OWASP
Open Web Application Security Project

NODE.JS FIXES SEVERE HTTP BUG THAT COULD LET ATTACKERS CRASH APPS

Node.js has released updates for a high severity vulnerability that could be exploited by attackers to corrupt the process and cause unexpected behaviors, such as application crashes and potentially remote code execution (RCE).

- The use-after-free vulnerability, tracked as CVE-2021-22930 is to do with how HTTP2 streams are handled in the language.

This week Node.js has pushed out fixes for high severity, use-afterfree vulnerability, tracked as CVE-2021-22930.

- Use-after-free vulnerabilities occur when a program tries to access a resource at a memory address that has been previously freed and no longer holds the resource.

- This can lead to data corruption, or unexpected behaviors such as application crashes, or even remote code execution (RCE) in some cases.

- The fixes landed in the latest Node.js release 16.6.0 and were also backported to versions 12.22.4 (LTS) and 14.17.4 (LTS).

- The vulnerability was triggered in cases where Node.js parsed incoming RST_STREAM frames, with no error code or a cancel code.

- The fix rolled out instead adds the incoming stream of RST_STREAM frames to a queue and processes the queue once it is safe to do so.

This would prevent any double-free or use-after-free errors. Node.js users should upgrade to the latest version 16.6.0, or a patched backported version.