Los investigadores de seguridad cibernética detallaron una vulnerabilidad de seguridad de alta gravedad recientemente parcheada en la popular biblioteca Fastjson, que podría explotarse potencialmente para lograr la ejecución remota de código.

Rastreada como CVE-2022-25845 (puntaje CVSS: 8.1), el problema se relaciona con un caso de deserialización de datos que no son de confianza en una función compatible llamada «AutoType». Fue parcheada por los mantenedores del proyecto en la versión 1.2.83 lanzada el 23 de mayo de 2022.

> *«Esta vulnerabilidad afecta a todas las aplicaciones Java que se basan en Fastjson versiones 1.2.80 o anteriores y que pasan datos controlados por el usuario a las API JSON.parse o JSON.parseObject sin especificar una clase para deserializar»,* dijo Uriya Yavnieli, de JFrog.

Fastjson es una biblioteca de Java que se utiliza para convertir objetivos de Java en su representación JSON y viceversa. AutoType, la función vulnerable a la falla, está habilitada predeterminadamente y está diseñada para especificar un tipo personalizado al analizar una entrada JSON que luego se puede deserializar en un objeto de la clase adecuada.

> *«Sin embargo, si el JSON deserializado está controlado por el usuario, analizarlo con AutoType habilitado puede generar un problema de seguridad de deserialización, ya que el atacante puede instanciar cualquier clase que esté disponible en Classpath y alimentar a su constructor con argumentos arbitrarios»,* explicó Yavnieli.

Aunque los propietarios del proyecto introdujeron previamente un modo seguro que deshabilita AutoType y comenzaron a mantener una lista de bloqueo de clases para defenderse contra fallas de deserialización, la vulnerabilidad recién descubierta sortea la última de estas restricciones para dar como resultado la ejecución remota de código.

Se recomienda a los usuarios de Fastjson que actualicen a la versión 1.2.83 o habiliten el modo seguro, que desactiva la función independiente de la lista de permitidos y la lista de bloqueo utilizada, cerrando de forma efectiva las variantes del ataque de deserialización.

> *«Aunque existe un exploit PoC público y el impacto potencial es muy alto (ejecución remota de código), las condiciones para el ataque no son triviales (pasar una entrada no confiable a API vulnerables específicas) y, lo que es más importante, se requiere una investigación específica del objetivo para encontrar una solución adecuada»,* dijo Yavnieli.