CONTRAST SECURITY

SOLUTION BRIEF

Runtime protection for critical financial applications

Defending highly sensitive data against advanced threats

As the financial services and banking industry becomes more digitized with mobile wallets, wealth management APIs, digital identity verification and similar services, cybersecurity risks are increasing. The pressure to innovate rapidly while maintaining compliance and defending highly sensitive data against advanced threats creates a challenging security landscape.

One major issue is the complexity of legacy systems being integrated with modern applications. This hybrid environment often lacks standardized security controls, making it difficult to uniformly protect applications across platforms. Many of these applications are vulnerable to threats like insecure deserialization and broken access control, particularly at the application logic level.

Traditional security tools like SAST and DAST often fall short when it comes to spotting and fixing vulnerable code. They generate large volumes of noise, require extensive manual validation and cannot operate at the speed or depth required by financial institutions. Moreover, SOC teams often do not have direct insight into application behavior, making it difficult to detect attacks that bypass infrastructure-based defenses.

Additionally, the rise of fintech integrations and open banking APIs has introduced new vectors for attackers to gain unauthorized access to accounts, inject malicious payloads or abuse business logic.

The rise of fintech applications and APIs has introduced new vectors for attackers

- Over 71% of financial institutions said zero-day attacks were the biggest issue they faced in regard to safeguarding their applications
- 65% growth in application attacks in the financial services industry over 12 months
- The average cost of a breach in financial services is $4.9 million

The evolving attack surface: Securing fintech and open banking APIs

The financial industry's digital footprint continues to grow. APIs, mobile apps and open banking integrations expand the attack surface, exposing institutions to common threats such as injection, broken access control and insecure deserialization — core concerns of the OWASP Top 10.

As online banking evolves with new technologies, a significant challenge arises from the way these innovations interact with older systems, existing apps and even applications developed by third parties. This interconnectedness can create new security weaknesses. For instance, banking institutions often use hundreds of different applications, some of which are developed by third parties but hosted on the bank's own systems. In such cases, traditional security scans like Static Application Security Testing (SAST) can't help because they require access to the application's source code, which isn't available for these third-party tools.

Security Operations Centers (SOCs) are traditionally built to monitor networks and endpoints, not application-layer activity. As a result, they often miss sophisticated, logic-based attacks that exploit business workflows rather than technical vulnerabilities. These blind spots leave applications exposed to threats that evade traditional detection mechanisms.

The sheer volume of security alerts presents another major challenge. SOC teams find it hard to distinguish genuine application security threats from false positives or background noise, significantly slowing down incident response times and increasing the risk of missed threats.

In addition, the adoption of AI-assisted development tools is a double-edged sword. While these tools accelerate code generation, they can also introduce insecure code at a speed that human teams find difficult to monitor and manage. This adds pressure to already stretched application security teams.

Finally, financial institutions face an intense regulatory burden. Compliance requirements such as PCI-DSS, GLBA, NYDFS 500, and new rules like DORA in the EU demand ongoing vigilance and robust application security practices, placing additional strain on internal teams.

Without modern, runtime-based application security strategies, banks risk serious breaches that lead to financial loss, regulatory penalties and reputational damage.

Bottom line: Application security risks are overwhelming banking security teams.

Security teams need visibility into live application behavior and the ability to act fast. They must address zero-day vulnerabilities, insecure AI-generated code, and complex attack chains that target APIs, backend systems and modern web stacks all in real time. Point-in-time testing (SAST, DAST) and siloed legacy tools are not enough.

Why is runtime application security critical for banks and financial institutions?

Visibility into live attack behavior

Runtime application security inside the application, delivering real-time insights into attacks. Teams can understand, prioritize and block active threats, without the noise and delays of traditional tools.

A buffer for patch delays

In regulated and complex environments, patching takes time. Runtime protection can enforce compensating controls immediately even before a patch is ready helping maintain compliance and uptime.

Protection against increasing AI-generated code

Banking services such as customer portals often include code generated by AI or from third-party tools. Runtime security detects and neutralizes vulnerabilities in the code before they're exploited.

Scalable stack-wide protection

Runtime solutions scale to protect web apps, APIs, containers and third-party integrations supporting digital transformation without expanding risk.

The Contrast advantage: Innovate faster without compromising on security and compliance

Benefit	Value
Detect and quickly block attacks	Monitors application behavior and data flow identifying suspicious patterns and anomalies, and when a verified attack is detected, instantly blocks the malicious behavior and alerts security teams with rich, actionable context.
Zero-day resilience	Instruments applications from within, rather than relying on signature-based detection, enabling the detection and response to both known and unknown threats, including zero-day exploits, through behavioral analysis within the application runtime.
Automated remediation guidance	Delivers precise, actionable security insights, accelerating response times and ensuring that vulnerabilities are fixed before they impact critical services. Dynamic risk scoring brings a smarter way to prioritize based on what's actually at risk in production.
Reduce MTTR and strengthen security posture	Provides a unified platform that eliminates the guesswork that plagues traditional tools, enabling accurate, automated prioritization and remediation allowing teams to filter out noise, enabling a focus on the most critical risks to data and application integrity.

Why Contrast is different

Together, Contrast AST and ADR create a feedback loop between development, security and operations. AppSec teams can prioritize fixes based on exploitability data from AST, while SOC teams gain high-fidelity alerts enriched with code-level context.

CONTRAST APPLICATION DETECTION AND RESPONSE (ADR)

Protect applications and APIs from exploits and zero days.

Detect attacks on applications and APIs so security operations teams can respond before exploits occur.

CONTRAST APPLICATION SECURITY TESTING (AST)

Monitor code as it runs. Identify vulnerabilities instantly.

Prioritize and address risks with faster application and API vulnerability detection and fewer false positives.

CONTRAST ONE

Defend your applications and APIs with Contrast One.

Managed application and API security powered by the people who built it.

Ready to see the Contrast runtime security platform in action?

Learn more

contrastsecurity.com

6800 Koll Center Parkway
Ste 235
Pleasanton, CA 94566
Phone: 888.371.1333