Educational, Executive

PCI DSS 4.0: What it Means for AppSec and How Apiiro's Deep ASPM Helps


Saoirse Hinksmon **Published March 11 2024** · 11 min. read

Thank you to these industry experts for their contributions to this post: **David Fairman**, CIO & CSO, Netskope; **Roxanne Carr**, SVP AppSec, Synchrony Financial.

PCI DSS 4.0 outlines significant changes for AppSec teams, placing greater emphasis on continuous, programmatic, and proactive application security than ever before. Dig into the new and updated PCI 4.0 requirements and learn how a deep ASPM can help with achieving compliance.

Nearly 20 years after its first release, the Payment Card Industry Data Security Standard (PCI DSS) is still a driving force in shaping secure software development and delivery requirements for organizations that process, store, transmit, or impact the security of cardholder data. If your organization fits that description, you are (hopefully) well aware of the latest update, PCI DSS 4.0, which introduces 64 new stipulations and major changes from PCI DSS 3.2.1.

At a high level, PCI 4.0 shifts towards everyday compliance rather than yearly checks, emphasizing continuous, programmatic, and proactive application security more than ever before. Thus, having a significant impact on application and product security teams.

PCI DSS 4.0 Implications for AppSec

If you don't have an integrated, comprehensive AppSec program with interoperable tools for unified visibility and testing coverage… you will now (or hopefully by the end of March when the first deadlines come up). PCI DSS 4.0 challenges AppSec teams to create and maintain well-rounded programs that can provide continuous security across application attack surfaces and resilience when an incident takes place.

Overall, to remain compliant with PCI DSS 4.0, AppSec teams will need:

- Formal and defined roles, responsibilities, and processes for detecting, prioritizing, remediating, and preventing security flaws in their software.
- Broad and deep vulnerability detection and security coverage across their entire application attack surface, now including APIs.
- Continuous inventory of bespoke and custom software, and third-party software components.
- Consistent, formal, objective processes for addressing security issues, particularly critical and high-severity vulnerabilities, that can be repeated, operationalized, and proven to mitigate risk.
- Detection and management plan for "significant" changes to systems.
- Secrets security detection, remediation, and management processes.
- Integrated vulnerability management processes outside of just AppSec, including network and broader systems.

- Secure code training for developers that is directly relevant to their role, the development language they use, and the tools they rely on.
- Ready access to evidence and acknowledgment that the requirements were understood and completed.

With many AppSec teams already strapped for resources and with developers often outnumbering AppSec personnel 100 to 1 (or more!), meeting these requirements at scale will require a concerted and holistic effort across teams, processes, and tools.

We'll dig into the new and updated PCI DSS 4.0 requirements that most impact AppSec teams and show how application security posture management (ASPM) offers a unified and automated solution to ensure governance across the development lifecycle.

What's changed in PCI DSS 4.0 for AppSec

The majority of changes impacting AppSec teams are detailed in these requirements:

- \6. Develop and Maintain Secure Systems and Software.
- \8. Identify Users and Authenticate Access to System Components (8.6.2 New requirement for not hard-coding passwords/passphrases into files or scripts for any application and system accounts that can be used for interactive login.)
- \11. Test Security of Systems and Networks Regularly.
- \12. Support Information Security with Organizational Policies and Programs.

Requirement 6 is the most important for AppSec teams, reinforcing the need for secure software development processes, including documentation of security policies, roles, and responsibilities. It introduces new requirements for reviewing code and addressing vulnerabilities earlier in the development lifecycle, maintaining a complete and continuous software inventory, implementing contextual developer security training, and more.

Requirements 8, 11, and 12 reinforce the need for strong access controls, preventing hardcoded secrets, extensive testing (both automated tests and penetration tests), and methods to operationalize your AppSec program like annual formal risk assessments, implementing security policies, assigning developer training, and more.

Meeting PCI DSS 4.0 Compliance with Apiiro's Deep ASPM

With more pressure on security than ever before (see the recent legal ramifications for Solarwinds, Citibank, etc.), PCI DSS 4.0 challenges AppSec teams to ramp up their programs, and likely with the same or a similar budget as previous years.

Here's how Apiiro can help you efficiently meet the new requirements outlined in PCI DSS 4.0.

Identify and address security vulnerabilities prior to being released

Requirement 6.2.3: Bespoke and custom software is reviewed prior to being released into production or to customers, to identify and correct potential coding vulnerabilities, as follows:

- Code reviews ensure code is developed according to secure coding guidelines.

- Code reviews look for both existing and emerging software vulnerabilities.
- Appropriate corrections are implemented prior to release.

Requirement 6.2.4: Software engineering techniques or other methods are defined and in use by software development personnel to prevent or mitigate common software attacks and related vulnerabilities in bespoke and custom software…

With the shifted focus on continuous security, most of the requirements outlined in 4.0 involve infusing security into each stage of the development lifecycle. To do this at scale, you need a unified approach to risk detection, prioritization, and remediation that can support proactive and automated risk responses.

How Apiiro helps: As an open platform, Apiiro integrates with application security testing and scanning tools to correlate all your findings. Plus, Apiiro has native software supply chain security, SCA, API security, secrets security, and more to ensure risks don't fall through the cracks. The coverage table verifies security coverage across all different tools and application components within your application attack surface, so you can easily identify gaps.

Maintain a complete application and software supply chain inventory

Requirement 6.3.2: New requirement to maintain an inventory of bespoke and custom software.

Requirement 6.3.2.a: Examine documentation and interview personnel to verify that an inventory of bespoke and custom software and third-party software components incorporated into bespoke and custom software is maintained, and that the inventory is used to identify and address vulnerabilities.

Requirement 12.5.1: An inventory of system components that are in scope for PCI DSS, including a description of function/use, is maintained and kept current.

Requirement 12.3.4: review software technologies used yearly.

PCI 4.0 introduces new requirements around maintaining a software inventory so that you can better understand and protect your application attack surface. Now, you must maintain a continuous inventory of "all bespoke and custom software and third-party components." With the new expanded scope, this inventory also extends to APIs.

How Apiiro helps: By connecting to your source control management (SCM) system and across your development stack (AST tools, CI/CD pipelines, K8s clusters, and more), Apiiro automatically builds and maintains a comprehensive inventory that encompasses both your proprietary code and all other code components your applications rely on. This extended software bill of materials (XBOM) includes code and software supply chain components, developer behavior, sensitive data, APIs, and more. Our graph-based model exposes the relationships between components, how they change over time, and associated risks. Additionally, this inventory breaks down all of the technologies in use, including frameworks, languages, encryption algorithms, and more, so that when you need to verify and prove that your inventory aligns with your policies, you have data-driven evidence on hand.

Track, audit, and programmatically respond to "significant" changes

Requirement 6.5.1: Changes to all system components in the production environment are made according to established procedures that include:

- Reason for, and description of, the change.
- Documentation of security impact.
- Documented change approval by authorized parties.
- Testing to verify that the change does not adversely impact system security.
- For bespoke and custom software changes, all updates are tested for compliance with Requirement 6.2.4 before being deployed into production.
- Procedures to address failures and return to a secure state.

Requirement 6.5.2: Upon completion of a significant change, all applicable PCI DSS requirements are confirmed to be in place on all new or changed systems and networks, and documentation is updated as applicable.

Requirement 11.4.2: Internal + (11.4.3 external) penetration testing is performed:

- Per the entity's defined methodology,
- At least once every 12 months
- After any significant infrastructure or application upgrade or change
- By a qualified internal resource or qualified external third-party
- Organizational independence of the tester exists (not required to be a QSA or ASV).

Requirement 12.10.5: The security incident response plan includes monitoring and responding to alerts from security monitoring systems, including but not limited to:

- Intrusion-detection and intrusion-prevention systems.
- Network security controls.
- Change-detection mechanisms for critical files.

Teams will need to scan everything in scope for PCI DSS (which now includes APIs and updated requirements around secrets), detect and address any significant changes, and verify that code reviews, penetration tests, and other risk assessment processes are executed.

How Apiiro helps: As a deep and open ASPM, Apiiro is uniquely positioned to automatically analyze development behavior and security signals from disparate tools, giving us an in-depth understanding of not only risks, but also code changes that may introduce risk. This makes it possible to not only correlate known vulnerabilities (like XSS, SQLi, etc.), but also identify risky code changes and toxic combinations, disparate risks that, when connected, may pose a serious threat to your business.

If opting for manual reviews, Apiiro can trigger code reviews and penetration tests through our workflows, so that if and when an issue is found (either through 3rd party tools or our native solutions), this review process is automatically initiated. Plus, we can also assign relevant training courses with Secure Code Warrior and provide a timeline to track and verify changes.

Prioritize security vulnerabilities based on potential likelihood and impact

Requirement 6.3.1: Security vulnerabilities are identified using industry-recognized sources for security vulnerability information, including alerts from international and national computer emergency response teams (CERTs).

- Vulnerabilities are assigned a risk ranking based on industry best practices and consideration of potential impact.

- Risk rankings identify, at a minimum, all vulnerabilities considered to be a high-risk or critical to the environment.
- Vulnerabilities for bespoke and custom, and third-party software (for example operating systems and databases) are covered.

Requirement 12.3.1: Each PCI DSS requirement that provides flexibility for how frequently it is performed (for example, requirements to be performed periodically) is supported by a targeted risk analysis that is documented and includes:

- Identification of the assets being protected.
- Identification of the threat(s) that the requirement is protecting against.
- Identification of factors that contribute to the likelihood and/or impact of a threat being realized.
- Resulting analysis that determines, and includes justification for, how frequently the requirement must be performed to minimize the likelihood of the threat being realized.

In PCI DSS 4.0, distinguishing between critical and low-risk vulnerabilities is crucial, as the new requirements include vulnerability classification standards and explicit timelines for remediating critical and high-severity risks. Plus, you will need to provide evidence justifying that the actions taken were appropriate in addressing the severity of each risk.

How Apiiro helps: In line with Requirement 6.3.1., Apiiro uniquely contextualizes all these findings and prioritizes risks based on the potential likelihood of a risk being exploited and the impact on the business in such a case. Apiiro's multidimensional risk rank and prioritization automatically layers industry risk standards, like CVSS scores CISA KEV, EPSS, exploitability, CVE/CWE, etc. with code-to-runtime context (including unique likelihood and impact) to distinguish security issues that are real risks to the organization. Apiiro then streamlines remediation processes with workflows, risk-specific context, and actionable guidance.

Remediate all critical and high vulnerabilities within 30 days

Requirement 6.3: Security vulnerabilities are identified and addressed.

Requirement 6.3.3: All system components are protected from known vulnerabilities by installing applicable security patches/updates as follows:

- Critical or high-security patches/updates (identified according to the risk ranking process at Requirement 6.3.1) are installed within one month of release.
- All other applicable security patches/updates are installed within an appropriate time frame as determined by the entity (for example, within three months of release).

Requirement 11.3: Verify that internal scans occurred at least once every three months in the most recent 12-month period, and that all high/critical vulnerabilities are addressed within 30 days (inline with requirement 6).

To remediate all high and critical vulnerabilities within 30 days, you need to be able to streamline your remediation processes. Doing so requires automation, but it also requires more context around every vulnerability.

How Apiiro helps: To meet the 30-day remediation deadline outlined in these requirements, Apiiro's policies, workflows, and custom SLAs can help align fix timelines with due dates. Plus, you can use the Explorer to identify any outstanding issues that are at risk of not meeting the 30-day deadline. Apiiro can also trigger penetration tests via workflows as well as incorporate pen test, bug bounty, and manual risk findings documented in ticketing systems. All of these coupled with Apiiro's ability to prioritize based on unique likelihood and impact makes it easy for teams to effectively and efficiently address critical and high risks within the 30 day deadline.

Prevent secrets from being hardcoded

Requirement 8.6.2: New requirement for not hard-coding passwords/passphrases into files or scripts for any application and system accounts that can be used for interactive login.

Requirement 8.6.2 introduces new requirements to ensure that your secrets aren't hard coded into scripts, configuration files, property files, or custom source code.

How Apiiro helps: Apiiro automatically detects secrets across your codebases and pipelines and evaluates their context, like whether it's valid, in test code, exposed to the internet, in a public repository, related to sensitive data, or is in a high business impact (HBI) application, and identifying the specific platform/system affiliations (e.g., AWS or OpenAI).

Integrate relevant secure code training for developers

Requirement 6.2.2: Software development personnel working on bespoke and custom software are trained at least once every 12 months as follows:

- On software security relevant to their job function and development languages.
- Including secure software design and secure coding techniques.
- Including, if security testing tools are used, how to use the tools for detecting vulnerabilities in software.

Requirement 12.6.2: New requirement to review and update (as needed) the security awareness program at least once every 12 months.

Requirement 12.6.3.1: New requirement for security awareness training to include awareness of threats and vulnerabilities that could impact the security of the CDE.

Requirement 12.6.3.2: New requirement for security awareness training to include awareness about the acceptable use of end-user technologies in accordance with Requirement 12.2.1.

These requirements emphasize that training needs to be directly aligned to job function and development languages and include secure software design and secure coding techniques.

How Apiiro helps: To make secure code training relevant, impactful, and non-disruptive for developers, Apiiro matches findings to relevant developer training courses and includes these in alerts. Each finding is mapped to a CWE, and the training courses are aligned to the developer's code language. With in-context and framework-specific training, developers can quickly complete training as a part of their everyday development cycles with contextual, just-in-time feedback.

Manage, prevent & measure application posture

To operationalize and maintain your AppSec program, Apiiro makes it easy to create and set policies, embed developer guardrails, assign relevant training, map security coverage, and streamline your AppSec program. To understand how much risk you're carrying and analyze trends, Apiiro Reports show your average risk age broken down by severity, your MTTR over time, and more. These reports can be filtered to specifically segment risks associated with sensitive data. With enterprise-grade features like SSO, RBAC, and audit logs, you can confidently scale your AppSec program without compromising on compliance.

Apiiro's ASPM helps AppSec teams do more with less by combining an open platform approach with native AST and SSCS solutions to unify AppSec risk visibility, assessment, prioritization, and remediation. To get started automating for PCI DSS 4.0 compliance, schedule a demo with our team.