The Client-Side Protection Guide

How to Select a Unified Solution to Meet PCI DSS v4 Requirements, Prevent Skimming Attacks, Protect Consumer Data Privacy, and Stop IP Theft.

jscrambler

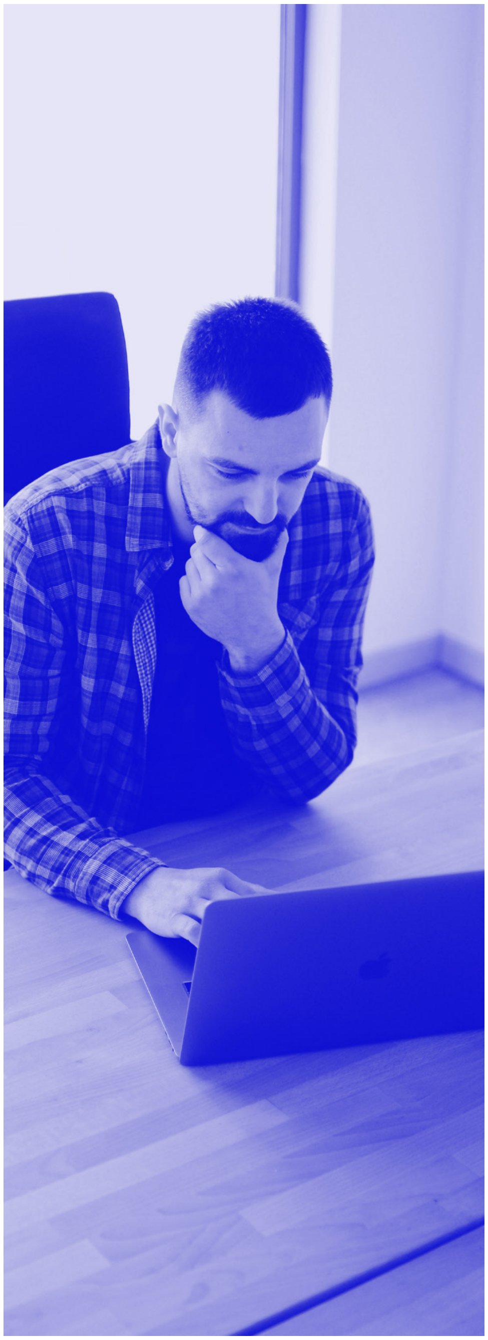


Table of Contents

Executive Summary

The past few years have witnessed a significant shift towards client-side innovation by digital businesses worldwide, largely driven by JavaScript. Since its invention in 1995 by Brendan Eich and subsequent integration into Netscape Navigator 2.0, JavaScript has served as a foundational technology for the World Wide Web. Initially used by web developers to incorporate simple animations into their websites, JavaScript has evolved to play a far more central role in enabling advanced online user experiences in recent years, driven by two key recent innovations - JavaScript frameworks and third-party scripts (tags, analytics scripts, marketing pixels, etc.).

Organizations face growing threats originating from the client side—an area often overlooked in traditional security frameworks. This guide uncovers why client-side risks have become one of the biggest blind spots in application security and why it demands immediate attention from security, compliance, and business leaders.

In this guide, you'll learn about:

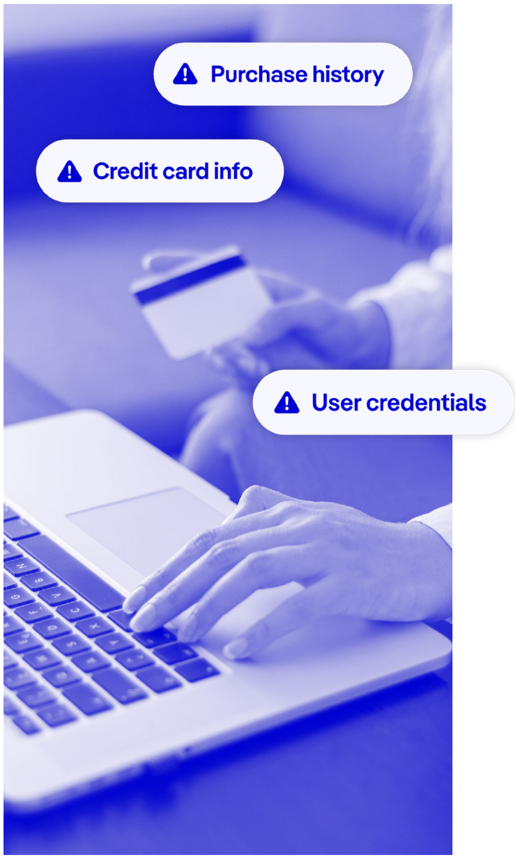
- **Why you might be overlooking the client-side**
- **The types of client-side attacks that impact business**
- **Who in your team should be tasked with selecting and evaluating a client-side protection solution**
- **What to look for in a client-side protection platform**
- **Where to start and end with a client-side protection checklist**

Whether you're just beginning to assess client-side risks or looking to strengthen your current defenses, this guide provides a comprehensive roadmap to safeguard the most exposed part of your application: the client side.

Why the Client Side is Your Biggest Blind Spot

Client-Side protection refers to the business policies and technologies that protect end users from malicious activities on dynamic web pages within the browser. The client-side security of web applications, or what happens in the user's browser, has been a low priority for businesses, thus increasing opportunities for fraudsters and threat actors to exploit end-user activities. In other words, relying solely on server-side security without also providing equal attention to client-side security measures leaves a significant gap in overall security health and customer protection.

The advent of JavaScript frameworks and the ease of integration of third-party JavaScript-based add-ons have enabled mainstream businesses to rapidly match the online experience that was previously the realm of the world's largest "digital native" businesses. With their back end fully accessible as APIs, development teams are now pouring their energy into client-side innovation. And thanks to JavaScript, delivering a GAFAM-grade experience is as simple as adding a JavaScript tag in the Tag Management System.

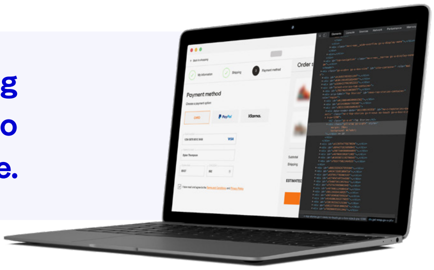


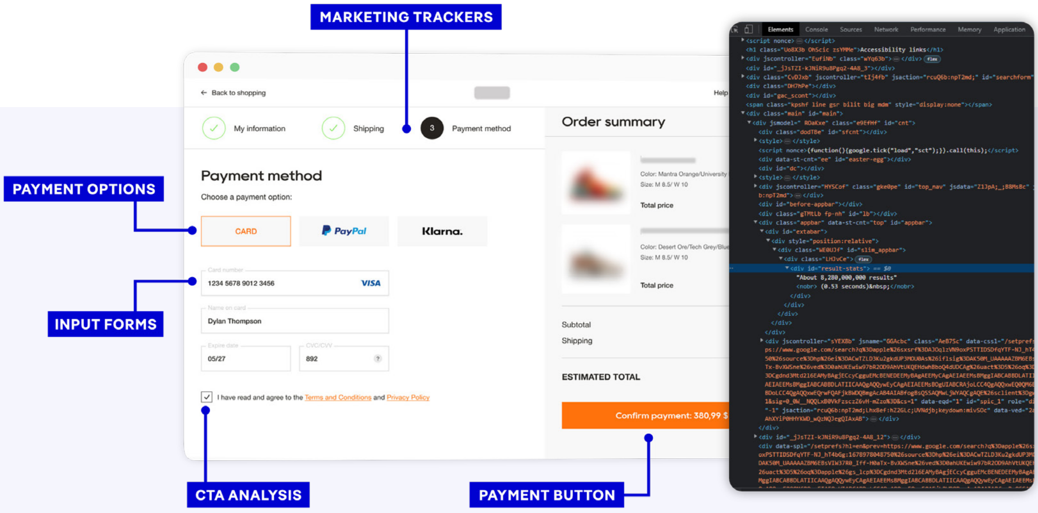
As a result, nearly all of the world's businesses can now match the savvy, advanced digital interfaces that until recently were exclusively offered by online giants such as Google, Amazon, and Facebook.

But here's the twist – the incredible ease of using JavaScript is also its downside.

All this pioneering client-side development hinges on JavaScript, a script that remains uncompiled—it's essentially text. This simplicity and elegance also render it highly vulnerable. JavaScript's key advantage lies in its rapid development pace, serving as a scripted, uncompiled language accessible to most junior developers.

However, the code appears plainly in the browser. Accessing any JavaScript code is as straightforward as navigating to "View ➔ Developer Options ➔ View Source Code" in Chrome.





Third-Party Add-Ons

Digital marketing and business teams significantly enhance online experiences by tapping into **thousands of third-party digital solutions.**

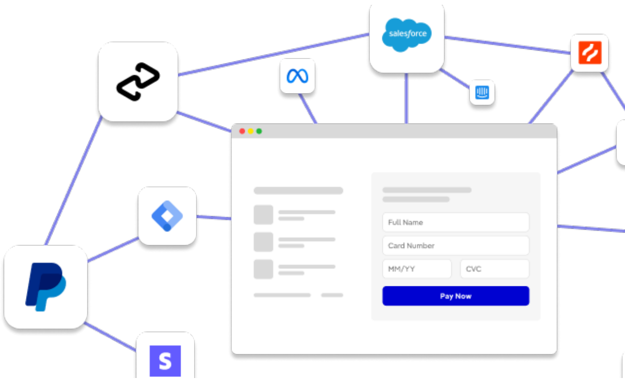
JavaScript Frameworks

Around **98.7%** of all websites across the globe now use JavaScript as their go-to client-side coding language.

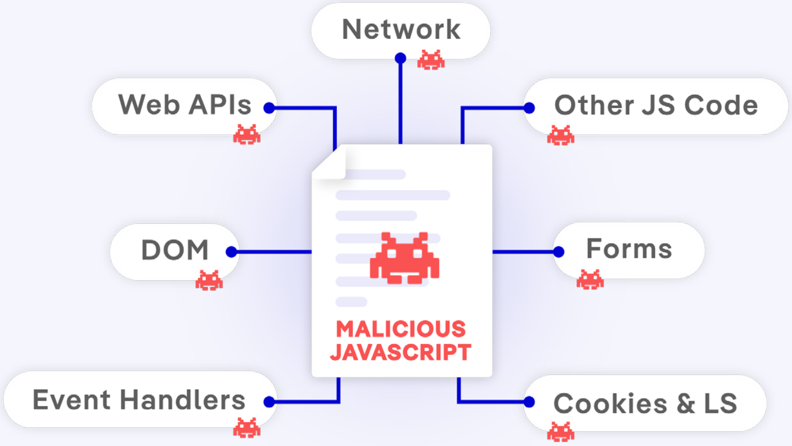
As businesses increasingly adopt client-side innovations based on first-party JavaScript development and the integration of third-party JavaScript-based add-ons, businesses must contend with new business risks.

Third-Party Add-Ons: What It Looks Like For Your Business

JavaScript and third-party add-ons are powering online experiences, giving companies the opportunity to capture more customer interest and insights. Digital marketing and business teams significantly enhance online experiences by tapping into thousands of third-party digital solutions for AB testing, analytics, advertising, marketing, payments, etc. But can third-party tags do more than what they are supposed to, or do harm to your website?



What Can Third Parties Do on a Website?



All scripts have the same power. They can:

- ⚠ Harvest any user input without the application owner or the users' knowledge;
- ⚠ Add extra code without your knowledge;
- ⚠ Hijack events;
- ⚠ Fully modify the behavior of the web page, tricking users into doing unwanted actions;
- ⚠ Tamper with other code in the same scope;
- ⚠ Contact any external domain and exfiltrate data they gathered, compromising compliance with regulations.

A recent Jscrambler report sheds light on the growing security concerns tied to third-party JavaScript tags embedded in websites. Although the vast majority of organizations (97%) recognize that these tags can access sensitive information, only a small fraction (13%) feel confident in knowing exactly what data is being collected. Nearly half (49%) experienced unauthorized data collection through these tags within the past year, and more than a quarter (26%) confirmed that confidential information was shared with external entities.

The report also highlights risks stemming from tools like Google Tag Manager, as 33% of companies reported that teams could add third-party tags without oversight, increasing the chance of security lapses and compliance violations.

While third-party tags are commonly used across websites, many companies remain unprepared to manage the associated risks. Only 36% have adopted formal strategies or tools to minimize threats like data skimming. Encouragingly, awareness is on the rise—68% of respondents acknowledged the critical need for client-side security solutions to prevent unauthorized access to user data.

97%
know that third-party tags collect sensitive data regularly

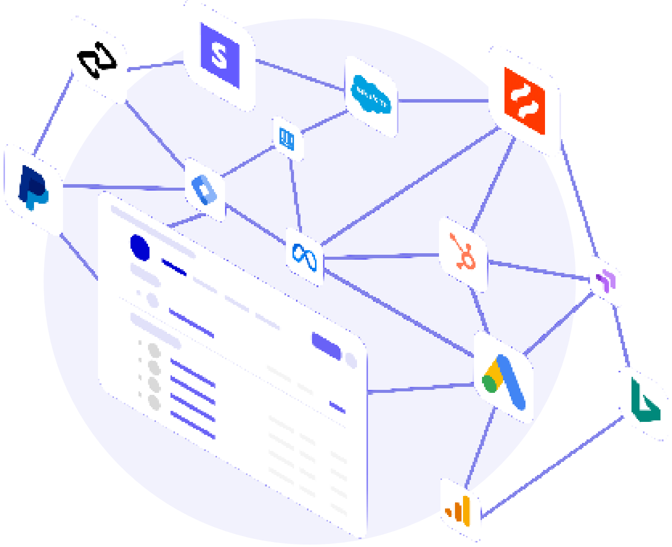
68%
believe a client-side protection and compliance solution should be deployed to protect user data

61%
state a tool that prevents digital skimming is key to achieving PCI DSS compliance

13%
are confident they know exactly what third-party tags are collecting

Examples of Third-Party Add-Ons Across Different Industries

Next are some examples of third-party vendors that may be present on your website, along with the types of data they might be collecting silently. On average, a website with a payment page has **60** third-party scripts.



E-Commerce

THIRD-PARTY ADD-ONS

Payment Gateways: Facilitate secure online transactions.

PayPal stripe Square

Analytics and Tracking: Monitor user behavior and site performance.

Google Analytics hotjar mixpanel

Customer Reviews and Ratings: Build trust through user-generated content.

yotpo. Trustpilot

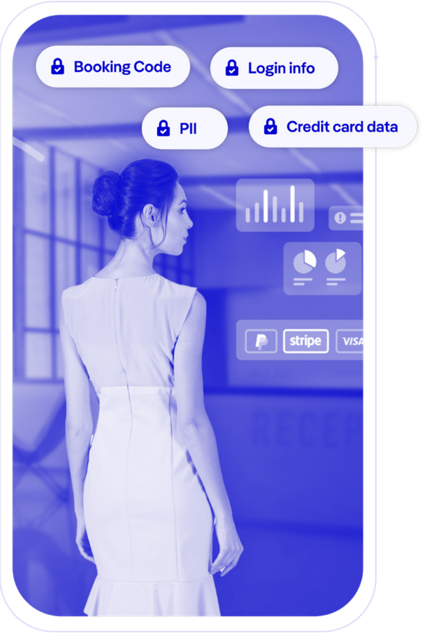
Live Chat and Customer Support: Enhance customer service and engagement.

Zendesk INTERCOM

Marketing Automation: Streamline email campaigns and customer segmentation.

mailchimp klaviyo

DATA ACCESSED



Financial Services

THIRD-PARTY ADD-ONS

Fraud Detection and Prevention: Identify and mitigate fraudulent activities.

Kount An Equifax Company riskified

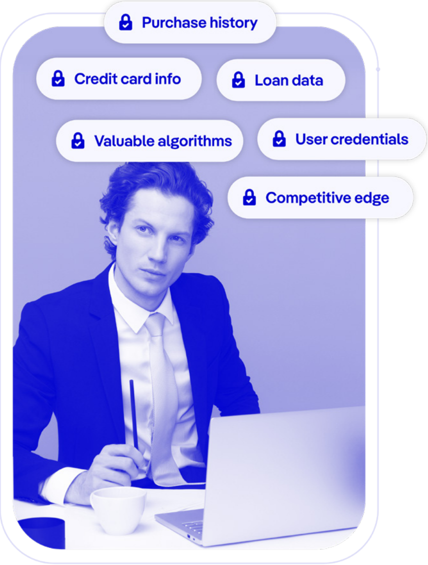
Financial Data Aggregation: Provide consolidated financial data insights.

PLAID ENVESTNET Yodlee

Customer Identity Verification: Securely authenticate user identities.

jumio onfido sumsub

DATA ACCESSED



Healthcare

THIRD-PARTY ADD-ONS

Telemedicine Platforms: Enable remote patient consultations.

Teladoc HEALTH   amwell

Electronic Health Records (EHR) Systems: Manage patient records digitally.

Epic   Cerner

Patient Portals: Allow patients to access their health information.

MyChart powered by Epic   athenahealth

HIPAA-Compliant Communication Tools: Ensure secure communication in compliance with regulations.

tigerconnect   PAUBOX

DATA ACCESSED

Third-party add-ons are essential for functionality across industries, but they also pose risks by accessing sensitive user data. When not properly monitored, these tools can become silent data leaks, exposing everything from credit card details to patient records.

High-profile data breaches across industries reveal just how vulnerable third-party systems can be. In e-commerce, British Airways fell victim to the **Magecart** hacking group, which injected malicious code into a third-party script and stole payment details from 380,000 customers. In financial services, BigBasket's **2020 breach** exposed data of over 20 million users due to a compromised third-party provider. Meanwhile, in healthcare, SingHealth's systems **were infiltrated** via a compromised workstation, leaking sensitive data of 1.5 million patients. Each case highlights the critical need for stronger oversight of third-party integrations to prevent data leakage.

380,000
British Airways Magecart victims

20 million
BigBasket users' data leaked

1.5 million
SingHealth patient's data leaked

The Most Pressing Client-Side Risks Today

JavaScript client-side innovation introduces four new risks for online businesses that rely on it as the cornerstone of their fast-paced business innovations. These risks include emerging security threats such as intellectual property (IP) theft, data leakage, and recent regulatory changes and demands. Understanding and addressing these challenges is essential for businesses aiming to innovate securely and sustainably. Let's examine each one in more detail.

New Security Threats

Websites are vulnerable to attacks

Payment data is a primary target

Threat actors compromise JavaScript to steal payment data

Client-Side digital innovation has introduced a new wave of security threats. The transparent nature of JavaScript, visible in any web browser, has created vulnerabilities exploited by malicious actors. Hackers can tamper with a website's JavaScript to modify its behavior and steal sensitive information like credit card details or valuable content such as streaming audio or video files. Hackers are using both first and third-party scripts as anchoring points for their attacks.

Businesses are increasingly being targeted by **skimming attacks**. There's also a rising tide of **supply-chain attacks** where the JavaScript of a third-party add-on is compromised, and all its downstream users suddenly face the risk of data theft. As businesses become increasingly reliant on client-side JavaScript development, JavaScript's weaknesses and client-side blind spots are being exploited. And this trend is only going to intensify as AI now powers a new generation of attacks, making them more sophisticated, insidious, and harder to detect than ever before.

In January 2025, a web skimming attack compromised multiple websites, including Casio UK. Seventeen sites were confirmed affected, with the number potentially rising as investigations continue. The breaches likely exploited vulnerabilities in Magento-based web stores, a common target for such attacks. Web skimming involves injecting malicious code into websites to steal sensitive user information during transactions.

Source: Jscrambler

17

Affected websites

Be in the Know: Common Client-Side Attacks

Digital Skimming and Magecart Attacks

Digital and Magecart attacks steal sensitive data (like credit card info) by injecting malicious JavaScript into websites, especially on payment pages. These scripts often mimic legitimate code, run undetected for long periods, and exploit third-party services.

Web Supply Chain Attacks

These attacks target third-party scripts used by websites. Hackers inject malicious code into trusted vendor components, compromising multiple sites at once and bypassing traditional defenses.

Pixel Data Exfiltration

Malicious JavaScript can hijack invisible tracking pixels to send sensitive user data (PII, behavior, interactions) to external servers—undetected and without user consent.

PII Harvesting

Attackers exploit JavaScript vulnerabilities to steal personal data from forms and cookies, including names, emails, and credit card info, leading to identity theft and fraud.

Customer Journey Hijacking

Via browser extensions, attackers inject pop-ups with coupons or competitor ads during shopping sessions, redirecting users away and hijacking conversions.

Iframe Hijacking

Attackers manipulate iframes, functions, or scripts in payment flows, using techniques like function hijacking and formjacking to steal payment or user data.

New Risks of Intellectual Property (IP) Theft

Competitors are also taking full advantage of this situation, freely exploring and stealing exposed JavaScript innovations. With all JavaScript laid out in the open, carefully crafted client-side web pages become a supermarket aisle. Here, the competition can casually stroll through, eyeing JavaScript innovations, and pick out the best elements they wish to replicate. Innovative digital offerings are on display, ready for the taking, putting your competitive edge at risk. JavaScript also serves as a means to stream audio and video files on the internet. If the JavaScript used in media streaming is compromised, it allows hackers to steal the media files.

Recently, Synopsys, a prominent company in electronic design automation (EDA) tools, filed a lawsuit against Real Intent, accusing them of unlawfully using its proprietary software code. According to the claims, Real Intent integrated Synopsys's code into its own products, leading to a jury ruling in favor of Synopsys and awarding $546,000 in damages. This case highlights the significance of safeguarding proprietary code and the legal risks associated with its unauthorized use.

Source: Jumatic

$546K
IN DAMAGES

New Data Leakage Risks

Online "partners", the third-party JavaScript solutions you implement on your web pages, are also feasting on the data collected from your client-side interactions. Why? Because their AI-powered products are insatiable. They thrive on this data consumption frenzy, constantly feeding off it to enhance their services. But here's the twist in the story: the data they're devouring without asking – it's not just any data. It's yours. It's your customers'. It's the data you and your customers thought was private, secure, and protected. It's being consumed, used, and processed, and all this is happening often without explicit permission. As third-party website add-on companies aim to collect more proprietary and sensitive data, the risk of data leaks and their severe consequences significantly rises.

In August 2022, Novant Health, a U.S.-based healthcare provider, revealed that a data breach had compromised the personal information of more than 1.3 million individuals. The incident stemmed from an incorrect configuration of the Meta Pixel tracking tool, which unintentionally sent sensitive patient details, such as names, contact information, medical appointments, and health records, to Facebook.

Source: BleepingComputer

1.3M
individuals afected

New Standards and Regulatory Challenges

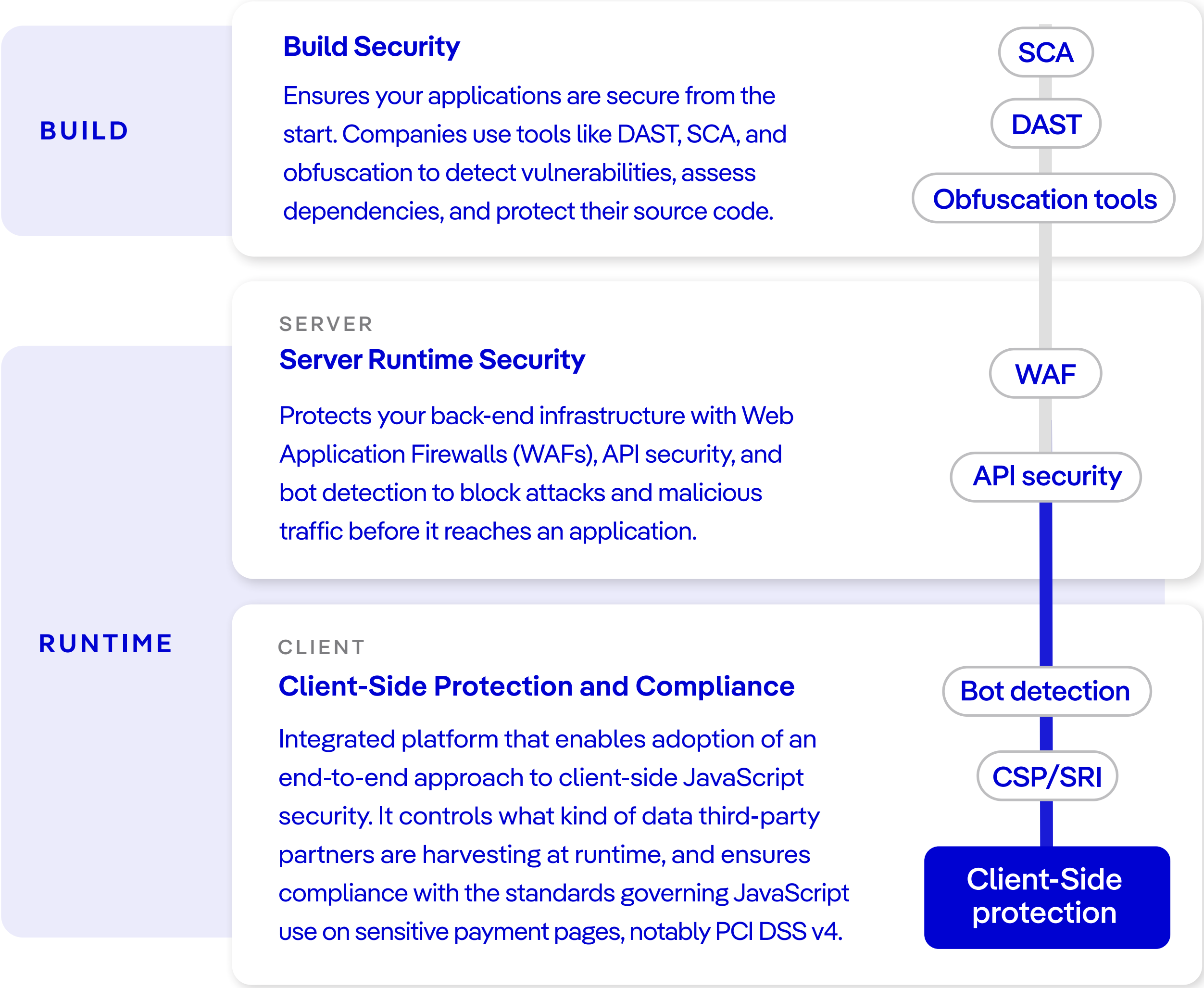
The universal usage of first and third-party JavaScript isn't just a trend; it's creating a perilously exposed client-side environment. The situation has become so critical that regulatory bodies are stepping in, setting new industry standards, and imposing stringent security requirements on e-commerce web applications.

Notably, the PCI Security Standards Council (PCI SSC), a global forum dedicated to establishing data security standards for secure payments on a global scale, unveiled PCI DSS v4 in 2022. This standard provides specific guidelines that mandate online merchants gain visibility, risk management capabilities, and control over the use of JavaScript on their payment pages. By April 1, 2025, merchants must become compliant, regardless of when their assessment takes place. If a merchant has a security incident between the effective date and their next assessment, they will be liable.

The new PCI DSS v4 standard requires e-commerce companies to employ measures to protect the payment pages on their websites against JavaScript skimming attacks. It needs to be implemented by **April 1st, 2025.** There are two requirements, 6.4.3 and 11.6.1, that are specifically designed to protect payment pages of websites that capture payment card data. The recent **SAQ A changes** only emphasize the importance of client-side security.

Client-Side Protection in the Web Application Security Ecosystem

Where is client-side protection in the web application ecosystem? There are many server-side solutions and tools that are usual items on the security tech checklist. There are also transitional solutions that sit between server-side and client-side. The need for purely client-side solutions grows as **skimming attacks** and JavaScript exploits become more common.



BUILD

Build Security

Ensures your applications are secure from the start. Companies use tools like DAST, SCA, and obfuscation to detect vulnerabilities, assess dependencies, and protect their source code.

RUNTIME

SERVER

Server Runtime Security

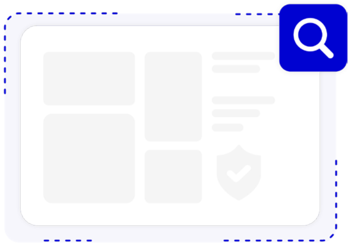
Protects your back-end infrastructure with Web Application Firewalls (WAFs), API security, and bot detection to block attacks and malicious traffic before it reaches an application.

CLIENT

Client-Side Protection and Compliance

Integrated platform that enables adoption of an end-to-end approach to client-side JavaScript security. It controls what kind of data third-party partners are harvesting at runtime, and ensures compliance with the standards governing JavaScript use on sensitive payment pages, notably PCI DSS v4.

Limitations of Known Tools and Solutions



Businesses have experimented with different approaches, such as developing in-house solutions, utilizing open-source security solutions, and implementing client-side protection modules within broader security suites.

In-House Solutions: Efforts to create internal script-protection solutions have often failed due to complexity and a lack of specialized knowledge.

Open-Source Solutions: Incompatible with the high stakes of client-side innovation (high probability of breaking the code and insufficient parameter granularity).

Non-Specialized Security Suites: The client-side protection solutions offered by CDNs, WAFs or the modules offered by Web App and API Protection (WAAP) security suites that are not dedicated to client-side protection are insufficient for both first- and third-party JavaScript protection.

- **CDNs (Content Delivery Networks):** CDN services aim to significantly enhance website performance and reliability by reducing latency through content delivery from the nearest location to the user. In addition, CDNs usually offer various web and application security capabilities, including DDoS protection and web application firewalls (WAFs). Because of that, CDNs often adopt a one-size-fits-all approach, prioritizing extensive coverage and incorporating client-side security as an add-on to their technology. While such complementary offerings might be suitable for smaller companies, enterprise-grade businesses require the depth, detail, and customization that a dedicated platform provides.

- **WAFs (Web Application Firewalls):** Those companies primarily focus on filtering and monitoring traffic, offering some level of third-party script data monitoring, but lack code obfuscation capabilities and RASP features, making them ill equipped to protect against threats that target client-side code in web applications, such as code tampering, reverse engineering, skimming and Magecart attacks.

CSP (Content Security Policy) and SRI (Subresource Integrity): CSP and SRI provide layers of security, but are not sufficient for comprehensive client-side JavaScript protection, particularly for first- and third-party JavaScript. CSP helps control sources from which resources can be loaded, but they don't address issues like unusual script behavior or unauthorized code tampering. It also lacks control over the code that is executed in the browser and requires extensive manual configuration and maintenance. SRI ensures that resources fetched from external servers haven't been tampered with, but they don't monitor or protect against runtime behavior changes or attacks that originate from trusted third parties. As such, while CSP and SRI are valuable, they must be complemented with more advanced solutions that can monitor and manage script behavior and integrity in real-time.

Your Client-Side Protection Team

When exploring a client-side protection solution, it's important to consider the different teams and roles within your organization that will be involved in evaluating, managing, and using it. Key stakeholders will benefit greatly from the proper cross-team selection of a client-side security platform. Business owners will issue the primary requirements of the solution. Technical evaluators will test and assess the product's capabilities. And finally, end users will be responsible for implementing and working with the solution on a regular basis.

Compliance

Leads compliance and risk operations

BUSINESS OWNER END USER

Titles: Head of Governance Risk & Compliance, PCI Compliance Manager, ISA, Information Security Risk & Compliance Manager

Role in the selection process: This experienced GRC leader is typically the primary decision-maker and budget holder for compliance solutions, driven by mandates like PCI DSS v4 or QSA recommendations. They collaborate with the CISO and CTO to evaluate solutions for compliance, automation, and usability, with final approval from senior leadership.

Security

Oversees all security operations

BUSINESS OWNER

Titles: Chief Information Security Officer (CISO), Chief Information Officer (CIO), Head of Information Security, Web Security, Security Architect

Role in the selection process: The CISO leads the decision-making for client-side protection, balancing security priorities with budget and executive sign-off. They rely on technical teams to validate performance and compatibility before championing the solution to leadership.

Development

Implements solutions and accelerates development

TECH EVALUATOR END USER

Titles: Leader of Application Development, Product Development, Software Engineer, Head of Engineering, Head of Software Engineering

Role in the selection process: Prompted by PCI DSS v4 needs or failed tests, this professional seeks client-side security to protect IP and prevent code abuse. While not always the budget owner, they assess vendors and advise technical leadership on performance and complexity concerns.

Line of Business

Leads digital transformation

BUSINESS OWNER

Titles: Head of Digital Business Unit, Chief Digital Officer, VP of Digital Operations, Digital Transformation Leader, Head of Digital Business Operations, Head of Product

Role in the selection process: This professional selects security tools that align with digital transformation and user experience goals, often driven by audits, SDLC needs, or breaches. While innovation-focused, they must consult IT and leadership when budget authority is limited and may hesitate over cost, performance, or third-party trust.

The Value of a Client-Side Solution For Each Stakeholder

Your cross-functional selection team will gain the following benefits from investing in the right unified client-side protection solution. Ensure each of these value areas are identified as part of your team's decision-making process.

Fast-Track PCI DSS Compliance

✓ Get a consolidated, real-time overview of vendors, scripts, and data.

✓ Meet new PCI DSS v4 standards and requirements easily.

✓ Save time and resolve cases faster by automating manual checks.

Compliance

Prevent Digital Skimming Attacks

✓ Identify and prevent Magecart and skimming attempts.

✓ Protect sensitive payment data while allowing the business to operate.

✓ Gain more payment page control without increasing staff demands.

Security

Block Consumer Data Leakage

✓ Securely enable marketing, analytics, and payment tag workflows.

✓ Block third-party tags from accessing sensitive data and prevent data leaks.

✓ Protect the privacy of consumers within your industry.

Security

Stop IP Theft & Enforce Software Licensing

✓ Protect your code from tampering, reverse engineering, and IP theft.

✓ Ensure your code is not modified to circumvent licensing restrictions.

✓ Maintain full app integrity and performance.

Development

Innovate Securely

✓ Innovate at the speed the market dictates while remaining secure by design.

✓ Ensure full visibility and eliminate client-side security blind spots.

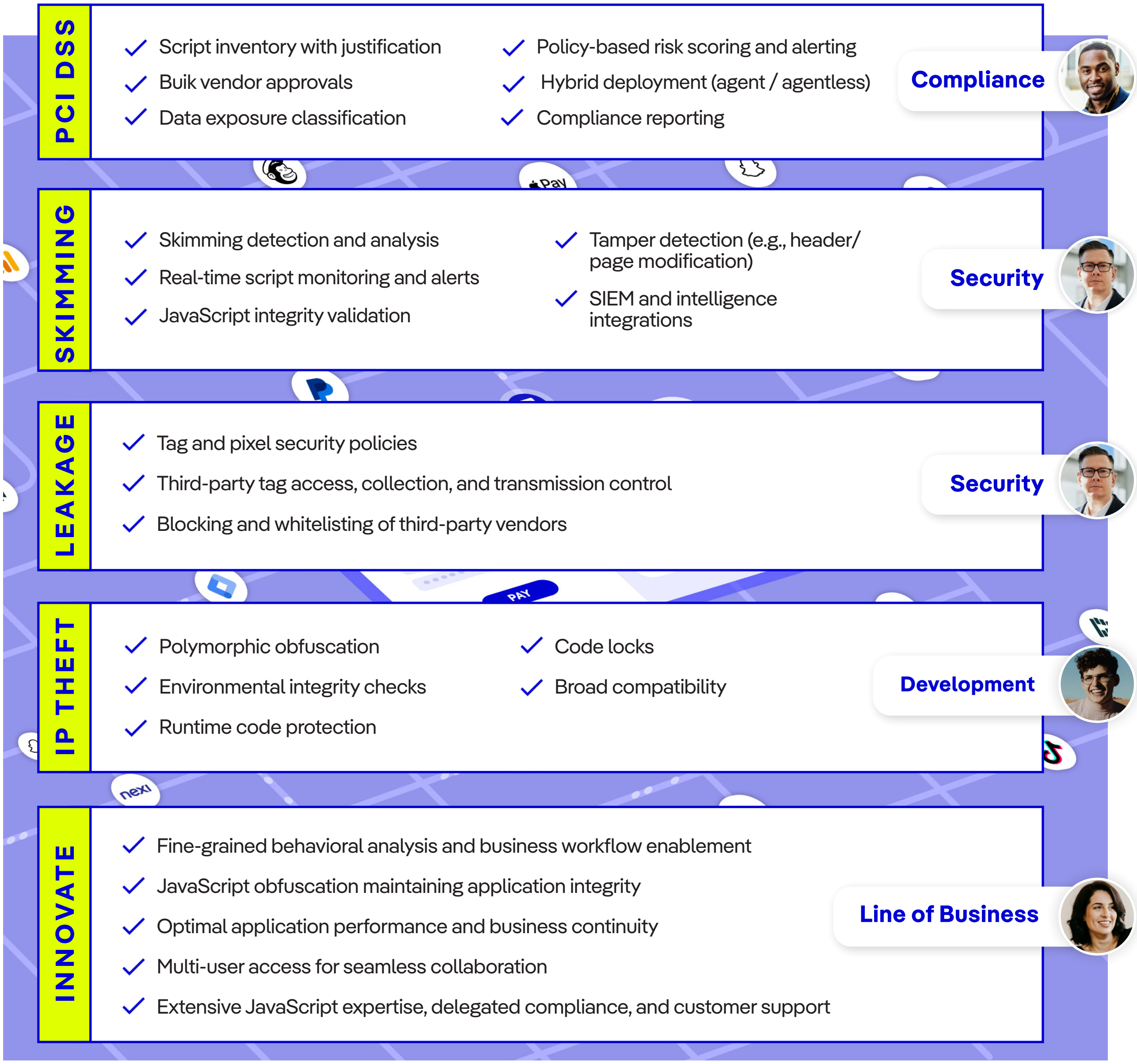
✓ Implement advanced client-side security without additional resources.

Line of Business

The Essential Capabilities of a Client-Side Protection and Compliance Solution

Businesses require a single integrated platform that enables them to adopt an **end-to-end approach** to client-side JavaScript security.

The ideal client-side protection and compliance platform should be entirely dedicated to safeguarding data and ensuring compliance for client-side innovation. It should provide a comprehensive solution with these key capabilities offered within the core platform in support of fast-tracking PCI compliance, preventing skimming attacks, blocking consumer data leakage, stopping IP theft, and innovating securely.



PCI DSS

- ✓ Script inventory with justification
- ✓ Buik vendor approvals
- ✓ Data exposure classification
- ✓ Policy-based risk scoring and alerting
- ✓ Hybrid deployment (agent / agentless)
- ✓ Compliance reporting

Compliance

SKIMMING

- ✓ Skimming detection and analysis
- ✓ Real-time script monitoring and alerts
- ✓ JavaScript integrity validation
- ✓ Tamper detection (e.g., header/ page modification)
- ✓ SIEM and intelligence integrations

Security

LEAKAGE

- ✓ Tag and pixel security policies
- ✓ Third-party tag access, collection, and transmission control
- ✓ Blocking and whitelisting of third-party vendors

Security

IP THEFT

- ✓ Polymorphic obfuscation
- ✓ Environmental integrity checks
- ✓ Runtime code protection
- ✓ Code locks
- ✓ Broad compatibility

Development

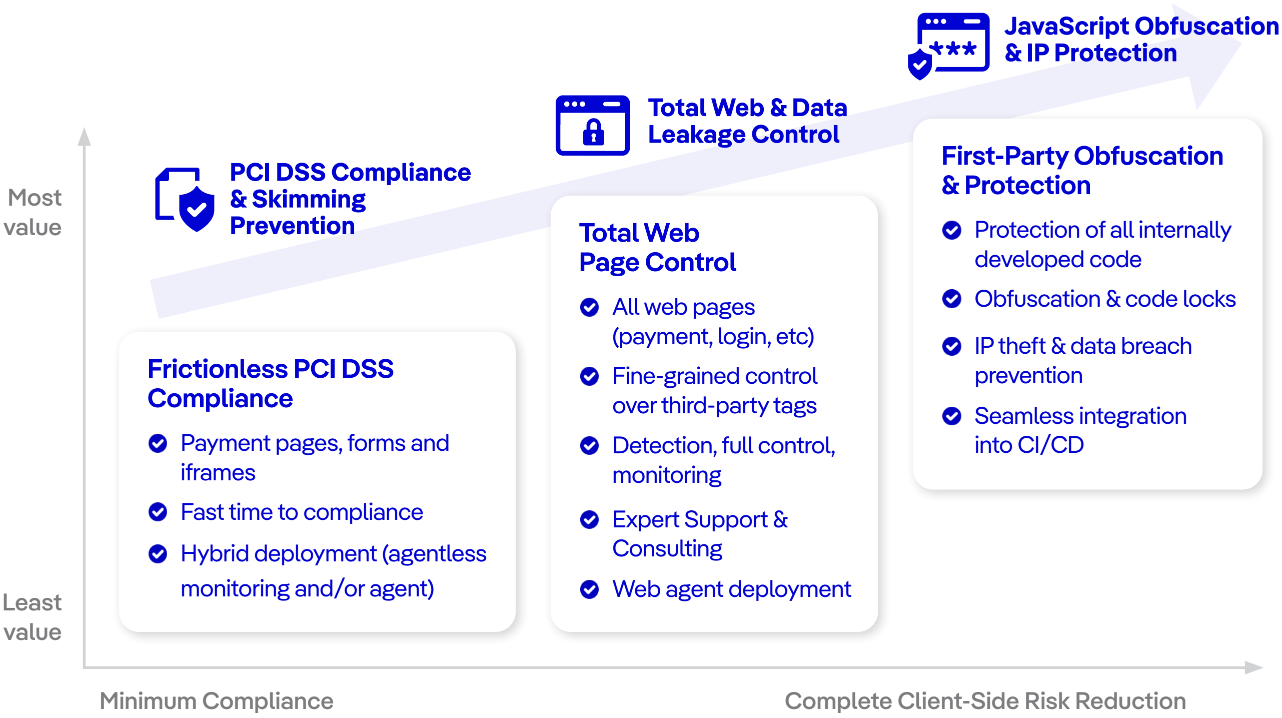
INNOVATE

- ✓ Fine-grained behavioral analysis and business workflow enablement
- ✓ JavaScript obfuscation maintaining application integrity
- ✓ Optimal application performance and business continuity
- ✓ Multi-user access for seamless collaboration
- ✓ Extensive JavaScript expertise, delegated compliance, and customer support

Line of Business

How to Start with Client-Side Protection

Client-Side protection is essential for modern businesses aiming to deliver secure and seamless digital experiences. Implementing robust client-side security measures ensures that all scripts, both first and third-party, are monitored and controlled, safeguarding user data and maintaining compliance with regulatory standards. By adopting a comprehensive client-side protection strategy, organizations can confidently innovate and enhance user experiences without compromising security.


Most value
Least value
Minimum Compliance
Complete Client-Side Risk Reduction
PCI DSS Compliance & Skimming Prevention
Total Web & Data Leakage Control
JavaScript Obfuscation & IP Protection

Frictionless PCI DSS Compliance

- Payment pages, forms and iframes
- Fast time to compliance
- Hybrid deployment (agentless monitoring and/or agent)

Total Web Page Control

- All web pages (payment, login, etc)
- Fine-grained control over third-party tags
- Detection, full control, monitoring
- Expert Support & Consulting
- Web agent deployment

First-Party Obfuscation & Protection

- Protection of all internally developed code
- Obfuscation & code locks
- IP theft & data breach prevention
- Seamless integration into CI/CD

The maturity table above outlines the progressive stages of client-side protection, starting from basic compliance to achieving full control and maximum security across your web application(s). As businesses advance in maturity, they move beyond simply meeting minimum requirements, such as protecting payment pages for PCI DSS compliance, and adopt more comprehensive measures like third-party tag control across all web pages, and first-party obfuscation to protect all exposed JavaScript from IP theft. At the highest level, organizations gain total web page control, enabling real-time detection, risk mitigation, and expert-driven compliance. This journey ensures not only regulatory alignment but also robust defense against modern client-side threats.

Client-Side Protection Checklists

Compliance Checklist for PCI DSS v4

To comply with PCI DSS v4, organizations must enhance visibility and control over JavaScript on payment pages. This includes a structured approach to inventory, governance, and protection. Here's a condensed guide to help assess your current state and align with the new standards:

1. Inventory JavaScript on Payment Pages

- ✓ List all scripts on payment pages with names, URLs, and their functions.
- ✓ Categorize as first- or third-party and identify responsible teams
- ✓ Evaluate whether each script is necessary.

2. Understand Current JavaScript Management Practices

- ✓ Review who controls script changes and how changes are approved.
- ✓ Check if scripts undergo integrity validation and third-party risk assessments.
- ✓ Determine if scripts are deployed manually or via automation and where they apply.

3. Develop a Change Control Workflow

- ✓ Create a documented process for reviewing and approving scripts.
- ✓ Define roles and track all script changes with approvals and integrity details.
- ✓ Keep your script inventory updated and aligned with your release cycle.

4. Implement Protective Controls

- ✓ Go beyond CSP and SRI given its limited functionality and resources needed.
- ✓ Implement client-side protection to automate visibility, monitoring, and control.

5. Choose Script Approval Timing

- ✓ Pre-deployment approval offers proactive security.
- ✓ Post-deployment approval may better suit dynamic environments.

Following these steps will help ensure compliance with PCI DSS v4 and strengthen client-side security on payment pages. If you'd like to see the full checklist, check out this **Buyer's Guide to PCI DSS Compliance**.

JavaScript Protection Checklist for Third-Party Code

Because data leakage prevention is a complex topic and mainstream security solutions aren't capable of preventing skimming attacks, it's important that companies know how to properly assess a security product to mitigate web skimmers. So, this checklist recommends technical tests that should be performed when testing such a product, as well as important technical requirements to consider when procuring a vendor.

Technical Tests

- ✓ **Block malicious event handlers:** Detect attempts to attach fake click or form-related events (e.g., onmouseover) used to steal input data.
- ✓ **Block unauthorized DOM changes:** Prevent the addition or removal of page elements like fake forms or buttons that mislead users.
- ✓ **Detect content manipulation:** Stop attackers from altering attributes or hiding elements to trick users (e.g., hiding spinners or messages).
- ✓ **Monitor data exfiltration:** Identify and block attempts to send stolen data to suspicious or unauthorized external domains.

Technical Requirements

- ✓ **Full website inventory:** Maintain visibility over all scripts and network activity to detect anomalies.
- ✓ **Behavior-based detection:** Choose tools that analyze behavior, not just signatures, to catch unknown threats.
- ✓ **Broad compatibility:** Ensure the solution works across all major browsers and devices.
- ✓ **Low impact on performance:** The solution should be lightweight to preserve page speed and user experience.
- ✓ **Easy integration:** Select tools that are simple to deploy and don't require major changes to your website.
- ✓ **Tamper-resistant code:** Protection logic must be resilient and able to run securely even alongside potentially malicious scripts.

By applying these checks, businesses can better protect their web applications from third-party risks without compromising performance or usability.

JavaScript Protection Checklist for First-Party Code

JavaScript is a prime target for attackers looking to steal data, intellectual property, or exploit vulnerabilities. Protecting your first-party code helps prevent tampering, reverse engineering, and unauthorized access. Security standards like ISO 27001 and OWASP also recommend code protection to reduce risks and maintain integrity.

Because JavaScript protection is a complex topic, it's important that companies are aware of the different levels of protection and effectiveness when assessing a product. Client-Side protection should be approached in three key layers:

JavaScript Obfuscation

✓ Transform readable code into a version that's difficult to understand or reverse-engineer. Effective obfuscation includes renaming identifiers, hiding logic with control flow changes, injecting dead code, and encoding sensitive data. Each protection run should produce unique, unpredictable code.

Data and Code Integrity

✓ Lock code to specific environments (e.g., browsers, domains, timeframes) to prevent it from running in unauthorized conditions. Embed hidden integrity checks throughout the app and protect critical APIs to ensure code hasn't been tampered with.

Runtime Protection

✓ Stop attackers in real-time by detecting debuggers, tampering, or emulators. Respond automatically by breaking the app, sending alerts, or triggering custom actions like clearing cookies or terminating sessions.

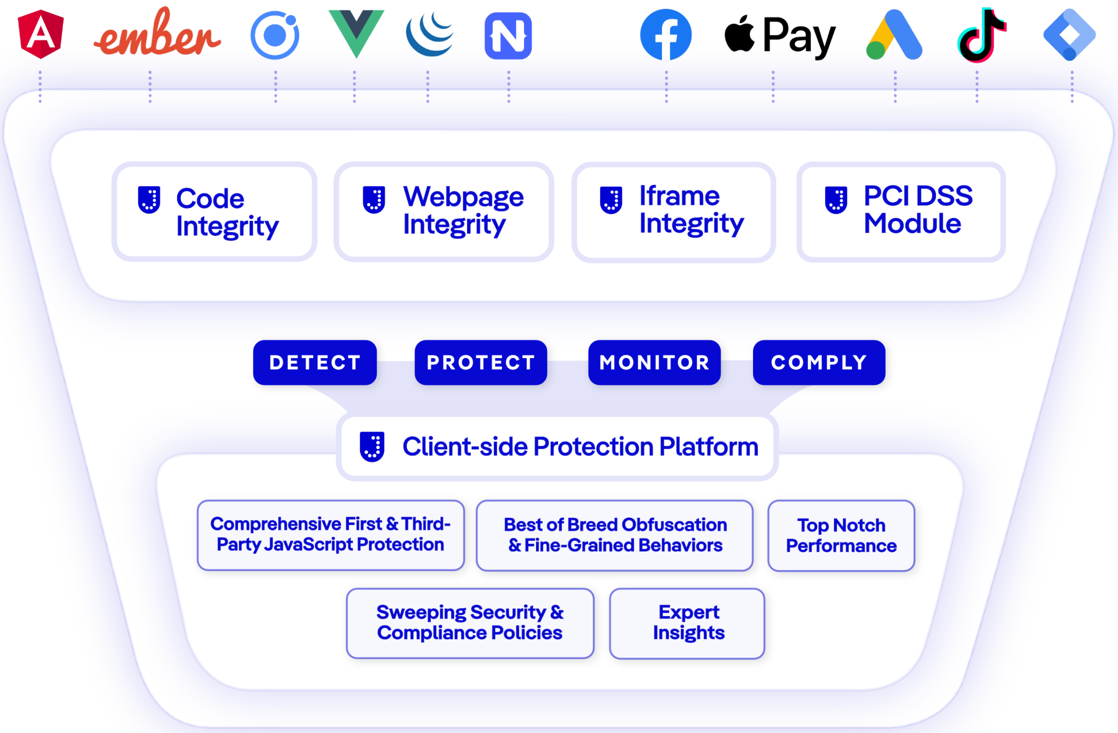
By combining these layers, businesses can make it significantly harder—and less rewarding—for attackers to exploit their JavaScript.

Jscrambler: Pioneer and Leader in Client-Side Protection and Compliance

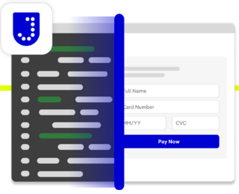
Jscrambler is the leader in Client-Side Protection and Compliance. We were the first to merge advanced polymorphic JavaScript obfuscation with fine-grained third-party tag protection in a unified Client-Side Protection and Compliance Platform.

Our end-to-end solution does more than protect your data—it empowers your business. With Jscrambler, your teams are free to take full advantage of client-side JavaScript, assured that your business benefits from sweeping protection against current and emerging cyber threats, data leaks, misconfigurations, and IP theft. Jscrambler is the only solution that enables the definition and enforcement of a single, future-proof security policy for client-side protection. We also make it easy to comply with new standards and regulations; our dedicated PCI module is designed specifically to help businesses meet the stringent new PCI DSS v4 requirements.

Trusted by digital leaders including top Fortune 500 companies, online retailers, airlines, media outlets, and financial services firms, Jscrambler lets you move fast and embrace a culture of fearless digital innovation, backed by the assurance that both your first- and third-party client-side JavaScript assets will remain secure and compliant.


ember
Pay
Code Integrity
Webpage Integrity
Iframe Integrity
PCI DSS Module
DETECT
PROTECT
MONITOR
COMPLY
Client-side Protection Platform
Comprehensive First & Third-Party JavaScript Protection
Best of Breed Obfuscation & Fine-Grained Behaviors
Top Notch Performance
Sweeping Security & Compliance Policies
Expert Insights

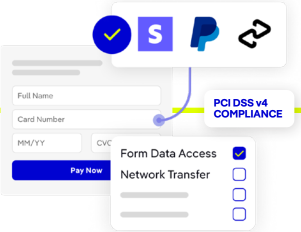
About Jscrambler products



Code Integrity

Future-Proof Protection Through Obfuccation

Protect your first-party code from tampering, debugging, reverse engineering, and more.



Webpage Integrity

Gain Fine-Grained Control Over Third-Party Tags

Ensure third-party scripts in your application run as intented and are free from fraud attacks

Iframe Integrity

Enabling PSPs to Deliver Seamless Compliance to Merchants

Support merchants PCI DSS compliance and SAQ A eligibility by protecting **payment pages** against skimming attacks — in a controlled, tamper-proof approach.

PCI Security Standards Council
PRINCIPAL PARTICIPATING ORGANIZATION

PCI DSS Module

Jscrambler Webpage Integrity offers a dedicated module designed to assist online businesses in meeting the rigorous requirements of PCI DSS 4. The PCI DSS module was developed specifically to help customers address the two new anti-skimming requirements of the PCI DSS v4, 6.4.3 and 11.6.1.

Key Features

- ✓ **Script Authorization (6.4.3):** Full visibility and control over payment page scripts to ensure integrity and compliance.
- ✓ **Real-Time Detection & Alerts (11.6.1):** Detects unauthorized script changes and HTTP header tampering with instant alerts.
- ✓ **Compliance Reporting:** Generates assessment-ready reports for QSAs.
- ✓ **Hybrid Architecture:** Supports both agentless and agent-based deployment.
- ✓ **Delegated Compliance:** Offloads PCI DSS v4 tasks to Jscrambler, reducing internal workload.

2025 WINNER CYBER SECURITY EXCELLENCE AWARDS

SPRING 2025 Highest User Adoption

Highest User Adoption WINTER 2025

BIG INNOVATION 2024

2025 GLOBEE AWARDS GOLD WINNER CYBERSECURITY

GLOBAL INFOSEC AWARDS WINNER CYBER DEFENSE MAGAZINE 2025

TOP INFOSEC INNOVATOR WINNER CYBER DEFENSE MAGAZINE 2024

Gartner HYPE CYCLE FOR APPLICATION SECURITY 2024

Book a Demo