Investigaciones recientes han descubierto que el marco de CONTINUACIÓN en el protocolo HTTP/2 puede ser aprovechado para llevar a cabo ataques de denegación de servicio (DoS).

El investigador de seguridad Bartek Nowotarski ha denominado esta técnica como Inundación de CONTINUACIÓN HTTP/2, y notificó el problema al Centro de Coordinación CERT (CERT/CC) el 25 de enero de 2024.

> Según un aviso del CERT/CC emitido el 3 de abril de 2024, *«muchas implementaciones de HTTP/2 no aplican límites adecuados ni realizan la desinfección necesaria de la cantidad de marcos de CONTINUACIÓN enviados en un único flujo».*
>
> *«Un atacante que pueda enviar paquetes a un servidor objetivo puede inundarlo con una secuencia de marcos de CONTINUACIÓN. Estos marcos no se añadirán a la lista de encabezados en memoria, pero aún serán procesados y decodificados por el servidor, o se añadirán a la lista de encabezados, lo que provocará un fallo de falta de memoria (OOM)».*

Al igual que en el protocolo HTTP/1, HTTP/2 utiliza campos de encabezado dentro de solicitudes y respuestas. Estos campos pueden formar listas de encabezados, que a su vez se serializan y dividen en bloques de encabezado. Estos bloques se fragmentan y se transmiten en marcos de CABECERA o en marcos de CONTINUACIÓN.

«El marco de CONTINUACIÓN (tipo=0x9) se emplea para continuar una secuencia de fragmentos de bloques de encabezado», como se especifica en la documentación de RFC 7540.

> *«Se pueden enviar cualquier número de marcos de CONTINUACIÓN, siempre y cuando el marco anterior esté en el mismo flujo y sea un marco de CABECERAS,*

> *PROMESA_PUSH o CONTINUACIÓN sin que se establezca el indicador END_HEADERS».*

El último marco que contenga encabezados tendrá el indicador END_HEADERS establecido, lo que indica al extremo remoto que es el final del bloque de encabezado.

Según Nowotarski, la Inundación de CONTINUACIÓN es una clase de vulnerabilidades presente en varias implementaciones del protocolo HTTP/2, que representa una amenaza más severa que el ataque de Restablecimiento Rápido descubierto en octubre de 2023.

> *«Un solo dispositivo (e incluso, en ciertos casos, una única conexión TCP o un puñado de marcos) puede interrumpir la disponibilidad del servidor, desde causar bloqueos hasta una degradación significativa del rendimiento. Es importante destacar que las solicitudes que constituyen un ataque no quedan registradas en los registros de acceso HTTP»,* declaró el investigador.

La vulnerabilidad, en su esencia, se relaciona con el manejo incorrecto de CABECERAS y múltiples marcos de CONTINUACIÓN, lo cual crea una condición de DoS.

En resumen, un atacante puede iniciar un nuevo flujo HTTP/2 contra un servidor vulnerable y enviar CABECERAS y marcos de CONTINUACIÓN sin establecer el indicador END_HEADERS, generando un flujo incesante de encabezados que el servidor HTTP/2 debe analizar y almacenar en memoria.

Los resultados exactos varían según la implementación, pero los impactos pueden ir desde bloqueos inmediatos tras enviar unos pocos marcos HTTP/2 y bloqueos por falta de memoria, hasta el agotamiento de la CPU, afectando así la disponibilidad del servidor.

> Nowotarski señaló que *«el RFC 9113 [...] menciona múltiples problemas de*

> *seguridad que pueden surgir si los marcos de CONTINUACIÓN no se manejan adecuadamente».*
>
> *«Al mismo tiempo, no hace referencia a una instancia específica en la cual se envíen marcos de CONTINUACIÓN sin el último indicador END_HEADERS, lo cual podría tener efectos adversos en los servidores afectados.»*

Este problema afecta a varios proyectos como amphp/http (CVE-2024-2653), Apache HTTP Server (CVE-2024-27316), Apache Tomcat (CVE-2024-24549), Apache Traffic Server (CVE-2024-31309), Envoy proxy (CVE-2024-27919 y CVE-2024-30255), Golang (CVE-2023-45288), el paquete Rust h2, nghttp2 (CVE-2024-28182), Node.js (CVE-2024-27983) y Tempesta FW (CVE-2024-2758).

Se recomienda a los usuarios actualizar el software afectado a la versión más reciente para mitigar las posibles amenazas. En caso de no contar con una solución, se aconseja considerar la desactivación temporal de HTTP/2 en el servidor.