Mitigazione vulnerabilità Ghostcat su Tomcat

Objettivo

Nel contesto dell'esercitazione assegnata nel corso di cybersecurity, l'obiettivo era identificare e mitigare la vulnerabilità Apache Tomcat AJP Connector Request Injection (Ghostcat - CVE-2020-1938) rilevata da uno scan Nessus, agendo direttamente sulla configurazione del server Tomcat installato sulla macchina vulnerabile.

Ambiente di lavoro

Macchina Kali Linux: usata per avviare Nessus e analizzare la rete.

Macchina target: Metasploitable con Tomcat installato.

Connessione Internal Network (no accesso a internet).

Nessun aggiornamento consentito: mitigazione eseguita manualmente, senza patch.

Scansione iniziale Nessus

È stata effettuata una prima scansione Nessus sulla rete interna.

La scansione ha rilevato tra le criticità principali la seguente vulnerabilità:

Apache Tomcat AJP Connector Request Injection (Ghostcat)

Porta coinvolta: 8009/tcp

Grado di rischio: Critico

Descrizione: la vulnerabilità consente a un attaccante remoto non autenticato di leggere file interni o eseguire codice tramite il connettore AJP non protetto.

Attività di mitigazione

1. Individuazione file di configurazione

Tomcat conserva il suo file di configurazione principale in:
cd /var/lib/tomcat5.5/conf/
Verifica del contenuto con:
Is
Presente il file server.xml, che definisce i connettori del server.
2. Modifica del file server.xml
Il file è stato aperto con l'editor nano:
sudo nano server.xml
Per cercare la riga relativa al connettore AJP, è stato utilizzato il comando di ricerca interno:
Ctrl + W
Ricerca di: 8009
Trovata la seguente riga vulnerabile:
<connector port="8009" protocol="AJP/1.3" redirectport="8443"></connector>
Questa riga è stata commentata per disattivarla, senza cancellarla:
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" / >
Il file è stato poi salvato (Ctrl + O) e chiuso (Ctrl + X).

```
GNU nano 2.0.7
                                                                                    Modified
                                  File: server.xml
  <!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
  Connector port="8443" maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
                acceptCount="100" scheme="https" secure="true clientAuth="false" sslProtocol="TLS" />
  <!-- Define an AJP 1.3 Connector on port 8009 -->
  <!-- <Connector port="8009"
           _enableLookups="false" redirectPort="8443" protocol="AJP/1.3" /> -->
   <!-- Define a Proxied HTTP/1.1 Connector on port 8082 -->
  <!-- See proxy documentation for more information about using this. -->
   <Connector port="8082"
                maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
                              R Read File Y Prev Page R Cut Text
Where Is V Next Page U UnCut Tex
 Get Help
              🔭 WriteOut
                                                                              C Cur Pos
                                              Next Page UnCut Text T
```

3. Riavvio del servizio

Dopo la modifica, per rendere effettive le modifiche è stata riavviata direttamente la macchina target con:

sudo reboot

(Alternativa possibile: sudo service tomcat5.5 restart)

Scansione finale Nessus

Dopo il riavvio, è stata eseguita una seconda scansione Nessus dalla macchina Kali.

La porta 8009 risulta non più accessibile, e la vulnerabilità Apache Tomcat AJP Connector Request Injection (Ghostcat) è scomparsa dalla lista delle criticità.

Conclusione

La vulnerabilità Ghostcat è stata efficacemente mitigata senza aggiornamenti del sistema, semplicemente disabilitando il connettore AJP nel file di configurazione server.xml di Tomcat.

L'intervento ha dimostrato come in ambienti legacy o vincolati, sia possibile ridurre i rischi con misure di hardening manuale, agendo sulle configurazioni di servizio.

- Panagiotis Diamantopoulos