Hive Pro

Threat Level

Red

Hiveforce Labs

THREAT ADVISORY

VULNERABILITY REPORT

HTTP/2 Bomb CVE-2026-49975: A Flaw Detonates Apache HTTP Server

Date of Publication	Admiralty Code	TA Number
June 12, 2026	A1	TA2026164

Summary

First Seen: June 2, 2026
Affected Products: Apache HTTP Server (httpd) with mod_http2 (HTTP/2 enabled), before mod_http2 v2.0.41
Impact: The flaw CVE-2026-49975 represents a direct and immediate threat to the availability of internet-facing web services. Because the attack abuses default HTTP/2 behavior rather than a misconfiguration, any exposed Apache HTTP Server with HTTP/2 enabled is potentially at risk, and an attacker needs only a single system on a 100 Mbps connection to render a vulnerable server inaccessible within seconds. The most damaging operational outcome is not necessarily an outright crash: an attacker can deliberately hold memory pressure just below the out-of-memory kill threshold to push the host into swap thrashing, degrading every other workload on the machine rather than triggering a clean worker respawn. With proof-of-concept code already public, the barrier to exploitation is low and the resulting outages of public websites, APIs, and gateways can carry meaningful service-disruption, reputational, and financial consequences.

CVE

CVE	NAME	AFFECTED PRODUCT	ZERO-DAY	CISA KEV	PATCH
CVE-2026-49975	HTTP/2 Bomb (Apache HTTP Server Denial-Of-Service Vulnerability)	Apache HTTP Server (mod_http2)	✗	✗	✓

Vulnerability Details

#1

A remote, unauthenticated denial-of-service technique they call the "HTTP/2 Bomb," discovered using the OpenAI Codex agent. It combines two well-known tricks into one efficient memory-exhaustion attack. The first is a compression bomb: HTTP/2's HPACK scheme lets a client reference a stored header with a single byte, which the server expands into a full header copy so one wire byte becomes one full allocation, repeated thousands of times per request. The second is a stall: the client advertises a zero-byte flow-control window so the server can never finish responding, then drips one-byte WINDOW_UPDATE frames to keep the connection alive and hold all that memory in place. The flaw exists in the default HTTP/2 configuration of most major web servers, including nginx, Apache, Microsoft IIS, Envoy, and Cloudflare Pingora.

#2

This advisory covers CVE-2026-49975, the Apache HTTP Server (mod_http2) version of the flaw, disclosed to Apache on May 27, 2026. The sibling issues in other servers received separate identifiers for example, Envoy (CVE-2026-47774) and Microsoft IIS (CVE-2026-49160) while Cloudflare says its existing architecture already blocks the attack.

#3

What makes this attack new is where the amplification comes from. Older HPACK bombs stored a large header value and referenced it repeatedly, so servers learned to cap the total decoded header size. This attack does the opposite: the referenced header is nearly empty, and the damage comes from the per-entry memory the server allocates around each reference so the size cap never triggers because there is almost nothing to decode.

#4

On Apache, the effect is especially severe because of how it handles cookies. HTTP/2 allows a cookie to be split into many small "crumbs," and Apache rebuilds the entire merged cookie string on every crumb while leaving each older copy in memory until the stream closes. Worse, those crumbs are not counted against the LimitRequestFields limit, the exact check meant to stop the original HPACK bomb. The result is roughly 4,000:1 amplification even with an empty cookie. In testing against Apache httpd 2.4.67, a single client consumed and held about 32 GB of server memory in roughly 18 seconds.

#5

There is no sign of exploitation in the wild, and no threat actor or malware is tied to this disclosure. However, a working proof-of-concept scripts on June 2, 2026, and the public fix commits reveal the technique directly, so the gap between disclosure and a usable exploit is now very short. The fix ships in mod_http2 v2.0.41, which makes cookie crumbs count against LimitRequestFields

Vulnerability

CVE ID	AFFECTED PRODUCTS	AFFECTED CPE	CWE ID
CVE-2026-49975	Apache HTTP Server (httpd) with mod_http2 and HTTP/2 enabled, prior to mod_http2 v2.0.41 (e.g., 2.4.67)	cpe:2.3:a:apache:http_server:*:*:*:*:*:*:*:*	CWE-400

Recommendations

Update mod_http2 Immediately: Upgrade to mod_http2 v2.0.41 or later, which is available from the standalone mod_http2 releases and contains the official fix that makes cookie headers count against the LimitRequestFields limit. Patching is the only complete remediation, and it should be applied as a priority to any Apache HTTP Server that terminates HTTP/2 traffic from the internet, especially given that working proof-of-concept code is already public.

Disable HTTP/2 Where You Cannot Patch Quickly: If you are unable to upgrade mod_http2 in the short term, mitigate exposure by setting "Protocols http/1.1" to disable HTTP/2 on the affected server. This removes the attack surface entirely at the cost of HTTP/2 performance benefits and should be treated as a temporary stopgap until the patched module is deployed.

Apply Partial Hardening on Unpatched Servers: Lowering LimitRequestFieldSize shrinks the per-stream blast radius by capping the merged cookie and therefore the usable crumb count, but it is only a partial mitigation because an attacker can still multiply the effect across many streams and connections. As an additional safety net, cap per-worker memory using cgroups, "ulimit -v", or container limits tight enough that a bombed worker is OOM-killed and respawned before it drags the host into swap, and ensure stalled streams have a bounded lifetime regardless of WINDOW_UPDATE activity. Note that lowering LimitRequestFields alone does nothing here, since the duplicate cookie crumbs are not counted against it on unpatched versions.

Front Public Servers with a Protective Layer: Place internet-facing Apache HTTP Server instances behind a reverse proxy, gateway, web application and API protection (WAAP) service, or Layer 7 load balancer that terminates public sessions and enforces a hard cap on the number of header fields per request, including cookie crumbs, independent of their total size. Combine this with access controls that prevent direct connections to the origin server from the internet.

Potential MITRE ATT&CK TTPs

Tactic	Technique	Sub-technique
Initial Access	T1190: Exploit Public-Facing Application	
Execution	T1059: Command and Scripting Interpreter	
Impact	T1499: Endpoint Denial of Service	T1499.003: Application Exhaustion Flood
		T1499.001: OS Exhaustion Flood

Patch Links

https://github.com/apache/httpd/commit/47d3100b252dc6668a9e46ae885242be9eeca9cd

https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git/commit/?id=49ba7b515c4c0719b866d16f068e62d16a8a3dd1

https://github.com/icing/mod_h2/releases

References

https://blog.calif.io/p/codex-discovered-a-hidden-http2-bomb

https://github.com/apache/httpd/commit/47d3100b252dc6668a9e46ae885242be9eeca9cd

https://github.com/califio/publications/tree/main/MADBugs/http2-bomb

What Next?

At **Hive Pro**, it is our mission to detect the most likely threats to your organization and to help you prevent them from happening.

Book a Demo of **HivePro**.

REPORT GENERATED ON

June 12, 2026 • 03:00 AM

More at www.hivepro.com