EnCyb

Severity: High

Advisory Type: Threat

InvisibleJS: Stealth JavaScript via Zero Width Unicode

Advisory Report

TLP: WHITE

SECURITY THREAT ADVISORY COUNCIL (STAC)

Empowering organizations with cutting-edge cybersecurity strategies to combat emerging digital threats effectively.

EXECUTIVE SUMMARY

InvisibleJS is an open-source JavaScript obfuscation tool that hides fully executable code inside files that appear visually blank by using **zero-width Unicode characters** along with **runtime decoding and execution**. The technique can bypass manual code reviews and traditional static analysis, making it attractive for stealthy malware delivery and supply-chain attacks.

- Active Region: Global
- Affected Sector: Software Development, Technology, Cybersecurity
- Affected Product: JavaScript / Node.js environments
- Severity: High
- Published Date: January 12, 2026

TECHNICAL DETAILS

- **Target:** Software supply chains, Node.js applications, JavaScript code repositories, CI/CD pipelines, and development environments relying on manual or static code review.
- **Root Cause:** Inadequate handling and inspection of zero-width Unicode characters in source code, allowing executable logic to be hidden in files that appear visually blank, combined with tooling that prioritizes syntactic validity over visual integrity.
- **Prerequisite For Exploitation:** Ability to introduce or modify JavaScript files in a project (e.g., via compromised dependencies, pull requests, or write access to repositories) and execution of the affected code in a Unicode-compliant JavaScript runtime, with insufficient runtime or pre-execution behavioral monitoring.

IMPACT

- Enables stealthy malware loaders concealed in visually blank JavaScript files
- Bypasses manual code review and weak/static code analysis controls
- Increases risk of supply-chain compromise through malicious dependencies
- Complicates incident response and forensic analysis due to hidden, non-printable payloads
- Undermines trust in source code integrity across CI/CD pipelines.

RECOMMENDATIONS

- Implement Unicode-aware static analysis to detect zero-width and non-printable characters in source code.
- Flag and investigate JavaScript files that appear blank but have non-zero file size or execution behaviour.
- Enforce strict **code review policies** and **dependency vetting** for third-party npm packages and pull requests.
- Integrate pre-execution sandboxing and automated build-time testing to identify anomalous or hidden code behaviour.
- Train developers and security teams on Unicode-based obfuscation and steganography techniques conduct periodic security scanning of repositories and dependencies for such evasion methods.

REFERENCE

- https://cyberpress.org/invisiblejs-hides-executable-es-modules/?

SECURE, SCALE, SUCCEED WITH

CONFIDENCE

EnCyb

www.encyb.com/contact-us/ EnCyb soc@encyb.com www.encyb.com