What is the Apache HTTP/2 RCE?

CVE-2026-119618 is a critical vulnerability in mod_http2, the HTTP/2 module of Apache HTTP Server. In assign_headers(), a worker thread writes into memory owned by the connection thread without proper synchronization (CWE-362), corrupting the heap and potentially allowing unauthenticated remote code execution. It affects Apache HTTP Server 2.4.55 through 2.4.70 with HTTP/2 enabled and is fixed in 2.4.71, released November 11, 2026. The only workaround is disabling HTTP/2 with Protocols http/1.1.

HOW IT WORKS

01 How the race corrupts memory

HTTP/2 multiplexes many streams over one TCP connection. mod_http2 handles that with two kinds of threads: a connection thread that owns the connection and its memory, and worker threads that process individual streams in parallel.

- Shared memory, no lock. When a worker builds a response, `assign_headers()` writes header data into memory that belongs to the connection thread, through an allocator that is not safe to use from two threads at once.
- The race. If the connection thread allocates or frees from the same memory pool while the worker is writing, the allocator's bookkeeping gets corrupted.
- From corruption to code execution. Corrupted heap metadata is a classic starting point for overwriting pointers the server later trusts. The advisory says the bug can lead to RCE; public research so far shows reproducible heap corruption under memory sanitizers, not a published working exploit.

Because the attacker controls how many streams run at once and how long each takes, the race can be pushed from rare to likely with many concurrent streams on one connection. It is the same bug class as an application-level race condition, one layer down in native memory.

02 Who is exposed

You are in scope if all of these hold:

HOW TO DEFEND

- Find out what you run. `httpd -v` (or `apache2 -v`) gives the version, and `apachectl -M | grep http2` shows whether mod_http2 is loaded. Then search your configuration for `Protocols` lines that include `h2` or `h2c`.
- Upgrade to Apache HTTP Server 2.4.71. On a distribution package, install your vendor's update rather than waiting for the upstream version number: distributions backport the fix to their own package versions, on their own schedules.
- If you cannot patch yet, turn HTTP/2 off. Set `Protocols http/1.1` server-wide and in every virtual host that sets its own `Protocols`, then restart. Confirm with `curl -sI --http2 https://your-host/` that the response comes back as HTTP/1.1. This is a workaround, not a fix, and costs some performance.
- Watch for crashes. Repeated child-process segfaults in the Apache error log, especially under bursts of HTTP/2 traffic from one client, are what failed exploitation attempts look like.

- The version. Apache HTTP Server 2.4.55 through 2.4.70, or the standalone mod_h2 module 2.0.8 through 2.0.42. Versions before 2.4.55 and mod_http2 before 2.0.8 are not affected.
- HTTP/2 is on. The `Protocols` directive lists `h2` (HTTP/2 over TLS) or `h2c` (cleartext), server-wide or in any virtual host. A server that only speaks HTTP/1.1 never runs the vulnerable code.
- An attacker can reach it. No credentials or user interaction are needed, only a connection that negotiates HTTP/2.

Apache in front of an application counts too: reverse proxies, PHP and WordPress hosts, control-panel servers and appliances that embed httpd. If a load balancer or CDN terminates HTTP/2 and talks HTTP/1.1 to Apache, the origin is not reachable over HTTP/2 from the internet, but check whether anything internal still connects to it with HTTP/2.

As of November 13, 2026, no in-the-wild exploitation had been confirmed. Crash-reproduction code is public, which usually shortens the time to a working exploit.

03 Where this fits among recent web-server flaws

CVE-2026-119618 has no catchy name, but it belongs in the same conversation as the X2Shell vulnerabilities: one bug in a component that sits under a large share of the web, reachable before any application code runs. It also echoes HTTP2Shell: both turn the machinery that parses and decodes HTTP traffic into the attack surface. The difference is the bug class. HTTP2Shell is a single out-of-bounds write in a decoder; this is a concurrency bug, which is harder to find with ordinary testing and harder to exploit reliably, but just as dangerous once someone does.

SOURCES

[1] Apache HTTP Server 2.4 vulnerabilities (CVE-2026-119618)

[2] NVD: CVE-2026-119618

[3] Red Hat CVE page: CVE-2026-119618

[4] Apache mod_http2 documentation (Protocols directive)

Find the path to a shell before an attacker does.

securelayer7.net/learn/web-exploitation/what-is-the-apache-http2-rce

Open online