DataStealth

PCI DSS v4.0 eSkimming Protection

Version 1.3 Oct. 18th, 2024

datastealth.io | info@datastealth.io

Table of Contents

Introduction

PCI DSS v4.0 brings stringent requirements to enhance data security in the face of evolving cyber threats. Notably, requirements 6.4.3 and 11.6.1 introduce critical mandates for monitoring and tamper detection to protect payment pages. These new stipulations are now incorporated into all Self-Assessment Questionnaire types, regardless of how many transactions they process, or whether they utilize a Third Party Service Provider (TPSP). However, merchants can claim to be SAQ A if they have “confirmed that their site is not susceptible to attacks from scripts that could affect the merchant’s e-commerce system(s).”

Requirement 6.4.3 focuses on the management and integrity verification of payment page scripts **and also includes the pages and navigational flows leading up to the payment page**. It mandates that “unauthorized code cannot be executed in the payment page as it is rendered in the consumer’s browser,” which requires creating and maintaining a comprehensive inventory of all scripts, and allowing only authorized scripts to be executed throughout the entire payment process.

Requirement 11.6.1 focuses on detecting unauthorized modifications to payment pages, including scripts and HTTP headers. It mandates these detection activities be performed at least once every seven days or at intervals determined by the entity’s targeted risk analysis, marking a significant shift towards proactive security measures.

Failure to comply with these new requirements can result in the suspension of payment processing, increased audit scrutiny, and reputational damage.

DataStealth's eSkimming Protection solution excels in addressing these challenges, offering real-time compliance solutions that go beyond the basic compliance thresholds set by PCI DSS v4.0. DataStealth meets and surpasses these two new requirements by ensuring continuous monitoring and proactive response capabilities, providing a robust framework for safeguarding sensitive payment information against unauthorized access and tampering. Unlike other solutions, DataStealth does not require application or workflow changes, but a simple DNS update. With **no additional development work needed for integration**, the path to enhanced security and compliance is drastically simplified.

This document is particularly relevant for decision-makers and security professionals within organizations subject to auditing.

PCI DSS v4.0 Overview

PCI DSS v4.0 sets the latest global data security standards for all entities that store, process, or transmit payment card data, including merchants, processors, acquirers, issuers, and service providers. It aims to enhance the security of payment card transactions, protect cardholders against misuse of their personal information, and ensure the integrity of the payment ecosystem.

6.4.3 - Detailed Analysis of Requirement

Requirement 6.4.3 extends beyond the strict management of scripts on payment pages; it also encompasses any page that incorporates scripts or links leading to the payment page. This broader applicability significantly increases the scope of this requirement, bringing a vast number of existing merchants who previously might not have been considered within scope due to their reliance on Third-Party Service Providers (TPSP) providing iFrames as a way to offload this requirement to a third-party. With this update, even those merchants utilizing TPSPs can be subject to this requirement, marking a substantial shift in scope, responsibility, and applicability. However, merchants can claim to qualify under SAQ A if they have confirmed that "their site is not susceptible to attacks from scripts that could affect the merchant's e-commerce system(s)." This approach would allow them to mark requirements 6.4.3 and 11.6.1 as "Not Applicable," but will require them to implement the control and technical solutions necessary to meet requirement 11.6.1.
The requirement specifies that organizations must have a process in place that ensures:

- A method is implemented to confirm that **each script is authorized**.
- A method is implemented to ensure the **integrity of each script**.
- An **inventory of all scripts** is maintained with written justification as to why each is necessary.

This new requirement targets scripts directly involved in payment processing and extends to **any** scripts loaded on the payment page, regardless of the intended purpose. The expanded scope requires merchants to create a script inventory and provide a business or technical justification for the presence of **every** script, including widely used ones like Google Analytics, TikTok, and others.

This requirement introduces a significant challenge, especially for existing pages that rely on a wide variety of scripts for functionality beyond payment processing. Organizations must evaluate each script's purpose critically and remove any script lacking a clear business or technical justification to meet the security standard.

11.6.1 - Detailed Analysis of Requirement

Requirement 11.6.1 focuses on ongoing tamper detection for the contents of payment pages. It specifies that organizations must have a process in place that ensures:

- A **change- and tamper-detection mechanism** is implemented to alert personnel to unauthorized modifications (including indicators of compromise, changes, additions, and deletions) to the scripts and HTTP headers and the contents of payment pages as received by the consumer browser.
- The mechanism is configured to **evaluate each payment page's scripts and HTTP headers**.
- The mechanism functions are performed **at least once every seven days** OR periodically (at the frequency defined in the entity's targeted risk analysis).

Browser-level monitoring is key to web security, as websites pull content from multiple external sources. While some may think the best place to detect unauthorized changes is on the consumer's browser (where all content, including JavaScript, is executed), this is high-risk.

Relying on the consumer's browser for compliance is flawed because script-based or sensor-based solutions depend on a predefined **execution order**. These solutions run an inventory script that checks other scripts on the page. If a malicious script loads before the inventory script, **it can go undetected**. This risk makes the solution vulnerable to attacks that manipulate script loading sequences, compromising the entire payment process.

These solutions have several other limitations:

- **Limited detection:** Script-based solutions often look for missing signals to confirm they are still active. This might catch broad attacks but fail to detect more subtle, targeted changes. For example, a sophisticated attacker could remove the script-based tool on 1% of the transactions, making it extremely difficult to detect the attack and creating a compliance gap.
- **The "script that monitors the scripts" may not be tamper-protected:** Every script embedded in a payment page must be inventoried, authorized, and protected from tampering. This requirement also applies to any script added to the page, and includes the often overlooked "script monitoring the scripts."
- **Alert-only**: Script-based solutions only issue alerts and can't block malicious scripts, which does not comply with requirement 6.4.3 which mandates blocking unauthorized code.
- **Browser compatibility issues**: Script-based solutions rely on compatibility with the consumer's browser. However, none of the script-based solutions provide 100% coverage of all browsers. Some of these solutions are only compatible with the five

most popular browsers leaving a significant coverage gap, rendering them non-compliant with this requirement.

eSkimming Protection Solution

DataStealth's eSkimming Protection is an innovative solution specifically designed to meet and exceed the stringent requirements of PCI DSS v4.0, mainly focusing on requirements 6.4.3 and 11.6.1. DataStealth provides a fundamentally different and more robust approach to securing payment pages that **only requires a small DNS change.**

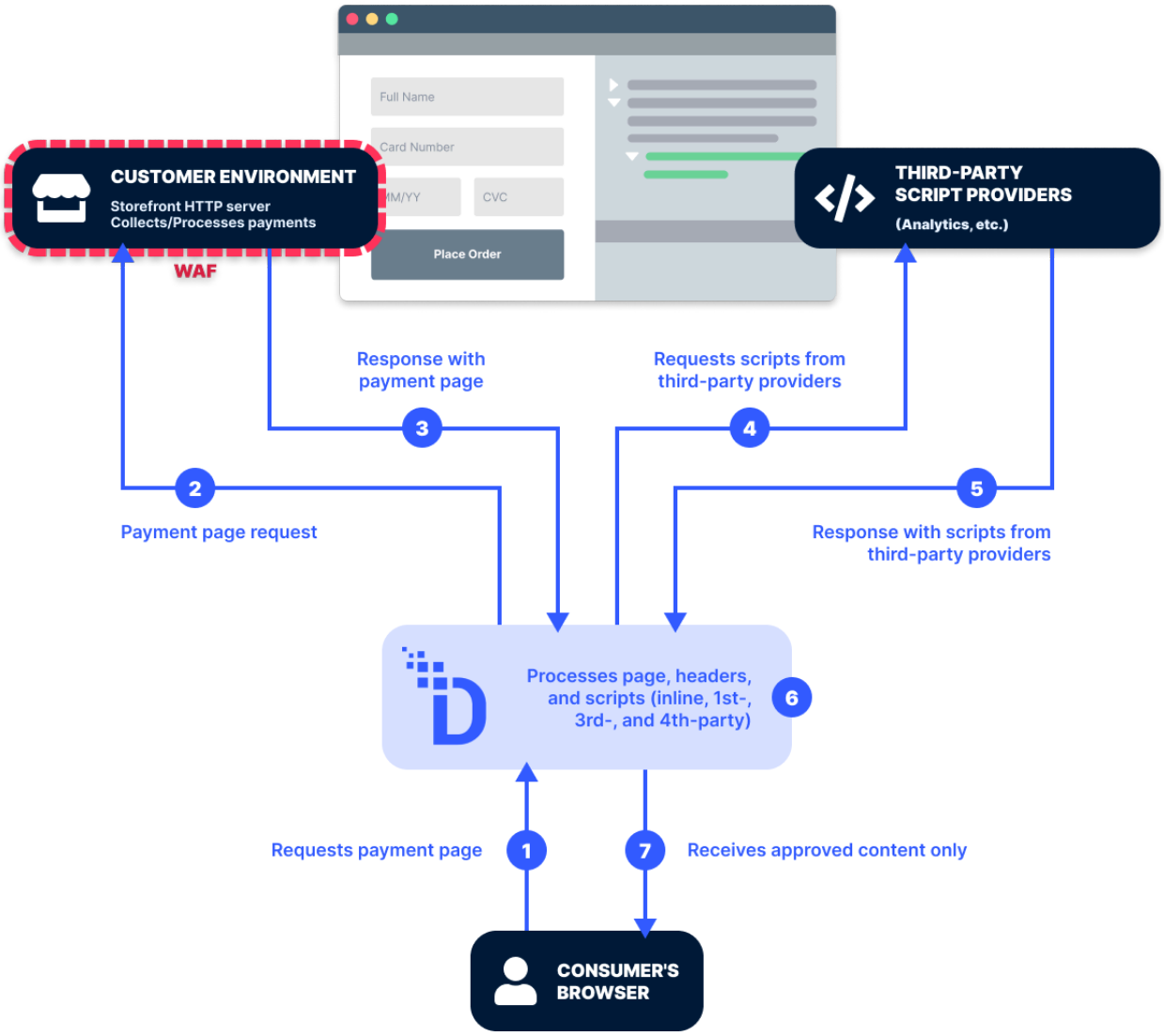

Figure 1 - DataStealth deployment

Key benefits of the DataStealth solution

No code change

- **Seamless integration:** Unlike script-based solutions, DataStealth does not require any scripts, agents, or collectors to be installed on the payment pages. This eliminates the risk associated with script injection and ensures that all potential vulnerabilities related to browser compatibility are avoided.
- **100% server compatibility:** DataStealth does not require any code to be added to the payment pages. For server platforms that don't allow users to modify page code, traditional script-based solutions are not an option—making DataStealth the ideal solution.

No dependency

- **100% browser compatibility:** DataStealth ensures that every consumer browser is fully protected, regardless of its market share or popularity. This universal compatibility guarantees that all traffic, from all users, is secured at all times.
- **Compliance Control:** Once the content reaches the consumer's browser, its only job is to render it. Your compliance is no longer dependent on a device or browser you don't control.

Proactive threat management

- **Real-time alerting and enforcement:** DataStealth's approach is proactive rather than reactive. It actively prevents malicious scripts from reaching the consumer's browser, rather than merely detecting and relying on the consumer's browser to block their execution. This proactive blocking is critical for maintaining the integrity of payment pages, significantly reducing the risk of tampering or unauthorized changes.
- **Automated compliance:** The solution automates compliance tasks, reducing the need for manual oversight and ensuring continuous adherence to 6.4.3 and 11.6.1.

Dynamic script removal

- **Efficient compliance acceleration:** DataStealth can dynamically remove any script that cannot be justified for technical or business reasons. This capability allows organizations to accelerate their compliance efforts with requirements 6.4.3 and 11.6.1, ensuring that only authorized and justified scripts are on the payment page.

Addressing Requirement 6.4.3 with DataStealth

Real-time Script Cataloging

Each time a payment page is loaded, DataStealth dynamically analyzes the payment page's structure to catalog each script, ranging from directly embedded scripts (1st party) to those loaded dynamically from external sources (3rd and 4th party).

A unique challenge in modern web applications is the extensive reliance on third-party libraries such as jQuery, React, and others, for building client-side applications. These applications rely heavily on these libraries and frequently reuse scripts across a single page, complicating tracking and integrity verification. The dynamic loading of third- or fourth-party scripts coupled with the external hosting and updating of these libraries, introduces vulnerabilities and dependencies that are difficult to manage.

DataStealth simplifies the process of cataloging and approving these scripts by isolating the script functions, regardless of how they are invoked, the frequency with which they are run, or where they are hosted. This method allows us to catalog any script function as a single entity, ensuring efficient monitoring without the need to check every instance individually.

As part of your path to compliance with DataStealth, we assist in addressing security concerns for both inline, first-, third- and fourth-party scripts. When an authorized first- or third-party script is used, we can enhance page security by embedding Subresource Integrity (SRI) tags. For dynamically loaded scripts or third-party scripts that initiate further script loads, DataStealth seamlessly redirects these to be managed through our PCI-compliant infrastructure. This process allows for integrity checks on the script payloads, assuring adherence to PCI DSS requirements.

Additionally, as this is a real-time process performed every time a page loads, we ensure a granular level of monitoring that traditional scheduled checks cannot match. Older and less effective solutions operate on a scheduled basis to conduct reviews only at specific moments, which may result in extended periods of vulnerability between each scan. By maintaining a real-time inventory of authorized scripts, DataStealth promptly identifies any unauthorized alterations, including indicators of compromise, changes, additions, and deletions, **every time a page is sent to a consumer browser** thereby significantly reducing the window of vulnerability to script tampering or injection attacks.

Comprehensive Script Integrity Checks

For scripts hosted on the payment page itself, DataStealth employs a sophisticated validation process each time the page is loaded. Recognizing that Subresource Integrity (SRI) cannot be applied to third- and fourth-party scripts due to their dynamic nature, we implement inline integrity checks of the content during the request processing.

Dynamic Content Monitoring

During the initial onboarding phase, DataSealth performs a comprehensive review of a client's web applications. This involves thoroughly examining scripts, CSS, and third-party elements to identify all external and dynamic content. This process identifies elements requiring intervention, such as content being rerouted through DataStealth, to allow the modification of scripts or the removal of unnecessary external components altogether. This focused, one-time analysis ensures that our clients' web applications are fortified against unauthorized modifications and broader web vulnerabilities from the outset.

Blocking Tampered Content Before it Reaches the Consumer's Browser

DataStealth ensures full compliance with requirement 6.4.3 by blocking unauthorized headers or scripts **before they reach the consumer's browser**. Alternate solutions that only issue alerts and rely on the organization to react swiftly and accurately to the tamper alert increase the risk of e-skimming attacks. DataStealth intercepts and analyzes **every header and script** in real-time, **every time a page is served to a consumer's browser**. Authorized scripts and pages proceed, while unauthorized or tampered scripts or pages are blocked instantly, ensuring they never reach the consumer's browser or compromise the payment process. This eliminates reliance on the consumer's browser for protection and fully aligns with the PCI DSS v4.0 mandate to ensure that "unauthorized code cannot be executed in the payment page as it is rendered in the consumer's browser."

Addressing Requirement 11.6.1 with DataStealth

For requirement 11.6.1, DataStealth introduces a distinctive service offering that uniquely positions DataStealth to assume responsibility for this critical requirement on behalf of our clients. This is not merely a tool or a product you need to install, maintain, and manage on an ongoing basis; it's a comprehensive service designed to safeguard the integrity of HTTP headers, scripts, and payment page contents against tampering.

As part of our annual Service Provider Level 1 audit, we can provide our customers with an Attestation of Compliance (AOC) that includes a detailed roles and responsibilities matrix (RACI). This allows DataStealth to assume responsibility for specific compliance requirements from our clients.

Immediate Tamper Detection Notification or Blocking

Unlike conventional methods that perform periodic checks, DataStealth's solution reviews the scripts, HTTP headers, and the content of payment pages every time they are accessed by a consumer. This real-time approach ensures that any unauthorized modification—be it an indicator of compromise, addition, deletion, or change—is detected

instantly. By minimizing the detection delay, DataStealth drastically reduces the potential damage and exposure window, providing an immediate response to threats. Additionally, conventional methods only alert clients upon integrity check failures. DataStealth has the unique capability to immediately block, rewrite, or take direct actions on the page to protect the consumer's browser.

Integration with Change Control Systems

DataStealth's eSkimming Protection integrates seamlessly with clients' change control systems, ensuring tamper alerts align with internal protocols and compliance.

Processing Tamper Events

We establish a process in advance with our clients, allowing them to select their preferred response to any unauthorized changes. This process can range from blocking traffic on payment pages when a tamper alert is received to simply notifying a designated contact while keeping the page active. By agreeing on this together in advance, we ensure that our actions align with our clients desired processes and protection levels.

Pre-Approved Change Windows

When updates to the payment pages are anticipated, DataStealth coordinates with clients to inform them about the timing of these changes. This allows us to temporarily disable alerts and notifications during the update process, ensuring no false alarms are triggered. After the updates are completed, we proceed to review the changes. This process ensures that our clients have complete control over their update schedules, with the assistance of the DataStealth Team readily available.

Change Management Process

Our solution is designed to work in concert with an organization's existing change control systems. Whether configured to block traffic or simply send alert notifications, this alignment ensures that any alarms generated by the tamper detection mechanism are swiftly evaluated and addressed according to the organization's compliance strategy. DataStealth facilitates a streamlined process whereby clients are promptly notified, allowing for a coordinated response:

1. **Incident Detection:** Upon detecting a change, the system immediately triggers an alert and may also begin blocking traffic if configured to do so:
    a. Notification is sent to designated client contact.

    OR

    b. Traffic is blocked and notification is sent to the designated client contact.

2. **Notification Dispatch:** The client receives a detailed notification outlining the nature of the change.
3. **Follow-Up Window:** A designated review period, mutually agreed upon with the client, is provided to assess and respond to any detected changes.
4. **Verification Process:** The client consults their team to verify the change, then proceeds with formal approval.
5. **Change Confirmation:** Upon receiving approval from the client's change management process, DataStealth promptly resolves the tamper alert and formalizes the change's acceptance.

Operational Considerations

PCI DSS v4.0 brings multiple stakeholders into the fold for managing website scripts and tamper detection. Organizations must **begin planning now**, as this broader scope means various departments (e.g., marketing, third-party providers, IT) may unknowingly introduce risks by deploying new scripts without proper security validation.

We anticipate that these new requirements will surface change management complexity with many teams, including marketing and third-party vendors, influencing which scripts are deployed on payment pages. Without centralized control, a business can quickly lose track of approved and secure scripts, increasing vulnerability. For example;

- Marketing teams might introduce new tracking scripts to monitor campaigns, which could unintentionally compromise script integrity on payment pages.
- IT teams using automated deployment tools could accidentally introduce outdated or unauthorized scripts to the production environment.

To ensure security and compliance, all departments must coordinate their change management practices. Delaying the planning process will **reduce the time available to adjust internal workflows and improve communication** between departments before the mandatory blocking of tampered pages and scripts takes effect on April 1st, 2025.

Best Practices

- **Operate in Alert-Only Mode Until March 2025:** Given the operational complexity and involvement of multiple stakeholders, DataStealth recommends operating in alert-only mode until March 2025. This approach allows organizations to identify operational gaps without interrupting business operations. During this period, organizations can refine stakeholder communication processes and tune monitoring systems.

- **Implement Strong Change Control Procedures:** Robust change control systems are critical to evaluating new scripts before being deployed to production. Organizations can minimize the risk of unauthorized script deployment by coordinating with all stakeholders and ensuring proper review processes.
- **Frequent Script Audits:** Enterprises should conduct regular audits to ensure that all scripts are necessary, properly authorized, and compliant with PCI DSS guidelines.

How Merchants Completing SAQ A Can Mark Requirements 6.4.3 and 11.6.1 as ‘Not Applicable’

A merchant completing SAQ A can mark PCI DSS Requirements 6.4.3 and 11.6.1 as "Not Applicable" if their environment meets specific conditions outlined in the SAQ A document.

Conditions for Marking Requirements 6.4.3 and 11.6.1 as "Not Applicable"

Complete Outsourcing of Payment Processing:

The merchant must completely outsource all electronic storage, processing, or transmission of account data to PCI DSS-compliant third-party service providers (TPSPs).

The merchant's systems must not store, process, or transmit account data electronically.

E-commerce Channel Specifics:

For e-commerce merchants, all elements of the payment page/form delivered to the customer's browser must originate directly from the PCI DSS-compliant TPSP.

The merchant must confirm that their website is not susceptible to attacks from scripts that could affect their e-commerce system(s), such as malicious skimming scripts. This requires implementing technical solutions and controls that would comply with requirement 11.6.1.

No In-Scope Systems:

If the merchant has a redirection mechanism (e.g., URL redirects) or embedded payment forms (e.g., iframes) on their website and relies entirely on a TPSP for payment processing, then they do not have any systems in scope for those requirements.

In such cases, the merchant's website does not directly impact how account data is transmitted, making these requirements irrelevant.

Eligibility Confirmation:

The merchant must confirm their eligibility for SAQ A by meeting all criteria listed in the "Merchant Eligibility Criteria" section of the document.

Documentation in Appendix C:

When marking these requirements as "Not Applicable," the merchant must provide an explanation in Appendix C of the SAQ A document, stating that no systems under their control are in scope for these requirements.

By meeting these conditions and documenting their reasoning in Appendix C, merchants can appropriately mark Requirements 6.4.3 and 11.6.1 as "Not Applicable" while remaining compliant with PCI DSS guidelines.

Conclusion

In conclusion, DataStealth's innovative approach to addressing the requirements of PCI DSS v4.0 positions it as a leader in the field of PCI security. Our solution offers a sophisticated approach to meeting and exceeding the stringent requirements set forth by PCI DSS v4.0, specifically those outlined in sections 6.4.3 and 11.6.1. By leveraging real-time monitoring and advanced data security technologies, DataStealth not only ensures compliance but also provides enhanced protection against the ever-evolving landscape of cyber threats.

The cost of not complying with 6.4.3 and 11.6.1 can be significant, including the suspension of payment processing, increased audit scrutiny, and reputational damage. Non-compliance may also lead to costly breaches, exposing sensitive customer data and resulting in financial losses and legal liabilities. Additionally, failure to meet these requirements can disrupt business operations, erode customer trust, and result in long-term damage to brand reputation.

DataStealth's solution helps organizations avoid these risks, ensuring they maintain compliance and safeguard their critical data assets, all while minimizing potential business interruptions and penalties.

To discuss your PCI DSS v4.0 strategy, please email us at info@datastealth.io or visit our website.