



ช่องโหว่ร้ายแรงใน FortiSIEM ถูกเผยแพร่โค้ดตัวอย่างโจมตี PoC ที่เปิดทางให้ผู้โจมตีส่งร่นโค้ดจากระยะไกลได้โดยไม่ต้องยืนยันตัวตน

นักวิจัยด้านความปลอดภัยไซเบอร์ได้เผยแพร่โค้ดตัวอย่างสำหรับพิสูจน์แนวคิดการโจมตี (Proof-of-Concept หรือ PoC) ที่ใช้ประโยชน์จากช่องโหว่ระดับร้ายแรงใน FortiSIEM ซึ่งเพิ่งได้รับการแก้ไขไปไม่นาน โดยช่องโหว่นี้อาจเปิดทางให้ผู้โจมตีสามารถสั่งรันโค้ดที่ต้องการบนระบบได้

ช่องโหว่ดังกล่าวถูกติดตามภายใต้รหัส CVE-2026-106173 (คะแนน CVSS 9.4) เป็นช่องโหว่ประเภท Path Traversal ระดับร้ายแรง ซึ่งอาจทำให้ผู้โจมตีสามารถส่งรหัสคีย์หรือคำสั่งบนระบบโดยไม่ได้รับอนุญาตผ่านการส่งแพ็กเก็ตข้อมูลที่ถูกสร้างขึ้นมาเป็นพิเศษ โดยไม่จำเป็นต้องผ่านขั้นตอนการยืนยันตัวตน

Fortinet ได้ออกแพตช์แก้ไขช่องโหว่นี้เมื่อสัปดาห์ที่ผ่านมา โดยมีรายละเอียดเวอร์ชันที่ได้รับผลกระทบและแนวทางการแก้ไขดังนี้

- FortiSIEM 7.5.0 (อัปเดตเป็นเวอร์ชัน 7.5.1 หรือใหม่กว่า)
- FortiSIEM 7.6 (ไม่ได้รับผลกระทบ)

ตามรายงานของ Horizon3.ai ช่องโหว่นี้เกิดจากการที่ผู้โจมตีสามารถเชื่อมต่อไปยังบริการ gRPC ที่เปิดรับการเชื่อมต่อผ่านพอร์ต TCP 27918 ได้โดยไม่ต้องยืนยันตัวตน จากนั้นใช้ประโยชน์จากช่องโหว่เพื่อเขียนไฟล์ที่มีอยู่เดิมนอกไดเรกทอรีที่ระบบกำหนดไว้ และนำความสามารถในการเขียนไฟล์ไปยังตำแหน่งที่ต้องการมาใช้เพื่อส่งคืนโค้ดจากระยะไกล

Horizon3.ai อธิบายว่า ส่วนประมวลผลคำขอ QueryResultReq ของบริการ gRPC ใน FortiSIEM มีการจัดการข้อมูล Metadata ที่ได้รับจากภายนอกโดยไม่ได้ตรวจสอบเส้นทางไฟล์อย่างเหมาะสม ทำให้ผู้โจมตีสามารถสร้าง Metadata ที่มีทั้งชุดอักขระสำหรับย้อนกลับไปยังไตรรกทอรีก่อนหน้า (..) และการแทรกอักขระ NUL Byte ผ่านกระบวนการเข้ารหัส HTTP/2 และ HPACK ของ gRPC เพื่อเขียนไฟล์ไปยังตำแหน่งต่างๆ ภายในระบบไฟล์ได้ตามต้องการ

ปัญหานี้มีสาเหตุมาจากการสร้างเส้นทางไฟล์ที่ไม่ปลอดภัยภายในฟังก์ชันที่ชื่อว่า `SendQueryResult()` ซึ่งนำค่าพารามิเตอร์ `"query-id"` ที่ผู้ใช้งานส่งเข้ามาไปใช้งานโดยตรง โดยไม่มีการตรวจสอบความถูกต้องอย่างเพียงพอ

กล่าวอีกนัยหนึ่งคือ ผู้โจมตีสามารถใช้ประโยชน์จากช่องโหว่นี้ด้วยการแทรกชุดอักขระสำหรับย้อนกลับไปยังไตรรกทอรีก่อนหน้า เช่น `"../"` เพื่อออกจากไตรรกทอรีที่ระบบกำหนดไว้ อย่างไรก็ตาม วิธีนี้เพียงอย่างเดียวจะทำให้ผู้โจมตีสามารถเลือกตำแหน่งไตรรกทอรีที่ต้องการเขียนไฟล์ได้ แต่ยังไม่สามารถกำหนดชื่อไฟล์เองได้ เนื่องจากโปรแกรมจะเพิ่มชื่อไฟล์ในรูปแบบ `"result-<source IP address>-0-0-<id>.dat"` โดยอัตโนมัติ ซึ่งค่า `<id>` เป็นตัวนับที่ระบบเซิร์ฟเวอร์เป็นผู้กำหนด

จุดนี้เองที่ทำให้ช่องโหว่อีกประการหนึ่งเข้ามามีบทบาท โดยผู้โจมตีสามารถแทรกอักขระ Null Byte (`"\0"`) ลงในพารามิเตอร์ `"query-id"` เพื่อทำให้ระบบปฏิบัติการตีความว่าอักขระดังกล่าวเป็นจุดสิ้นสุดของข้อความ และละทิ้งข้อมูลทั้งหมดที่อยู่หลังจากนั้น ส่งผลให้ส่วนชื่อไฟล์ที่โปรแกรมเพิ่มต่อท้ายโดยอัตโนมัติถูกมองข้ามไป เมื่อเส้นทางไฟล์ดังกล่าวถูกส่งไปยังฟังก์ชันสำหรับเปิดไฟล์

ผู้โจมตีสามารถใช้ประโยชน์จากพฤติกรรมนี้ด้วยการสร้างค่า `"query-id"` ที่มีอักขระ `"\0"` อยู่หลังชื่อไฟล์ที่ต้องการเขียนทับ เช่น `"../..../..../opt/charting/redisshb.sh\0..."` โดยไฟล์ดังกล่าวเป็น Shell Script ที่ใช้ตรวจสอบสถานะการทำงานของบริการ Redis และถูกตั้งค่าให้ทำงานโดยอัตโนมัติทุกๆ 60 วินาทีผ่าน Cron Job

Zach Hanley หัวหน้าวิศวกรด้านการโจมตี (Chief Attack Engineer) ของ Horizon3.ai อธิบายว่า ข้อมูลที่ถูกเขียนจะไปยังเส้นทางไฟล์ที่ระบุไว้ก่อนหน้าอักขระ Null Byte โดยตรง หากเส้นทางดังกล่าวชี้ไปยังไฟล์ที่มีอยู่แล้ว ระบบจะเขียนทับไฟล์นั้นทันที โดยยังคงเจ้าของไฟล์และสิทธิ์การเข้าถึงเดิมไว้ นอกจากนี้ ยังมีวิธีโจมตีอีกรูปแบบหนึ่งที่ใช้คำขอ 2 ครั้ง เพื่อสร้างไฟล์ใหม่พร้อมกำหนดชื่อไฟล์ได้อย่างแม่นยำ

ข้อมูลอ้างอิง

Oct 20, 2026, By Ravie Lakshmanan

- <https://thehackernews.com/2026/10/critical-fortisiem-flaw-gets-public.html>