Payment Fraud Trends in Higher Ed and How to Stop Them at the Browser

October 22, 2025

Matt McGuirk
Head of Field Engineering
Source Defense

CAMPUSGUARD®

ScriptSafe

Web Application Client-side Security

JavaScript: The Root Cause of Client-side Security Problems

source DEFENSE

Client-Side security, specifically eSkimming prevention starts with JavaScript

JavaScript can perform any action and behave in any way it wants on any site where it is running (over 97% of all websites)

JavaScript has no native security controls

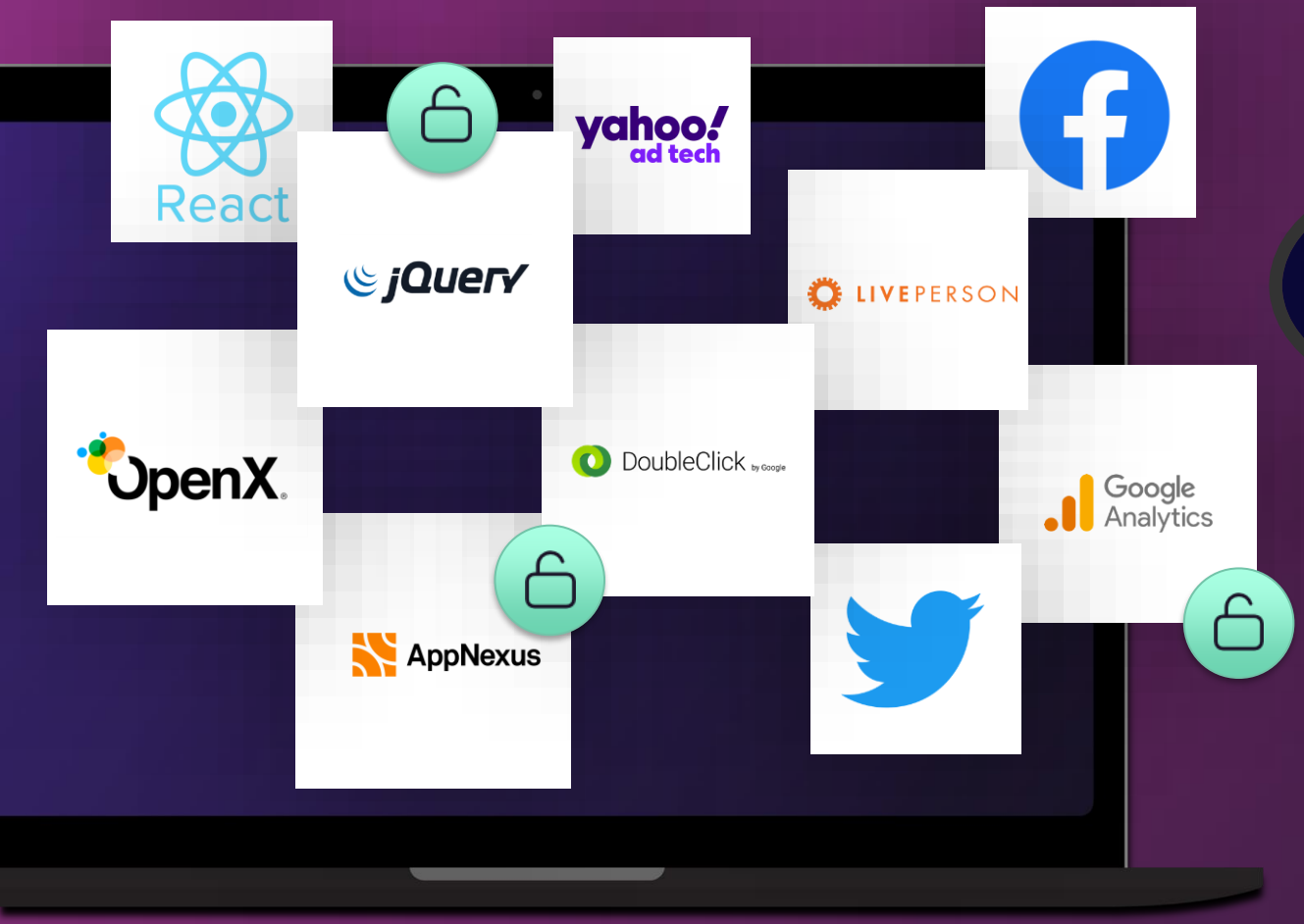
Attacks that use JavaScript

- Magecart (brand name)
- eSkimming (PCI terminology)
- Formjacking, Credential Harvesting, etc.

Protecting Data at the Point of Input

A New Challenge for the PCI Community

source DEFENSE

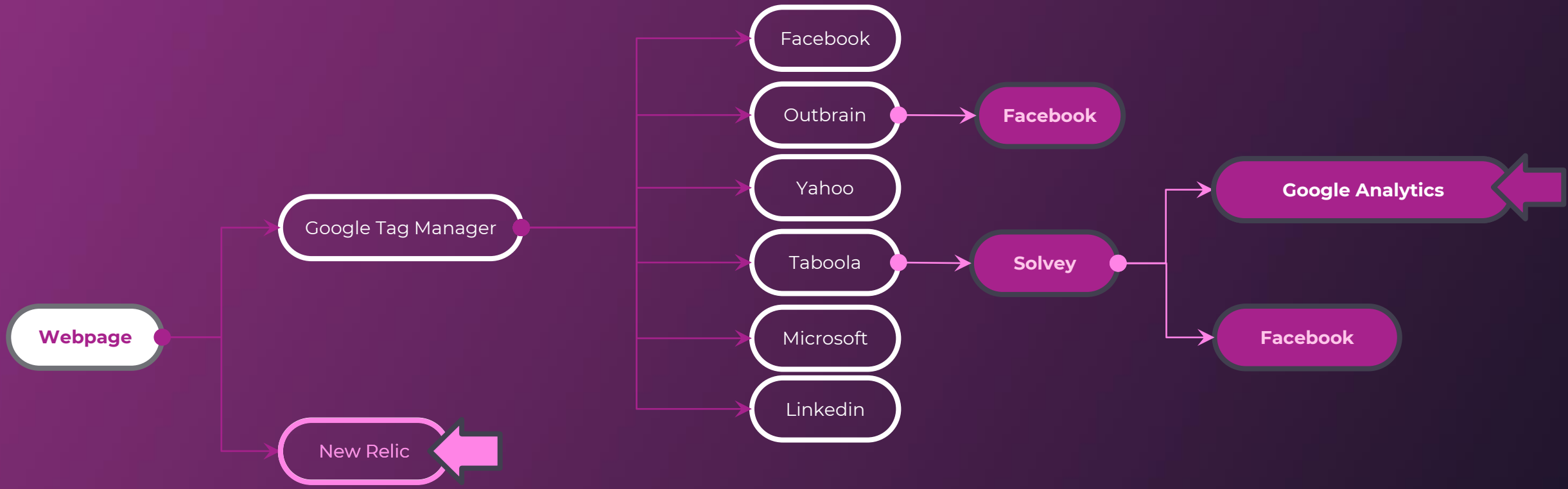


82%

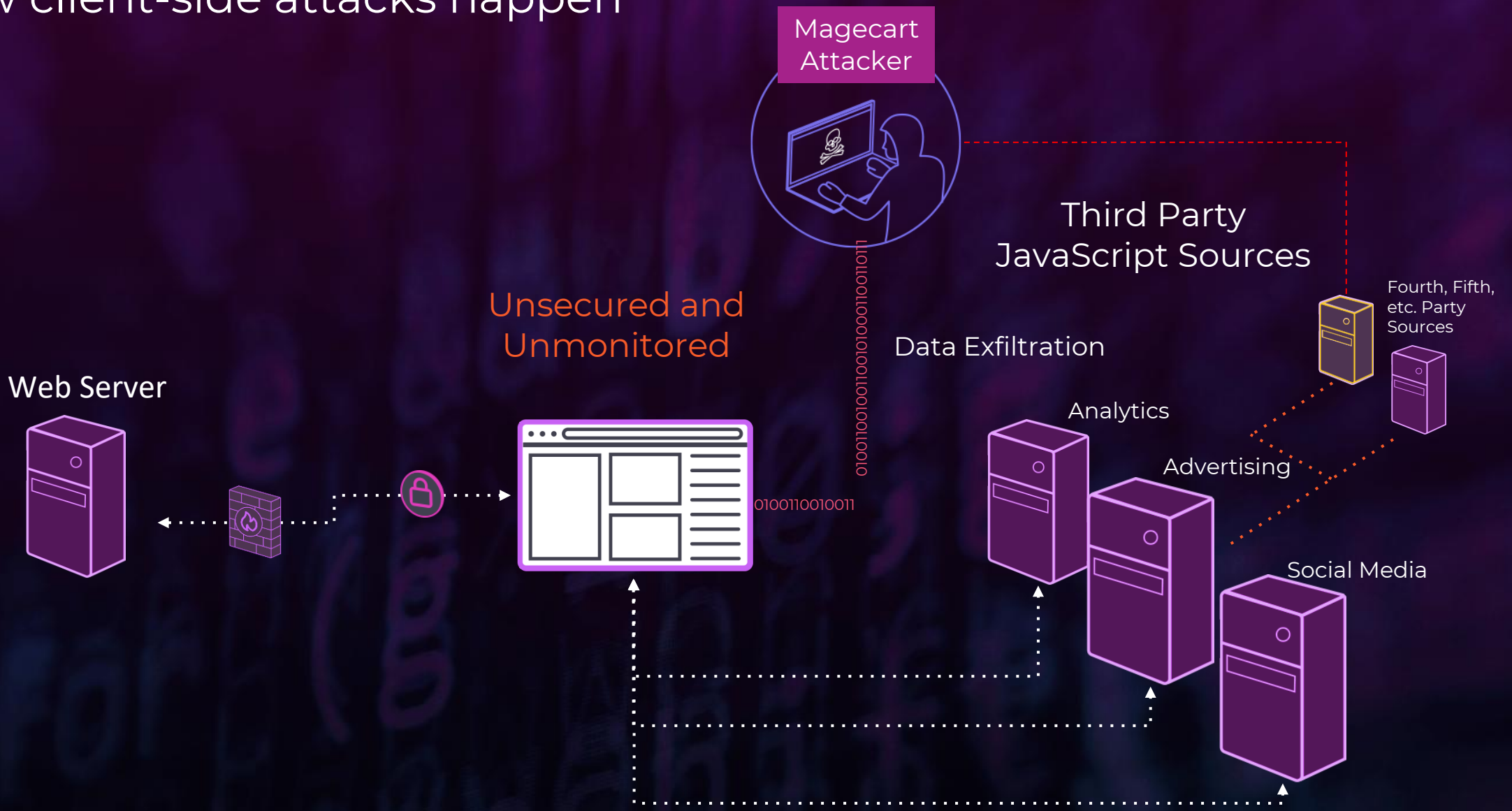
of code and actions
are outside your control

What is a client-side supply chain attack?

source DEFENSE



How client-side attacks happen



PCI DSS 4.0

source DEFENSE

PCI DSS v4.0
Requirement 6.4.3

Requirements and Testing Procedures	
Defined Approach Requirements	**Defined Approach Testing Procedures**
6.4.3 All payment page scripts that are loaded and executed in the consumer's browser are managed as follows: • A method is implemented to confirm that each script is authorized. • A method is implemented to assure the integrity of each script. • An inventory of all scripts is maintained with written business or technical justification as to why each is necessary.	**6.4.3.a** Examine policies and procedures to verify that processes are defined for managing all payment page scripts that are loaded and executed in the consumer's browser, in accordance with all elements specified in this requirement.
	6.4.3.b Interview responsible personnel and examine inventory records and system configurations to verify that all payment page scripts that are loaded and executed in the consumer's browser are managed in accordance with all elements specified in this requirement.
Customized Approach Objective	
Unauthorized code cannot be executed in the payment page as it is rendered in the consumer's browser.	

1. **Inventory payment page scripts**
2. **Confirm authorization and provide a justification for each payment page script**
3. **Implement a method to assure the integrity of each payment page script**

source DEFENSE

PCI DSS v4.0
Requirement 11.6.1

Requirements and Testing Procedures	
11.6 Unauthorized changes on payment pages are detected and responded to.	
Defined Approach Requirements	**Defined Approach Testing Procedures**
11.6.1 A change- and tamper-detection mechanism is deployed as follows: • To alert personnel to unauthorized modification (including indicators of compromise, changes, additions, and deletions) to the security-impacting HTTP headers and the script contents of payment pages as received by the consumer browser. • The mechanism is configured to evaluate the received HTTP headers and payment pages. • The mechanism functions are performed as follows: – At least weekly **OR** – Periodically (at the frequency defined in the entity's targeted risk analysis, which is performed according to all elements specified in Requirement 12.3.1).	**11.6.1.a** Examine system settings, monitored payment pages, and results from monitoring activities to verify the use of a change- and tamper-detection mechanism.
	11.6.1.b Examine configuration settings to verify the mechanism is configured in accordance with all elements specified in this requirement.
	11.6.1.c If the mechanism functions are performed at an entity-defined frequency, examine the entity's targeted risk analysis for determining the frequency to verify the risk analysis was performed in accordance with all elements specified at Requirement 12.3.1.
Customized Approach Objective	**11.6.1.d** Examine configuration settings and interview personnel to verify the mechanism functions are performed either: • At least weekly **OR** • At the frequency defined in the entity's targeted risk analysis performed for this requirement.
E-commerce skimming code or techniques cannot be added to payment pages as received by the consumer browser without a timely alert being generated. Anti-skimming measures cannot be removed from payment pages without a prompt alert being generated. *(continued on next page)*	

1. **Monitor HTTP response headers**
2. **Monitor payment page script contents**

source DEFENSE

In a nutshell

Merchants and service providers must:

1. **Inventory payment page scripts**
2. **Provide a written justification for authorized scripts**
3. **Implement a mechanism to verify the integrity of payment page scripts**
4. **Implement a mechanism to verify HTTP headers retrieved from the payment page at least once every seven days**

Common Misconceptions

source DEFENSE

“I fully redirect my checkout to my eCommerce provider…I’m good!”

“My eCommerce provider is going to block all scripts on payment pages…I’m good, good!!”

“I fully outsource payments – we segregated this environment from our website on purpose…I’m goooooood!!!”

“We only run first party scripts on our payment pages…I’m totally good!!!!!”

ScriptSafe

REAL-TIME CODE CONTROL AND SUPPLY CHAIN VISIBILITY

- **Isolate** foreign scripts from the web page
- **Block** harmful activities such as Magecart style attacks
- **Ensure** beneficial behavior
- **Thwart** attacks in real-time
- Securely **integrate** 3rd party code **with minimal effort**

30 Day Action Plan

From Zero to Compliant in 30 Days or Less

Checking the box for 6.4.3 and 11.6.1 in 30 days or less

source DEFENSE

Good news,
you're here already!

DISCOVERY

Day 1

- What do we have?
- What do we need?
- What's available?
- What's next?

EVALUATION

Day 7 through 14

- Identify teams
- Analyze
- Agree

ACQUISITION

Day 15 - 30

- Procurement
- Scheduling
- Operations

COMPLETION

Day 30-60

- Deployment
- Training
- Verification

Working with CampusGuard and Source Defense

Evaluations completed in as little as a week
(or even less)

Source Defense can create a testing environment for your payment pages in practically no time at all. Evaluations can take as little as one week from start to finish

Bespoke support for your process

Source Defense provides project-specific documentation and materials for all evaluation outcomes, commercial figures, and compliance-mapping so you can spend less time thinking about vendors

Deploy and deliver in days

Source Defense can deploy our solutions in as little as 24 hours. We provide zero-code deployment options, and code integration is as simple as copy-and-paste.