JavaScript Security Scan Report

Target URL: https://elitery.com

Scan Date: 2025-07-03 11:35:19

Scanner Tools:
• Semgrep - Static Code Analysis
• Grype - Dependency Vulnerability Scanning
• TruffleHog - Secret Detection

Executive Summary

Metric	Count
Files Scanned	66
Total Security Issues	0
High Severity Issues	0
Medium Severity Issues	0
Low Severity Issues	0
Secrets Found	9
Vulnerabilities in Dependencies	0

Risk Assessment

Overall Risk Level: CRITICAL

Secret Detection (TruffleHog)

Found 9 potential secrets in the code.

High Confidence Secrets

AWS Secret Key: ChAI*******************************cgzM
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 399)
Scanner: Custom Regex
AWS Secret Key: 0123*******************************ABCD
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 501)
Scanner: Custom Regex

Medium Confidence Secrets

AWS Secret Key: 0123*******************************ABCD
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 287)
Scanner: Custom Regex
AWS Secret Key: ABCD*******************************klmn
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 415)
Scanner: Custom Regex
AWS Secret Key: eyIw*******************************YWxz
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 444)
Scanner: Custom Regex
AWS Secret Key: 0123*******************************ABCD
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 497)
Scanner: Custom Regex
AWS Secret Key: 0123*******************************ABCD
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 498)
Scanner: Custom Regex
AWS Secret Key: 0123*******************************ABCD
File: https___www.googletagmanager.com_gtag_js_id=G-F64BXWDJZL.js
(Line 500)
Scanner: Custom Regex
AWS Secret Key: com/*******************************/vid
File: inline_1.js (Line 1)
Scanner: Custom Regex

Recommendations

1. Implement a secure code review process before deploying JavaScript code.

2. Use Content Security Policy (CSP) headers to prevent XSS attacks.

3. Regularly update all JavaScript dependencies to their latest secure versions.

4. Remove all hardcoded secrets and use environment variables or secure secret management systems.

5. Implement pre-commit hooks to prevent secrets from being committed to version control.