

CISO's Guide API Security

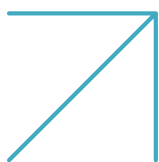
Understanding the evolving API threat landscape,
its newest challenges and what to do next





Table of Contents

Executive Summary.....	3
Introduction.....	3
Part I: The API Threat Landscape.....	4
API Usage & Change Dynamics.....	4
Attack Trends.....	4
Part II: CISO & Enterprise API Security Challenges.....	5
Part III: How to Overcome API Security Challenges.....	8
Operating Model & Runbook Considerations.....	10
Conclusion.....	11

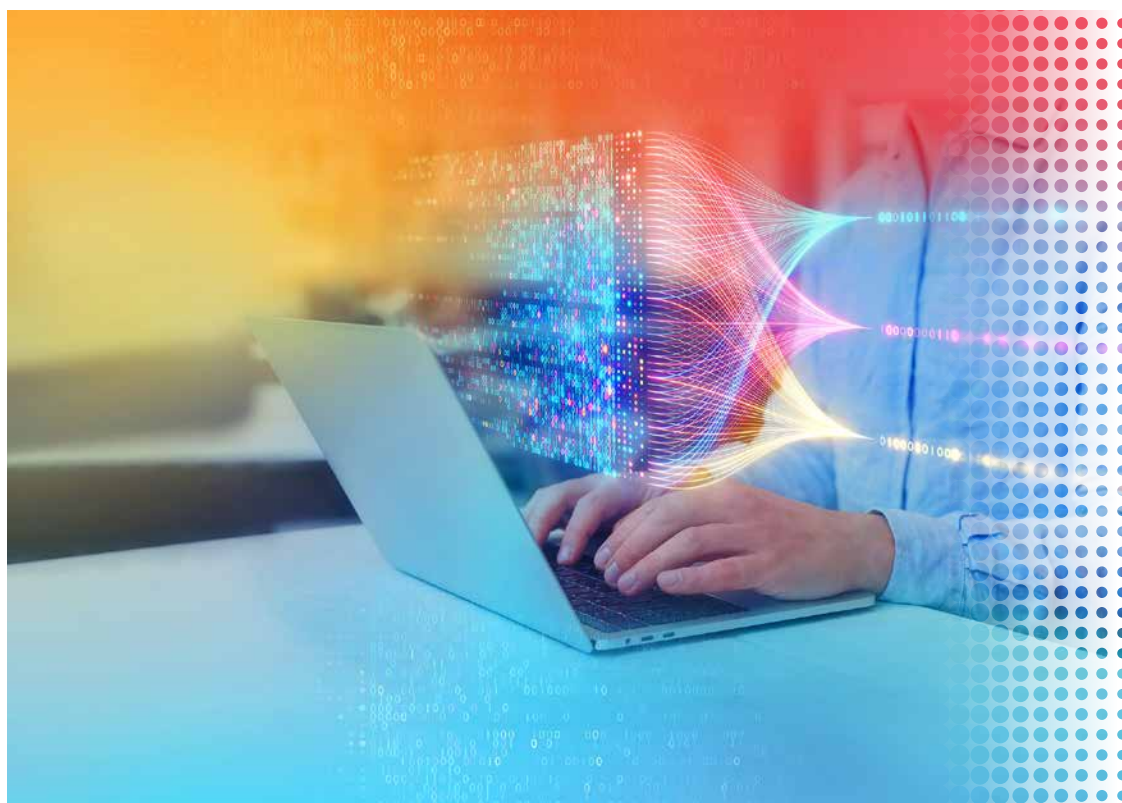


Executive Summary

- **APIs are business critical—and risky.** Velocity of change, third party dependence and incomplete documentation expand exposure faster than most controls can keep up.
- **Attacks have shifted to business logic.** Adversaries use GenAI to map workflows and automate multi step abuse that appears legitimate, evading traditional tools.
- **CISO mandate:** Establish automated discovery, runtime protection, unified posture management and consistent policies across environments—without slowing delivery.
- **Operational focus:** Reduce tool sprawl and false positives, align Dev/Sec ownership, and measure success through MTTD/MTTR, coverage, and policy consistency.

Introduction

APIs are the connective fabric of digital business, powering customer experiences, partner ecosystems and automation at scale. As adoption accelerates, so does complexity and risk. Attackers are evolving faster than legacy tools, and the most damaging exploits now target business logic, not traditional vulnerabilities. For CISOs, API security is a strategic capability tied to resilience, compliance and growth.



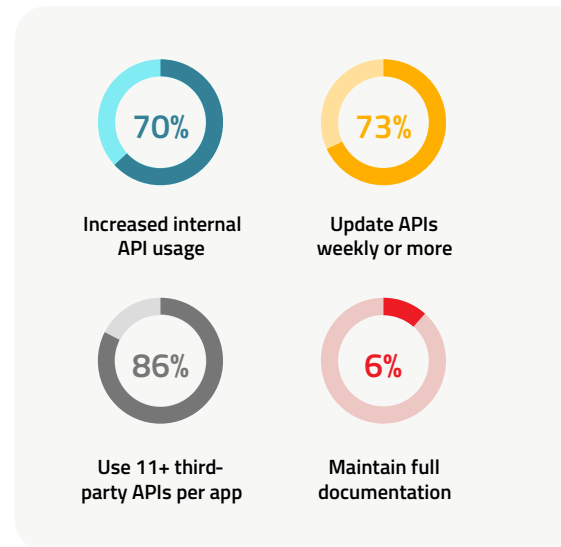
Part I: The API Threat Landscape

The latest Radware research reveals the following insights into the ways organizations are using APIs:

API Usage & Change Dynamics

- **70%** of organizations report increased internal API usage
- **73%** update APIs weekly; daily updates have increased **6x** since 2023
- **86%** use **11+ third party APIs per application**
- **Only 6%** maintain full documentation of all APIs

What this means for CISOs: Rapid changes, deep dependency chains and documentation gaps create structural visibility and governance challenges.



Attack Trends

A recent Radware survey shows that API-targeting attacks are more than a passing occurrence—in fact, they're commonplace. Here's what organizations told us:

- **75%** experience API attacks quarterly or more
- **67%** encounter business logic attacks (BLA) quarterly
- **86%** experienced BLA in the past year

Research from Radware's Cyber Threat Intelligence team digs deeper, providing an up-close look at specific API attack types and targets:

- **84%** of credential-stuffing scripts now target API endpoints
- **26%** of API-targeted attack configurations focus on older API versions
- **94%** of credential stuffing attacks implement advanced business logic elements

Attackers increasingly use GenAI to reverse-engineer business logic, simulate real user journey flows and chain multi-step attacks.

Implication: GenAI eliminates the need for insider knowledge; adversaries can automatically infer workflows and logic paths, making API-based fraud and exploitation easier than ever.

Landscape Summary

Expanding API ecosystems, rapid change velocity and emerging AI-driven attack techniques widen the enterprise attack surface. Most organizations still lack the runtime visibility and defenses needed to keep pace.

Part II: CISO & Enterprise API Security Challenges

With cybercriminals' focus squarely set on all types of visible and hidden APIs, organizations must step up their API security or face attack and downtime. Here's a look at seven API challenges currently standing in their way:

1. Poor Visibility & Control

Organizations struggle to maintain complete, accurate API inventories across business units, cloud platforms and legacy environments. As a result, shadow, outdated and third-party APIs frequently go unmonitored.

Operational & cost impact:

- Manual discovery consumes significant analyst time
- Blind spots result in surprise incidents and expensive investigations
- Fragmented tools increase TCO and complicate ownership

How to overcome this challenge: Continuous, automated discovery and classification—with ownership attribution, schema extraction and risk tiering

2. Real-time Protection Gaps

APIs present a vast and diverse threat surface. Runtime protection must address:

- **Business Logic Attacks (BLAs)** targeting API sequences and business workflows
- **Embedded threats** and injections hidden in API calls and nested JSON/XML or encoded payloads
- **Bot/ATO abuse** that mimics legitimate API flows
- **API-layer L7 DDoS** designed to exhaust functional endpoints
- **Third-party/supply-chain attacks** via compromised SDKs or external APIs

Even well-patched APIs remain at risk, as logic flaws exploit intended behavior. GenAI accelerates reconnaissance, reducing incident response windows.

Operational & cost impact:

- High alert noise and triage fatigue slow MTTR
- Over-blocking impacts revenue; under-blocking increases fraud
- Emergency releases and hotfixes disrupt engineering priorities

How to overcome this challenge: Schema-aware, behavioral runtime protection with inline and out-of-path modes—covering logic abuse, embedded threats, L7 DDoS, bots, and authentication enforcement

3. Lack of Posture Management & Risk Assessment

Static posture tools miss real-world behavior. True risk emerges only when combining design specifications, configurations and runtime traffic insights.

Operational & cost impact:

- Noisy, theoretical findings waste DevSecOps time
- Misprioritization delays remediation of critical risks
- Hard-to-explain risk models complicate audits and board reporting

How to overcome this challenge: Runtime-driven posture management with risk scoring based on actual usage, exposure, and behavior

4. Inconsistency Across Environments

Distributed APIs live across multi-cloud on prem and hybrid environments—each with different gateways, deployment models and ownership.

Operational & cost impact:

- Drift between environments creates exploitable weak points
- Tool sprawl increases licensing and operational overhead
- Troubleshooting becomes slower and costlier

How to overcome this challenge: Unified policy engine with multi-gateway integration and cross-environment drift detection when all environments are connected

5. Dev/Sec Team Silos

Fast-moving development cycles outpace manual documentation and policy creation. Security often lacks context, while developers lack real time-risk visibility.

Operational & cost impact:

- Late-stage findings delay releases
- Rework increases engineering costs
- Ambiguous ownership slows incident response

How to overcome this challenge: Traffic driven documentation, shared inventories, and contextual API testing that adapts to business logic and covers the full OWASP API Top 10—including logic attacks

6. Compliance & Governance Difficulties

Regulators expect demonstrable control over API risk: inventory, logic-abuse detection, supply-chain oversight, and auditable policies, **including OWASP API Top 10 coverage and business logic protections.**

Operational & cost impact:

- Evidence collection burdens teams.
- Insufficient documentation leads to audit findings.
- Lack of forensic clarity increases legal and reputational exposure

How to overcome this challenge: Runtime-driven, explainable controls aligned to regulatory expectations, with logs and posture insights suitable for audits

7. Integration & Business Enablement Issues

Security must support—not slow—modern architectures. Teams need open, interoperable protection that fits their tooling without friction.

Operational & cost impact:

- CI/CD friction slows feature velocity
- Custom integrations raise maintenance costs
- Fragmented controls across WAAP, bot, and DDoS increase TCO

How to overcome this challenge: Cloud-native compatibility, coexistence with existing controls, and multi-gateway support in a unified management plane



Part III: How to Overcome API Security Challenges

A modern strategy and solution selection framework for CISOs includes the following capabilities:



Automated API Discovery & Risk-Based Inventory

Continuous identification of all APIs (first-party, third-party, shadow, deprecated) using runtime traffic and gateway integrations

Look for:

- Traffic-based discovery across environments
- Gateway-based identification of APIs
- Auto-generated schema extraction
- Ownership attribution and lifecycle tracking
- Risk scoring based on data exposure, auth models, and usage patterns

Operational outcomes: More accurate inventories, fewer blind spots, improved MTTR and simplified audits



Runtime Protection & Business Logic Attack Detection

Real-time protection against embedded attacks, logic abuse, bot traffic, L7 DDoS, and unauthorized access.

Look for:

- Behavioral and workflow-aware logic detection and mitigation.
- Schema validation (JSON/XML) for content integrity
- Embedded threat detection
- Integrated bot/ATO protection
- API-layer DDoS mitigation
- Inline and out-of-path deployment options
- Token and authentication enforcement

Operational outcomes: Lower false positives, reduced fraud/abuse, fewer emergency patches and clearer forensics



Unified Posture Management

Runtime-driven posture insights combining traffic analysis, configurations, and API usage patterns.

What to look for:

- Runtime risk scoring
- Misconfiguration detection
- Risk insights tied to real behavior
- Prioritization of real risk for Dev and Sec teams

Operational outcomes: Fewer theoretical alerts, faster resolution no alert fatigue, and clearer prioritization



Consistent Protection Across All Environments

Uniform security controls enforced across multi-cloud and on-prem environments

Look for:

- Multi-gateway and multi-native cloud stack integration
- Unified management plane
- Cross-environment drift detection
- Consistent policies across deployment models

Operational outcomes: Lower TCO, fewer audit findings and faster troubleshooting



Dev/Sec Collaboration

Shared visibility and tools that support both developers and security teams

Look for:

- Runtime-generated documentation and schema extraction
- Contextual API testing covering OWASP API Top 10 (including business logic attacks)
- Unified portal for both Dev and Sec
- Automated policy generation from live traffic

Operational outcomes: Faster delivery, fewer late-stage surprises and reduced rework



Compliance & Governance

Controls that meet regulatory expectations with clear evidence.

Look for:

- Coverage of OWASP API Top 10
- Business logic attack protection
- Runtime forensics and posture insights for audits
- Immutable logging and explainable policy outcomes

Operational outcomes: Shorter audit cycles, stronger defensibility, fewer findings



Seamless Integration & Deployment Flexibility

Security that adapts to modern, distributed architectures.

Look for:

- Cloud-native compatibility
- Multi-gateway coexistence
- Unified management with WAAP/bot/DDoS portfolios
- Low-latency inline or monitor-only modes

Operational outcomes: Less pipeline friction, streamlined operations and reduced tool sprawl

Operating Model & Runbook Considerations

Ownership & RACI: Clear owners per API, with risk-tiered SLOs

Triage & Response: Logic attack, bot surge, embedded threat, and L7 DDoS playbooks

Policy Lifecycle: Monitor → alert → block, with drift tracking

Metrics: Coverage percentage, false positive rate, MTTD/MTTR, fraud reduction

Cost Control: Consolidate overlapping controls; leverage unified API + bot + DDoS platform

Conclusion

To stay ahead of an increasingly hostile and API-driven threat landscape, CISOs must shift from fragmented, reactive controls to a unified, runtime-driven API security strategy. The organizations that will succeed are those that combine continuous discovery, behavioral protection against business logic abuse, and consistent policy enforcement across every environment—without slowing innovation.

Modern API security is no longer an enhancement to application security; it's a foundational requirement for operational resilience, regulatory defensibility and sustained digital growth.

This document is provided for information purposes only. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law. Radware specifically disclaims any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. The technologies, functionalities, services, or processes described herein are subject to change without notice.

© 2026 Radware Ltd. All rights reserved. The Radware products and solutions mentioned in this document are protected by trademarks, patents and pending patent applications of Radware in the U.S. and other countries. For more details, please see: <https://www.radware.com/LegalNotice/>. All other trademarks and names are property of their respective owners.

