

▼ Secure Web Applications a Critical Component of Payment Card Industry Data Security Standard (PCI-DSS) Compliance

Under the requirements of the **Payment Card Industry Data Security Standard (PCI-DSS)**, organizations that process customer credit card and financial information are required to ensure that their applications – in particular the web-facing components - are free of vulnerabilities that could lead to a compromise of customer data. Companies that fail to meet the requirements of the PCI Standard will be deemed non-compliant.

While there is no denying the costs associated with achieving compliance, for vendors who rely on the Internet the cost of non-compliance is even greater. In addition to the monthly fines levied by the credit card institutions, there are also the costs of lost business when – in the event of a breach – the same companies refuse to process client payments. This is in addition to the investigation, containment and legal costs associated with an incident, and the resulting damage to reputation and goodwill.

Impact of PCI-DSS on Web Application Security

The PCI-DSS standard was put in place by the major credit card vendors to ensure that merchants securely process and transmit customer and cardholder data. The standard, now administered by the **PCI Security Standards Council**, defines 6 overall categories and 12 specific requirements addressing Technical, Physical and Administrative security measures for cardholder data.

For online business, particular attention must be paid to Requirement 6 which addresses security for custom web applications. A number of PCI practices detailed under this requirement will become mandatory as of June 30 2008. These new requirements address the need to ensure that applications are developed with security in mind and that protections are put in place to protect against application-level exploits.

Requirement 6: Develop and maintain secure systems and applications

6.2 Establish a process to identify newly discovered security vulnerabilities and update standards to address new vulnerability issues.

Armorize Advantage: For commercial software, patching has become an industry in itself. But in the case of custom developed web applications, there are no associated patch release processes. Thus, if a custom web application is vulnerable from the outset, it remains so; even with a secure server and network infrastructure.

Armorize CodeSecure™ automates the task of identifying and removing critical code-level web application vulnerabilities throughout the development lifecycle. Accessible through a web browser or the Integrated Development Environment (IDE), CodeSecure™ offers both enterprise and desktop-level source code analysis to ensure applications are resistant to exploits such as Cross Site Scripting (XSS) and SQL Injection. Armorize constantly researches new attack categories and classes to ensure that CodeSecure™ can detect program code that would leave applications open to these attacks.

▼ Secure Web Applications a Critical Component of Payment Card Industry Data Security Standard (PCI-DSS) Compliance

6.3 Develop software applications based on industry best practices and incorporate information security throughout the Software Development Life Cycle (SDLC)

6.3.7a Review of custom code prior to release to production or customers in order to identify any potential coding vulnerability

6.3.7b Verify code reviews are conducted for new code and after code changes

Armorize Advantage: Security-specific review of source code is a critical step in detecting web application vulnerabilities. As it can be initiated early in the SDLC, it addresses code-level vulnerabilities before the application goes live. However, if actioned manually, it is time-consuming, offers low repeatability and is difficult and costly to implement as an ongoing process. By automating this process, Armorize CodeSecure™ presents a highly efficient and repeatable low cost source code analysis solution that can be deployed at key points in the SDLC ensuring secure web application code at minimum cost and effort.

Note: This requirement applies to code reviews for custom software development as part of the SDLC. As of June 30, 2008, custom code for web-facing applications will be subject to additional controls as outlined in Requirement 6.6

6.5 Develop all web applications based on secure coding guidelines such as the Open Web Application Security Project (OWASP) guidelines. Review custom application code to identify coding vulnerabilities. Cover prevention of common coding vulnerabilities in software development processes, to include vulnerabilities identified by OWASP.

Armorize Advantage: CodeSecure™ integrates with the development process to address web application vulnerabilities identified by OWASP ensuring that web application code is free of vulnerabilities such as Cross Site Scripting (XSS), Injection (SQL, Command, File, etc.), Remote File Inclusion, Direct Object Reference, Information Leakage, Session Flaws and Unrestricted URL Access.

6.6 Ensure that all web-facing applications are protected against known attacks by either reviewing custom code for common vulnerabilities or installing an application layer firewall in front of web-facing applications. *(Note: This method is considered a best practice until June 30, 2008, after which it becomes a requirement)*

As per *Information Supplement: Requirement 6.6 Code Reviews and Application Firewalls Clarified (2008-04-15)*, the requirements for 6.6 can be addressed by either one of the following options:

Option 1: Having all custom application code reviewed for common vulnerabilities by an organization that specializes in application security

Armorize Advantage: The standard states that the application code review can be achieved by “proper use of automated application source code analyzer (scanning) tools” and that as long as the Source Code Analyzer addresses exploits defined by OWASP, minimum requirements are considered to have been met. CodeSecure™ can identify the actual line of code vulnerable to a range of exploits - including those defined by OWASP - with reporting providing a full trace between the coding flaw and the Internet facing vulnerability. This greatly reduces false positives while detailed recommendations for mitigation address security awareness and education for developers.

▼ Secure Web Applications a Critical Component of Payment Card Industry Data Security Standard (PCI-DSS) Compliance

Option 2: Installing an application layer firewall in front of web-facing applications

Armorize Advantage: The information supplement for Requirement 6.6 defines policy, logging, inspection and self protection capabilities for the Web Application Firewall (WAF). It also states that while implementing a WAF is one option to meet Requirement 6.6, it does not eliminate the need for a secure software development process as outlined under Requirement 6.3, and that "*proper implementation of both options would provide the best multi-layered defense.*"

While ideally the code review process ensures that web application code is secure, it may not be practical to rewrite source code for an application that is already running. In this case, a WAF configured to address identified source code vulnerabilities acts as a compensating control. Armorize SmartWAF™ meets this requirement by importing the CodeSecure™ reports and dynamically reconfiguring its rule set to explicitly block web application exploits targeted at the identified vulnerabilities.

Armorize Technologies Solutions Assist in Achieving PCI Compliance

PCI Compliance is not easy. It requires skills, processes and technology across the entire Information Security spectrum to ensure that credit card information is safe during transmission, processing and storage.

For Internet-enabled businesses, the web application is the primary interface between the internal information infrastructure and the Internet. Therefore, it is critical that these web-facing application offer maximum security while facilitating business. By deploying Armorize CodeSecure™, code level vulnerabilities that leave applications open to exploits such as XSS and Injection are identified and eliminated.

With Armorize SmartWAF™, extra protection is ensured as the findings of CodeSecure™ are imported to dynamically create rules blocking exploits targeting code-level vulnerabilities identified by the Source Code Analysis process.

About Armorize Technologies

Armorize Technologies provides next-generation Web Application Security Solutions traversing the System Development Life Cycle (SDLC).

Armorize was formed in 2005 with its headquarters in Santa Clara, CA, and its R&D centre in Taipei, Taiwan. The culmination of years of research and innovation, its award-winning solutions include static source code analysis with CodeSecure™, real time web application protection with SmartWAF™ and malicious code detection with HackAlert™.

The company has a global customer base across the government, financial, e-commerce and services sectors.

[Read more about our Web Application Solutions](#)