

Detection tuning against the noisiest rules and coverage gaps. Executive review against the baseline.

Every measurement is against your own starting numbers, not a benchmark.

Outcomes

Hours become minutes. Manual investigation effort drops sharply, and time to investigate moves from hours to minutes.

Detection improves, not drifts. Detection quality improves continuously instead of degrading as rules age.

Analysts do judgment work. Analyst hours go to the cases that need human judgment, not routine evidence gathering.

Audit-ready by default. Investigations become consistent and auditable enough to hand to a regulator.

Frequently asked

Does AiStrike replace QRadar?

No. QRadar keeps collecting, correlating, and generating offenses. AiStrike layers above it. There is no log migration and no rule migration.

Does it replace our SOAR playbooks?

No. Deterministic playbooks remain available and are the right answer in regulated workflows. AiStrike adds investigation and reasoning ahead of the playbook and can hand off to it for execution.

Which offenses get investigated?

Whichever ones you scope. Most teams start with a defined set of offense types and expand as the output proves out.

Will it act on its own?

Routine, high-volume actions can be automated. Anything critical waits for analyst approval. That gate is a design principle, not a configuration default.

How is this auditable?

Every investigation produces structured findings, timestamps, linked evidence, and the reasoning behind the conclusion. The record is the same one the analyst worked from.

What if we migrate off QRadar?

The integration is API-based and the AI layer is SIEM-independent. The SOC workflow carries over.

See it run against your offenses

A scoped proof of concept on live QRadar data, with success criteria agreed up front.

Request a demo