Investigadores especializados en ciberseguridad publicaron una prueba de concepto (Proof of Concept, PoC) que demuestra la explotación de una vulnerabilidad crítica de seguridad en Fortinet FortiSIEM, la cual fue corregida recientemente y podría permitir la ejecución de código arbitrario.

La vulnerabilidad, identificada como CVE-2026-106173 y con una puntuación CVSS de 9.4, corresponde a un fallo crítico de recorrido de rutas (path traversal). Esta debilidad podría permitir que un atacante sin autenticación previa ejecute código o comandos no autorizados mediante el envío de paquetes diseñados específicamente para explotar el problema.

La semana pasada, Fortinet publicó las correcciones correspondientes para las siguientes versiones del producto:

- FortiSIEM 7.5.0 (actualizar a la versión 7.5.1 o una posterior).
- FortiSIEM 7.6 (no se encuentra afectada).

De acuerdo con Horizon3.ai, el problema se produce cuando un atacante sin autenticar establece una conexión con el servicio gRPC, que escucha en el puerto TCP 27918, y aprovecha la vulnerabilidad para sobrescribir un archivo existente ubicado fuera del directorio previsto. De esta manera, la capacidad de escribir archivos arbitrarios puede convertirse en un mecanismo para ejecutar código de forma remota.

«El controlador QueryResultReq de gRPC en FortiSIEM procesa metadatos que no son de confianza sin aplicar una validación adecuada de las rutas», explicó Horizon3.ai. «Mediante la creación de metadatos que combinan secuencias de recorrido de directorios (../) e inyección de bytes nulos en las capas de codificación HTTP/2 y HPACK de gRPC, un atacante puede escribir archivos arbitrarios en el sistema de archivos».

El origen del problema se encuentra en la construcción insegura de rutas dentro de una función denominada `SendQueryResult()`. Esta función utiliza directamente un parámetro proporcionado por el usuario, llamado `query-id`, sin someterlo a controles de validación suficientes.

En términos prácticos, un atacante puede aprovechar esta deficiencia para introducir secuencias de navegación entre directorios, como `../`, y salir de la ubicación inicialmente prevista. Sin embargo, esta técnica únicamente le permite desplazarse hasta un directorio de su elección y escribir archivos en él; no le permite seleccionar directamente el nombre del archivo, ya que el programa agrega automáticamente uno propio: `result-<source IP address>-0-0-<id>.dat`. En este formato, `<id>` corresponde a un contador cuyo valor es controlado por el servidor.

En este punto interviene una segunda vulnerabilidad: es posible insertar un byte nulo (`\0`) en el parámetro `query-id`. El sistema operativo interpreta este carácter como el final de una cadena de texto y descarta todo el contenido que aparece después de él. Como consecuencia, el sufijo del nombre de archivo que el programa agrega automáticamente se ignora cuando la ruta se entrega a la función encargada de abrir el archivo.

Un atacante podría aprovechar este comportamiento para construir un valor de `query-id` que incluya `\0` después del nombre del archivo que desea sobrescribir. Un ejemplo sería `../../../../../opt/charting/redishb.sh\0...`. El archivo objetivo es un script de shell que realiza comprobaciones del estado del servicio Redis y que se ejecuta cada 60 segundos mediante una tarea programada de cron.

«La escritura se realiza exactamente en la ruta que aparece antes del byte nulo», señaló Zach Hanley, director de ingeniería de ataques en Horizon3.ai. «Cuando se apunta a un archivo existente, su contenido se sobrescribe directamente, mientras se conservan el propietario y los permisos originales. Una segunda variante utiliza dos solicitudes para crear un archivo nuevo con un nombre específico».