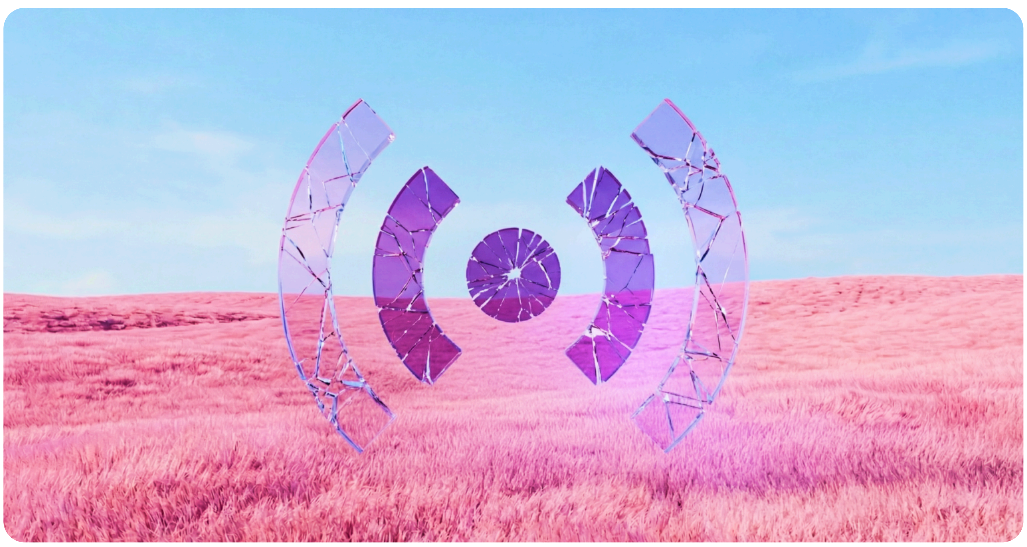
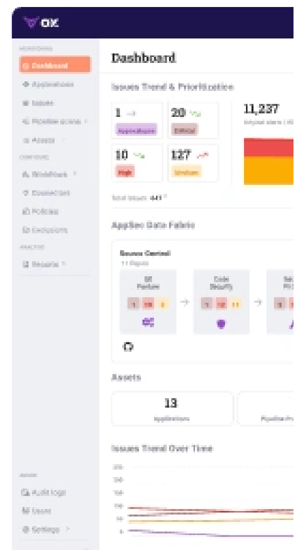
Live Server VS Code Extension Allows Remote Exfiltration of Local Files

Table of Contents

Moshe Siman Tov Bustan (https://www.ox.security/author/moshe-siman-tov-bustan/)
Nir Zadok (https://www.ox.security/author/nir-zadok/)

February 17, 2026





OX Security discovered a critical vulnerability (CVE-2025-65717) in VS Code's Live Server extension (72M+ installs) that allows attackers to exfiltrate local files by luring a developer to a malicious webpage while the extension is running. The issue was disclosed in August 2025 with no maintainer response to date.

CVE-2025-65717 — Live Server

Severity: Critical (CVSS 9.1)
IDE: VS Code
Extension: Live Server
Affected Versions: All versions of Live Server
Impact: Data exfiltration

Our Findings

Live Server is a Visual Studio Code extension that starts a local development HTTP server and automatically reloads the browser when files in the workspace change, supporting both static and dynamic pages. It provides configurable options such as the server root, port, host, default browser, proxy settings, and HTTPS. The extension also supports multiple workspace roots and watches for file changes to trigger live reloads, allowing developers to preview changes in real time without manually refreshing the browser.

We discovered a vulnerability in the Live Server extension for VS Code that allows a remote, unauthenticated attacker to exfiltrate files from a developer's local machine. Attackers only need to send a malicious link to the victim while Live Server is running in the background.

Technical Analysis

Attack Scenarios: How could this be exploited in the wild?

Stealing sensitive source code — Crawling localhost can expose proprietary code, scripts, or configuration files.

Exfiltrating credentials — Any files, including environment variables inside the .env files, containing API keys, passwords, or .env secrets could be sent to an attacker-controlled domain.

Harvesting local data — Local files, logs, or databases served by a development server can be exposed.

Attack Flow

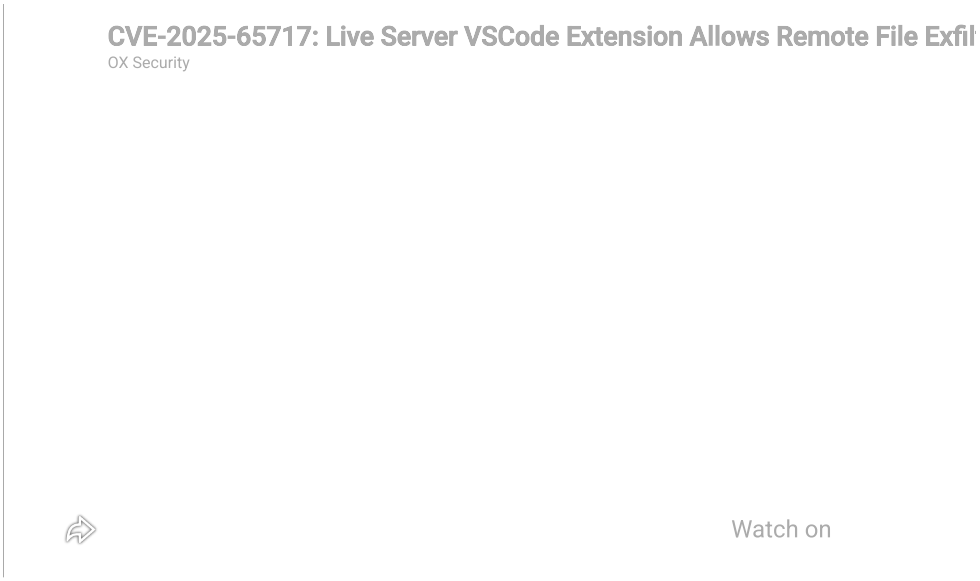
What We Discovered

When Live Server is running, and a developer opens a malicious HTML page (hosted remotely) in their default web browser, the page can use JavaScript to access http://localhost:5500 (http://localhost:5500/), allowing it to recursively crawl all files served by Live Server and exfiltrate them to an attacker-controlled domain.

Live Server doesn't implement CORS protections by default, allowing any webpage to make cross-origin requests to localhost:5500. This enables remote sites to fetch local files as if they were legitimate same-origin requests.

The following malicious HTML file was served on our server, simulating a malicious website:

Video PoC



Tags:

Vulnerability Insights(https://www.ox.security/blog/category/vulnerability-insights/)

Research(https://www.ox.security/blog/category/research/) CVE(https://www.ox.security/blog/category/cve/)

DevSecOps(https://www.ox.security/blog/category/devsecops/)

Application Security(https://www.ox.security/blog/category/application-security/)

Software Supply Chain Security(https://www.ox.security/blog/category/software-supply-chain-security/)

Related Content

April 22, 2026

Xinference allegedly hacked by TeamPCP, Malicious Package In PyPi

(https://www.ox.security/blog/xinference-allegedly-hacked-by-teampcp-malicious-

February 17, 2026

Four Vulnerabilities Expose a Massive Security Blind Spot in IDE Extensions

(https://www.ox.security/blog/four-vulnerabilities-expose-a-massive-security-blind-spot-in-ide-

December 30, 2025

900K Users Compromised: Chrome Extensions Steal ChatGPT and DeepSeek

package-in-pypi/)

By
Moshe Siman Tov Bustan (https://www.ox.security/author/moshe-siman-tov-bustan/)

extensions/)

By
Moshe Siman Tov Bustan (https://www.ox.security/author/moshe-siman-tov-bustan/) &
Nir Zadok (https://www.ox.security/author/nir-zadok/)

Conversations (https://www.ox.security/blog/malicious-chrome-extensions-steal-chatgpt-deepseek-conversations/)

By
Moshe Siman Tov Bustan (https://www.ox.security/author/moshe-siman-tov-bustan/)

Compliance (https://www.ox.security/assess-appsec-processes-ensure-compliance/)

API Exposure Management (https://www.ox.security/ox-for-api-exposure-management/)

Dev Empowerment (https://www.ox.security/eliminate-the-need-to-revisit-old-code-and-workflows-use-case/)

duck/)

vs Veracode (https://www.ox.securi veracode/)

vs Checkmarx (https://www.ox.securi checkmarx/)

vs Snyk (https://www.c vs-snyk/)

(https://www.linkedin.com/company/ox-security) (https://x.com/OX__Security) (https://www.youtube.com/@OXSecurity) (https://www.instagram.com/lifeatox/)