LOG4J – HILFE ZUR SELBSTHILFE

Erste Schritte zur Behandlung der Sicherheitslücken in der Log4j Protokollierungsbibliothek für Java-Anwendungen, Stand: Version 1.11, 05.01.22, 07:00 Uhr

SACHVERHALT

In Apache **Log4j**, einer weit verbreiteten Java-Protokollierungsbibliothek, wurde eine Zero-Day-Schwachstelle für **Remote Code Execution** (CVE-2021-44228[1], **Base Score 10.0**) entdeckt, die es Angreifern ermöglicht, ohne Authentifizierung die vollständige Kontrolle über Systeme zu übernehmen.

Die Sicherheitslücke wurde am 9.12.2021 über GitHub[2] öffentlich bekannt gegeben. Zudem wurden nach Veröffentlichung weitere Sicherheitslücken gefunden.[3] Betroffen sind mindestens die Versionen 2.0 bis 2.17.0 von Apache Log4j. Die nicht mehr gepflegten Versionen 1.x haben mit CVE-2021-4104 eine ähnliche Schwachstelle. Ausgenutzt wird die Sicherheitslücke nach aktuellem Stand bereits seit dem 01.12.2021[4] jedoch erst mit der Veröffentlichung am 9.12.2021 wurden auch Massenangriffe bekannt.

Generell ist **alles unterhalb von Log4j Version 2.17.1[5]** (nach aktuellem Stand) unter bestimmten Umständen verwundbar! Dies beinhaltet auch die bereits von den Produkt-Herstellern zu Verfügung gestellten Patches, da diese oft nur die Version 2.15 oder 2.16 installieren und damit trotzdem ggf. noch angreifbar sind.

Die Log4j-Protokollausgabe ermöglicht die Einbeziehung von Variablen. Diese Funktionalität ermöglicht es Angreifern jedoch auch, externe Java-Bibliotheken über **${jndi:ldap://** und **${jndi:ldaps://** aufzurufen, was die Möglichkeit eröffnet, ohne großen zusätzlichen Aufwand Shell-Dropping durchzuführen. Darüber hinaus können Bedrohungsakteure **${jndi:rmi** nutzen, um Befehle innerhalb der aktuellen Umgebung auszuführen. In Cloud-Diensten konnten über die Log4j-Protokollausgabe Zugangsdaten - wie beispielsweise Access Keys - ausgelesen werden, wodurch unter Umständen ein weitreichender Zugriff auf die Cloud-Dienste möglich wird.[6]

[1] https://nvd.nist.gov/vuln/detail/CVE-2021-44228
[2] https://github.com/apache/logging-log4j2/pull/608
[3] zuletzt am 28.12.2021, CVE-2021-44832
[4] https://twitter.com/eastdakota/status/1469800951351427073
[5] Die Log4j Version 2.12.2 ist nicht von der Schwachstelle betroffen, da sie der Kompatibilität mit Java Version 7 dient.
[6] https://twitter.com/Laughing_Mantis/status/1469804737566425089

CC BY SA

TLP:WHITE

HISOLUTIONS

HiSolutions AG
Schloßstraße 1
12163 Berlin

info@hisolutions.com
www.hisolutions.com

+49 30 533 289-0
+49 30 533 289-900

Hinweis: Im Folgenden werden die aktuell verfügbaren Informationen genutzt, um eine mögliche Vorgehensweise zu definieren. Das Dokument wird laufend aktualisiert und auf die aktuellen Gegebenheiten und neuen Erkentnisse angepasst.

Dieses Dokument richtet sich an IT-ExpertInnen und AdministorInnen, welche in Eigenregie die Systeme überprüfen und absichern können. Eine Erläuterung für Verbraucherinnen und Verbraucher hat das BSI veröffentlicht[7].

Achten Sie darauf, immer die aktuellste Version zu nutzen:

https://research.hisolutions.com

Bitte melden Sie uns fachliche Ergänzungen, die wir dann gerne aufnehmen.

[7] https://www.bsi.bund.de/DE/Themen/Verbraucherinnen-und-Verbraucher/Cyber-Sicherheitslage/Schwachstelle-log4Shell-Java-Bibliothek/log4j_node.html

TLP:WHITE

BETROFFENE SYSTEME UND GEFÄHRDUNGSLAGE

BETROFFENE SYSTEME

Es existiert aktuell noch keine vollständige Liste von betroffenen Systemen, da die Komponente in vielen Softwareanwendungen eingesetzt wird. Aktuell gepflegte Listen mit angreifbarer Software sind unter anderem

- https://www.techsolvency.com/story-so-far/cve-2021-44228-log4j-log4shell/#affected-products
- https://gist.github.com/SwitHak/b66db3a06c2955a9cb71a8718970c592
- https://github.com/NCSC-NL/log4shell/tree/main/software
- https://github.com/cisagov/log4j-affected-db

Auch wenn Sie keine der genannten Produkte im Einsatz haben, sollte nach Möglichkeit jedes System auf die Schwachstelle überprüft werden. Hierzu zählen neben Windows- und Linux-Servern auch Appliances wie Firewalls, Logging-Server, Hypervisoren, Router, Managed Switches und andere Systeme, auf denen potenziell Java zum Einsatz kommen kann. Bei diesen Infrastruktursystemen sind meist die Management-Komponenten betroffen.

GEFÄHRDUNDSLAGE

Laut BSI sind vor allem exponierte IT-Systeme im Internet und die Ausbreitung im lokalen Netzwerk ein Problem.

Webanwendungen geben oft die empfangenen Strings weiter, oder diese werden auf dem Weg durch Sicherheitssysteme wie eine Web Application Firewall oder ein Monitoring Tool überwacht. Dies kann dazu führen, dass betroffene Systeme automatisch versuchen können, Schadcode nachzuladen. Hierbei gibt es keine Begrenzung auf bestimmte Protokolle. Es sind bereits Fälle bekannt geworden, in denen LDAP und DNS genutzt worden sind.

Sollte kein Schadcode ausgeführt werden, könnten jedoch trotzdem sensible Umgebungsvariablen wie Access Keys oder Passwörter weitergegeben werden, welche dann für weiterführende Angriffe genutzt werden.

Zudem können durch „Lateral Movement“ auch interne Systeme wie Virtualisierungssysteme oder Datenbanksysteme Ziel des Angriffes sein, obwohl sie nicht aus dem Internet erreichbar sind.

TLP:WHITE

INDUSTRIAL CONTROL SYSTEMS

Zudem kann die Software auch in industriell genutzen Maschinen und Produktionsumgebungen genutzt werden. Daher sollten auch diese Systeme und Komponenten (ICS, SCADA, DCS, SPS, IIoT und IoT) hinsichtlich dieser Sicherheitslücke überprüft und die Hersteller kontaktiert werden.

Wir empfehlen, mit den entsprechenden Fachabteilungen in Kontakt zu treten und zu prüfen, ob alle Geräte und Softwarekomponenten inventarisiert wurden. Systeme zur automatisierten Datenerfassung (Scanner, Kameras, RFID, BLE,…) werden oft in getrennten Netzwerken betrieben und "entziehen" sich damit dem Blick eines Netzwerkmonitorings. Zudem werden diese oft durch Hersteller oder Integratoren betreut, so dass hier die Verantwortlichkeit nicht immer eindeutig geklärt ist.

PRIORISIERUNG

Wir empfehlen, priorisiert vorzugehen. Dabei sollten zuerst die relevanten Security-Systeme (wie beispielsweise das Security Information and Event Management (SIEM), Intrusion Detektion Systeme und Web Application Firewalls) gepatcht werden. Dann alle Systeme, welche direkt an das Internet angeschlossen sind (Front-End Systeme). Im Anschluss dann alle weiteren Systeme.

EIGENSCHAFTEN VON POTENZIELL GEFÄHRDETER SOFTWARE

Es sind potentiell alle Systeme betroffen, bei denen die folgenden Merkmale zutreffen:

- Technologie-Basis Java (bzw. JVM, also auch jRuby, jPython, Scala, Kotlin, ...) + Log4j (Version 2.0 bis 2.17.0).
  Hinweis: In der Version 2.16.0 wurde die Eigenschaft log4j2.enableJndi eingeführt. Diese Eigenschaft ist standardmäßig deaktiviert[8].

- Der Angriffspfad funktioniert aktuell in **jeder** Java-Version (bis Version 2.15.x, ab 2.16.0 nur sofern der Standardwert in „true" geändert wurde), solange sich die in der serialisierten Nutzlast verwendeten Klassen im Klassenpfad der Anwendung befinden.

- Nutzereingaben werden verarbeitet und mindestens in Teilen über Log4j geloggt (selbst in der x-ten Weitergabe). Dies kann auch Meldungen über fehlgeschlagene Anmeldeversuche betreffen!

- Das System hat Zugriff ins Internet oder eine DNS-Auflösung, die auch für Adressen im Internet funktioniert.

[8] „When true, Log4j components that use JNDI are enabled. When false, the default, they are disabled." (vgl. https://logging.apache.org/log4j/2.x/manual/configuration.html)

CC BY SA

Hinweis: Es wurde festgestellt, dass der Fix zur Behebung von CVE-2021-44228 in Apache Log4j 2.15.0 in bestimmten Nicht-Standard-Konfigurationen **unvollständig** war. Dazu wurde die CVE-2021-45046[9] vergeben. Es wurde ein **weiterer Fehler** (CVE-2021-45105) gefunden, der in Version Apache Log4j 2.17.0 behoben wurde. Für diese Version existiert ebenfalls eine Schwachstelle (CVE-2021-44832), die jedoch nur von einem Angreifer ausgenutzt werden kann, der bereits Schreibrechte auf die Logging-Konfiguration hat. Apache Log4j 2.17.1 behebt auch diese Schwachstelle.

[9] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-45046

TLP:WHITE

SCHRITTE ZUR BEHANDLUNG DER SCHWACHSTELLE

Schritt 1: Überprüfen einer möglichen Verwundbarkeit und erste Maßnahmen

LOG4J-DETECTOR

Führen Sie auf allen Systemen, wo dies möglich ist, den Log4j-Detector von https://github.com/mergebase/log4j-detector aus. Hierfür muss eine Java-Umgebung auf dem System installiert werden, sofern dies noch nicht der Fall war. Da viele Anwendungen ihre eigene Java-Umgebung mitbringen, kann auch ein System ohne global installierte Java-Umgebung betroffen sein.

Der Detector wird unter Linux/Unix mittels

```
java -jar log4j-detector-2021.12.15.jar / > hits.txt
```

bzw. unter Windows mittels

```
java -jar log4j-detector-2021.12.15.jar C:\ > hits.txt
```

(und ggf. weiteren Laufwerken) ausgeführt.

Finden sich nach erfolgreichem Durchlauf in der `hits.txt` Einträge, ist das System über die Log4Shell-Schwachstelle verwundbar und sollte schnellstmöglich vom Netzwerk getrennt, analysiert und dann abgesichert werden.

PRÜFUNG AUF LINUX-SYSTEMEN

Über die folgenden Konsolenbefehle können Sie Linux-Systeme auf das Vorhandensein von log4j untersuchen:

```
ps aux | grep -i 'log4j'

find / -iname "*log4j*"

lsof | grep -i „log4j“
```

TLP:WHITE

Sofern hiermit Dateien gefunden wurden, müssen diese einzeln mit dem folgenden Befehl überprüft werden, ob es sich um eine verwundbare Version handelt:

```
   unzip -p gefundene-log4j-datei.jar META-INF/MANIFEST.MF | grep
Bundle-Version
```

Durch diesen Konsolenbefehle können Sie herausfinden, wo Ihre Anwendungen Logdaten ablegen:

```
   lsof | grep '\.log'
```

Hinweis: Diese Suche findet nur derzeit aktive Log4j-Instanzen und Installationen direkt im Dateisystem, jedoch keine in (teilweise verschachtelten) Archiven vorhandenen Log4j-Dateien.

LOG4J-SCANNER VERSCHIEDENER HERSTELLER

Inzwischen wurden von vielen etablierten Herstellern von Antivirus- und Endpoint-Protection-Software, Schwachstellen-Scannern und Netzwerksicherheitssoftware Funktionen zum Scannen auf verwundbare Log4j-Versionen zur Verfügung gestellt. Diese können Sie nutzen, wenn Sie die entsprechenden Produkte im Einsatz haben. Beachten Sie dabei, dass eine Detektion, die ausschließlich ohne Zugangsdaten über das Netzwerk durchgeführt wird, in der Regel nur bestimmte Protokolle um- und nicht alle vulnerablen Versionen vollständig erfasst. Lokale Scantools müssen mindestens mit Leserechten auf alle zu scannenden Verzeichnisse ausgeführt werden.

LISTE ALLER EINGESETZTEN PRODUKTE ERSTELLEN

Es empfiehlt sich eine Liste von allen Produkten zu erstellen, welche bei Ihnen Log4j einsetzen. Diese Liste sollte anschließend regelmäßig überprüft werden. Es sollte festgehalten werden, ob die Version angreifbar ist, ob es bereits einen Patch gibt, wann dieser installiert worden ist und bei größeren Organisationsstrukturen, wer hierfür zuständig ist, beispielsweise eine bestimmte interne Abteilung oder ein IT-Dienstleister.

Hinweis: So können Sie nicht nur strukturiert feststellen, welche Systeme Sie gepatcht haben, sondern ob sich ein Angreifer bereits erfolgreich eingenistet und die Sicherheitslücke selber gepatcht hat, um weiteren Angreifern die Option zu entziehen.

Sollten Sie angreifbare Produkte haben, so befolgen Sie bitte auf jeden Fall die Empfehlungen in den Schritten 2 bis 4.

ABSTIMMUNG MIT PRODUKT-HERSTELLERN UND DIENSTLEISTERN

Treten Sie mit Ihren Produkt-Herstellern und Dienstleistern in Kontakt, um Updates zu erhalten und einzuspielen. Achten Sie dabei insbesondere darauf, dass Sie sicherstellen, dass dem Hersteller / Dienstleister auch bewusst ist, dass Version 1.x ebenfalls angreifbar

ist. Generell ist alles unterhalb von Log4j Version 2.17.1 (nach aktuellem Stand) unter bestimmten Umständen verwundbar.

Hinweis: Nur, weil kein JNDI in der Software zum Einsatz kommt, bedeutet dies nicht, dass Log4j nicht dennoch auf JNDI-Anfragen reagiert. Im Gegenteil!

Hinweis: Aussagen wie "wir setzen gar kein Java ein" können ebenfalls falsch sein: Es kann beispielsweise ein jPython/Jython, jRuby, Scala, Kotlin, Groovy, Clojure, o.ä. auf der JVM befindlich sein. Log4j ist flexibel und wird dann gerne „mal schnell“ als etablierter Logging-Standard eingesetzt.

TLP:WHITE

Schritt 2: Verhindern weiterer Angriffe

Hinweis: Die folgenden Schritte beinhalten Maßnahmen, welche von technisch versiertem Personal durchgeführt werden sollten. Bitte geben Sie dieses Dokument im Zweifel an Ihre IT-Dienstleister, welche zusammen mit Ihnen diese Schritte abarbeiten können.

VERWUNDBARE DIENSTE ABSCHALTEN

Sofern Sie Produkte einsetzen, welche aktuell verwundbar sind und nicht gepatcht werden können, sollten diese – falls möglich – abgeschaltet oder die Netzwerkverbindungen unterbrochen oder segmentiert werden. Bitte beachten Sie, dass auch nicht an das Internet angeschlossene Systeme verwundbar sein könnten, da die „Angriffs-Strings“ intern im Netz weitergegeben werden. Beispielsweise können das auch Intranet-Portale wie interne Buchungssysteme oder Krankmeldesysteme aus dem HR-Bereich sein.

Server sollten generell nur solche Verbindungen (insbesondere in das Internet) aufbauen dürfen, die für den Einsatzzweck zwingend notwendig sind. Andere Zugriffe sollten durch entsprechende Kontrollinstanzen wie Paketfilter und Application Layer Gateways unterbunden werden. Mehr dazu im Abschnitt Egress-Filter (Ausgehenden Datenverkehr unterbinden).

Hinweis: Es reichen teilweise schon mittels JNDI ausgeführte DNS-Anfragen aus, um ein System zu kompromittieren.

Für eine detaillierte Analyse sollten Datensicherungen (nach Möglichkeit durch Snapshots) erstellt werden. Virtuelle Systeme, die nicht sofort mit den genannten Maßnahmen geschützt werden können, sollten in den Standby versetzt werden, damit der Arbeitsspeicher für eine mögliche forensische Analyse erhalten bleibt.

BEVORZUGTE SCHUTZMAßNAHME: UMGEHEND PATCHEN

Sofern bereits vorhanden, spielen Sie bitte umgehend die Patches der Hersteller ein!

Dabei ist ggfs. ein Neustart notwendig. Bitte prüfen Sie täglich mehrfach, ob Patches bereitgestellt worden sind und fragen Sie bei Bedarf die Hersteller aktiv an oder öffnen Sie ein Support-Ticket.

Verifizieren Sie nach dem Patchen (beispielsweise mit Log4j-Detector), dass keine verwundbaren Log4j-Bibliotheken mehr eingesetzt werden.

SCHUTZMAßNAHMEN BEI BETROFFENEN SYSTEMEN OHNE PATCH

Sobald herstellerseitig ein Update der Anwendung bereitgestellt wurde, ist dieses unverzüglich zu installieren.

Sollte kein Update bereitstehen, sollten alle Möglichkeiten, die Schwachstelle auszunutzen, deaktiviert werden. Hierzu muss die JNDI-Lookup-Funktion der Log4J-Implementierung abgeschaltet werden.

TLP:WHITE

Dies geschieht am zuverlässigsten über die Löschung der entsprechenden .class-Datei. In den meisten Fällen dürfte es hierdurch zu keiner Beeinträchtigung der Anwendung kommen.

Hinweis: Bitte erstellen Sie sicherheitshalber für jede Datei, die durch Log4j-Detector in der `hits.txt` identifiziert wurde, eine Sicherungskopie, bevor Sie die folgenden Schritte ausführen!

Liegt die JndiLookup.class-Datei direkt in der gefundenen Datei (dies kann eine Zip-, Jar-, War-, Ear- oder Aar-Datei sein), kann sie direkt aus dieser gelöscht werden:

- mittels 7zip oder einem anderen GUI-Archivprogramm
  (Pfad ist der `hits.txt` zu entnehmen)
- mittels folgendem Aufruf (Pfad gemäß der `hits.txt` anpassen):

```
   zip -q -d log4j-core-*.jar
org/apache/logging/log4j/core/lookup/JndiLookup.class
```

- Sollten verschachtelte Archive gefunden worden sein (erkennbar an den ! zwischen den Pfaden in der `hits.txt`), müssen die einzelnen Archive erst entpackt, dann bereinigt und hinterher wieder gepackt werden

Sollte das Ersetzen der .class-Datei keine Option sein, kann nach Rücksprache mit dem Hersteller der Software ggf. die verwendete anfällige Version mit der aktuellsten Version der log4j-Bibliothek ersetzt werden.

Diese kann unter https://logging.apache.org/log4j/2.x/download.html heruntergeladen werden und sollte mindestens die Version 2.17.0 haben. Aus dem Archiv wird nur die Datei log4j-core.jar benötigt. An dieser Stelle kann jedoch nicht garantiert werden, dass diese Lösung reibungslos funktioniert. Bitte prüfen Sie diese Option daher genau mit ihren Produkt-Herstellern und IT-Dienstleistern.

Desweiteren sollte der Parameter `log4j2.formatMsgNoLookups=True` gesetzt werden, um JNDI-Lookups global zu verhindern. Dies sollte zusätzlich zu den genannten Maßnahmen geschehen, da keine der Maßnahmen 100%ige Sicherheit garantieren kann. An folgenden Stellen sollte der Parameter gesetzt werden:

- Als Umgebungsvariable des Users, der den Java-Prozess ausführt:
  `JAVA_OPTS=-Dlog4j2.formatMsgNoLookups=True`
- Als globale Umgebungsvariable:
  `LOG4J_FORMAT_MSG_NO_LOOKUPS=True`
- Als Java-Übergabeparameter im Script anhängen, das zum Start der Anwendung verwendet wird:
  `-Dlog4j2.formatMsgNoLookups=True`
- Ab Java 8u121 Setzen der folgenden Parameter beim Java-Start:
  `-Dcom.sun.jndi.rmi.object.trustURLCodebase=False`
  `-Dcom.sun.jndi.cosnaming.object.trustURLCodebase=False`

Achtung: Starten Sie nach Umsetzung von Maßnahmen das gesamte System neu!

TLP:WHITE

PROAKTIVES ERKENNEN VON ANFRAGEN (ODER AUCH NICHT)

Es ist davon auszugehen, dass aktuelle Web Application Firewall oder IPS Systeme die Angriffe nur zu einem Teil erkennen können. Sofern Sie über die technischen Möglichkeiten verfügen, können Sie zur besseren Erkennung der Angriffe auch auf die Angriffsstrings prüfen.

Allerdings werden die kritischen Anfrage-Teile wie jndi:ldap:// bereits zum Teil maskiert, beispielsweise nach folgenden Mustern:

```
${jndi:${lower:l}${lower:d}${lower:a}${lower:p}://${hostName}.}
${${::-j}${::-n}${::-d}${::-i}:${::-l}${::-d}${::-a}${::-p}://${hostName}.}
```

Es gibt unzählige Möglichkeiten, JNDI-Aufrufe zu verschleiern. Das reine Filtern der Anfragen bietet **auf keinen Fall ausreichenden Schutz**!

EGRESS-FILTER (AUSGEHENDEN DATENVERKEHR UNTERBINDEN)

Eine generelle Vorgehensweise, welche bereits Standard sein sollte, ist das Limitieren des Datenverkehrs. So sollte z.B. eine Datenbank nur mit fest definierten IT-Systemen reden dürfen und niemals eine direkte Verbindung ins Internet aufbauen. Auf diese Weise lässt sich eine mögliche Ausbreitung ggf. eindämmen. Hierbei sei auch auf mögliche DNS- und ICMP-Pakete verwiesen, welche oft eine einfache Ausleitung (Tunneling) von Daten in das Internet erlauben.

BACKUPS UND CI/CD PIPELINES

Die oben genannten Maßnahmen dienen zum akuten Schutz der Umgebung. Bitte achten Sie auch darauf, dass die Umgebung möglicherweise durch die Wiederherstellung von Backups oder durch noch nicht angepasste Bibliotheken im Kontext einer CI/CD Pipeline wieder angreifbar werden kann. Sie sollten daher sicherstellen, dass die Updates oder Anpassungen in allen „Golden Images“ wie auch in der Entwicklungskette integriert worden sind.

TLP:WHITE

Schritt 3: Maßnahmenempfehlungen (bei Kompromittierungsverdacht)

Sofern Sie angreifbar waren, sollten die Systeme vom Netz getrennt und separat untersucht werden. Zudem können Sie den Thor-Lite Scanner[10] nutzen, um nach möglichen Webshells oder anderem wie z. B. Ransomware zu suchen. Eine Anleitung zur Nutzung haben wir in unserem Research Blog[11] veröffentlicht.

Hinweis: Aufgrund der Vielzahl von möglichen Angreifern kann dies nur als ein Indiz gesehen werden. Wiederholen Sie die Prüfung regelmäßig mit aktualisierten Signaturen.

Sie sollten Ihre Systeme gründlich auf mögliche Ransomware überprüfen. Hierzu empfiehlt sich unsere Anleitung aus dem Hafnium Kontext[12].

Bei einer potenziellen Kompromittierung ist es auf jeden Fall ratsam, dass die folgenden Maßnahmen durchgeführt werden, um mögliche Folgeangriffe zu erkennen oder gar zu verhindern:

- Erhöhen der Protokollierung
  - Erhöhen Sie die Mindestaufbewahrungsfrist der Logfiles bei ausreichend freiem Speicher auf folgenden IT-Systemen:
    - Security Log des Domänenkontrollers
    - Netflow Daten (sofern vorhanden)
    - Web Proxy-Log
    - Firewall-Logs
  - Überprüfen der Logfiles auf besonders relevanter Ereignismeldungen
    - Hierbei können die SIGMA Regeln[13] verwendet werden, um die Logfiles automatisiert zu überwachen
    - Überwachen Sie externe erreichbare Fernzugänge ohne Mehr-Faktor-Authentifizierung auf unbefugte Zugriffe
- Nutzung des Thor-Lite[10] Scanners. Eine Anleitung zur Nutzung haben wir in unserem Research Blog veröffentlicht.
- Systematisches Ändern von Zugangsdaten und Passwörtern, insbesondere bei administrativen Benutzerkonten
- Überprüfen des Systems auf neu angelegte Benutzerkonten, Dienste oder zusätzlich ausgeführte Programme bzw. abgelegte Dateien
- Führen Sie regelmäßig eine vollständige Virenprüfung Ihrer IT-Systeme durch
- Erwägen Sie die Einführung einer Mehr-Faktor-Authentifizierung für externe Zugänge zu Ihrem Netzwerk
- Sofern noch nicht vorhanden, führen Sie eine Offline-Datensicherung[14] ein

[10] https://www.nextron-systems.com/thor-lite/
[11] https://research.hisolutions.com/
[12] https://research.hisolutions.com/wp-content/uploads/2021/03/HiSolutions-Ueberwachen-Ihrer-Systeme.pdf
[13] https://github.com/SigmaHQ/sigma/tree/master/rules/windows/builtin
[14] https://research.hisolutions.com/2021/03/schutz-gegen-ransomware-hisolutions-selbsthilfe-offline-backup/

TLP:WHITE

- Sensibilisierung von Personal und Dienstleistern im Hinblick auf mögliche Phishing-Angriffe/CEO-Fraud-Szenarien. Warnen Sie insbesondere vor E-Mails, die mit Bezug auf bestehende E-Mails auf geänderte Bankverbindungen oder Lieferadressen hinweisen bzw. ungewöhnliche Anlagen enthalten. Hilfestellung zum Erkennen von verdächtigen E-Mails stellt das BSI auf seiner Website[15] bereit.

Wir empfehlen die Beobachtung der Systeme gemäß diesen Empfehlungen für einen Zeitraum von 12 Monaten. In der Vergangenheit haben wir häufig 'Ruhezeiten' (nach initialer Kompromittierung bis zum eigentlichen Angriff) von 2-4 Monaten, in einigen Fällen aber auch bis zu 9 Monaten gesehen.

Prüfen Sie in der nächsten Zeit regelmäßig die unter Schritt 4: Beobachtung der Lage aufgeführten Seiten auf neue Hinweise.

DATENSCHUTZ

Der Hessische Beauftragte für Datenschutz und Informationsfreiheit (HBDI) weist zusätzlich noch darauf hin, dass das **Schließen der Schwachstelle nicht ausreichend** ist. Die Verantwortlichen müssen **zusätzlich überprüfen**, ob es bereits zu **erfolgreichen Angriffen** gekommen ist. In diesem Fall sind entsprechen weiterführende Maßnahmen zu ergreifen und zu prüfen, ob eine Meldung von Verletzungen des Schutzes personenbezogener Daten gemäß Art. 33 DS-GVO erfolgen muss[16].

Das Bayerische Landesamt für Datenschutzaufsicht hat eine „Handreichung zur Log4Shell-Erstanalyse“[17] mit einer Checkliste an Maßnahmen, die zu ergreifen sind.

[15] https://www.bsi.bund.de/DE/Themen/Verbraucherinnen-und-Verbraucher/Cyber-Sicherheitslage/Methoden-der-Cyber-Kriminalitaet/Spam-Phishing-Co/spam-phishing-co_node.html
[16] https://datenschutz.hessen.de/pressemitteilungen/unmittelbarer-handlungsbedarf-wegen-schwachstelle-in-java-bibliothek-log4j
[17] https://www.lda.bayern.de/media/veroeffentlichungen/handreichung_log4shell_baylda.pdf

TLP:WHITE

BEOBACHTUNG DER LAGE UND BEKANNTE ANGRIFFE

Schritt 4: Beobachtung der Lage

Wir gehen davon aus, dass sich die Ausnutzung der Lücke weiter entwickeln wird und in den nächsten Tagen und Wochen sukzessive mehr mögliche Angriffspfade gegen verschiedene Anwendungen und Systeme bekannt werden.

Nutzen Sie die Zeit, um alle – oder zumindest möglichst viele – der bei Ihnen genutzten Anwendungen mit verwundbaren Versionen von Log4j zu finden und zu patchen oder den Workaround anzuwenden.

Prüfen Sie in der nächsten Zeit regelmäßig die folgenden Seiten auf neue Hinweise, da die Dokumente fortlaufend aktualisiert und angepasst werden:

- **HiSolutions**: Log4Shell-Schwachstelle in Log4j: Überblick - Hilfe zur Selbsthilfe https://research.hisolutions.com/log4shell
- **BSI**: Kritische Schwachstelle in Java-Bibliothek Log4j https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Empfehlungen-nach-Angriffszielen/Webanwendungen/log4j/log4j_node.html
- **BSI**: Kritische Schwachstelle in log4j veröffentlicht (CVE-2021-44228) https://www.bsi.bund.de/SharedDocs/Cybersicherheitswarnungen/DE/2021/2021-549032-10F2.pdf
- **BSI**: Arbeitspapier Detektion und Reaktion https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Cyber-Sicherheit/Vorfaelle/log4j-Schwachstelle-2021/log4j_Schwachstelle_Detektion_Reaktion.pdf
- **BSI**: Kritische "Log4Shell" Schwachstelle in weit verbreiteter Protokollierungsbibliothek Log4j (CVE-2021-44228) https://www.bsi.bund.de/SharedDocs/Cybersicherheitswarnungen/DE/2021/2021-549177-1032.pdf

TLP:WHITE

BEKANNTE ANGRIFFE

Es ist bislang noch nicht bekannt, wer und zu welchem Zweck Server angreift. Die Anzahl der Angreifergruppen hat sich seit der Veröffentlichung stark vergrößert. Aus diesem Grund lässt sich aktuell nicht gut abschätzen, welche Maßnahmen ausreichend sind, um die Systeme zu bereinigen. Aktuell öffentlich bekannt gewordene Angriffe sind (Auszug):

Angreifer-Gruppe	Angriffsart	Betroffene Systeme	Referenz
Khonsari	Ransomware	Windows	https://therecord.media/first-log4shell-attacks-spreading-ransomware-have-been-spotted/
MIRAI	Botnetz	IoT Geräte mit ARC-Prozessor	https://blog.netlab.360.com/threat-alert-log4j-vulnerability-has-been-adopted-by-two-linux-botnets/
Muhstik (Tsunami Variante)	Botnetz + Backdoor	IoT Geräte und Linux	https://blog.netlab.360.com/threat-alert-log4j-vulnerability-has-been-adopted-by-two-linux-botnets/
Elknot	DDoS Botnetz	Linux und Windows	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
m8220	Mining Botnetz	Linux und Windows	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
SitesLoader	Botnetz	Linux	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
xmirg.pe	Mining Botnetz	Linux und Windows	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
xmirg.ELF	Mining Botnetz	Linux	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
attack tool 1	Botnetz	Linux	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
attack tool 2	Botnetz	Linux	https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/
Orcus	Remote Access Trojaner	Windows	https://businessinsights.bitdefender.com/technical-advisory-zero-day-critical-vulnerability-in-log4j2-exploited-in-the-wild
Unbekannt	Reverse Bash Shell	Linux und Windows	https://businessinsights.bitdefender.com/technical-advisory-zero-day-critical-vulnerability-in-log4j2-exploited-in-the-wild

TLP:WHITE

StealthLoader	Trojaner	Windows	https://blog.checkpoint.com/2021/12/14/a-deep-dive-into-a-real-life-log4j-exploitation/
PHOSPHORUS	Ransomware	Windows	https://www.microsoft.com/security/blog/2021/12/11/guidance-for-preventing-detecting-and-hunting-for-cve-2021-44228-log4j-2-exploitation/
HAFNIUM	Ransomware	Virtualisierung	https://www.microsoft.com/security/blog/2021/12/11/guidance-for-preventing-detecting-and-hunting-for-cve-2021-44228-log4j-2-exploitation/
Kinsing	Backdoor	Linux	https://medium.com/proferosec-osm/log4shell-massive-kinsing-deployment-9aea3cf1612d
Staatliche Akteure	Spionage		https://twitter.com/kevincollier/status/1471099511179259904
TellYouThePass	Ransomware		https://www.curatedintel.org/2021/12/tellyouthepass-ransomware-via-log4shell.html
Unbekannt	Mining		https://www.bleepingcomputer.com/news/security/log4j-attackers-switch-to-injecting-monero-miners-via-rmi/
Conti	Ransomware	VMware	https://www.bleepingcomputer.com/news/security/conti-ransomware-uses-log4j-bug-to-hack-vmware-vcenter-servers/
Lapsus	Ransomware		https://therecord.media/lapsus-ransomware-gang-hits-sic-portugals-largest-tv-channel/
Aquatic Panda	Unbekannt		https://venturebeat.com/2021/12/29/china-based-group-used-log4j-flaw-in-attack-crowdstrike-says/

Die Lage wird wahrscheinlich unübersichtlich bleiben, da die verschiedenen Angreifer mit sehr unterschiedlichen Vorgehensweisen und Zielen arbeiten. Es wird zudem mit wurmartigen Verbreitungen zu rechnen sein.

Hinweis: Da die Angriffe seit mindestens 1.12.2021 laufen, sollten aktuelle Logfiles gesichert werden, um diese später auswerten zu können. Falls möglich, sollten Netflow Logs angelegt werden, um den Datenfluss zu überwachen.

TLP:WHITE

UNGEEIGNETE MAßNAHMEN

Filter auf WAF / ReverseProxy

Wie oben gezeigt wird, werden viele Anfragen aktuell verschleiert – aktuell etwa die Hälfte. Entsprechend gering sind die Erfolgsaussichten, dies als Schutzmaßnahme zu werten, wenn bereits ein erfolgreicher Versuch reicht. Zusätzlich können – je nach Anwendung – die Angriffe auch durch Einträge im UserAgent, Cookies, Inhaltsdaten (Foto-exif-Daten, Metadaten von SMIME/SSL-Zertifikaten etc.) ausgelöst werden.

Verringern des Log-Levels

Eine Änderung der Log-Einstellungen innerhalb von Log4j hilft nicht, weil die zu loggenden Daten von Log4j ausgewertet und missbraucht werden, bevor die Log-Einstellungen zum Zuge kommen.

Logging/Alarmierung im SIEM

Die üblichen SIEM-Systeme wie z.B. Splunk, GrayLog, und alles was ElasticSearch einsetzt, sind betroffen und dienen im schlimmsten Fall nur der weiteren Ausbreitung einer Kompromittierung.

(Weiter-) Verwendung der veralteten Log4j-Version 1.x

Diese Version ist nach aktuellem Wissensstand ebenfalls mittels Log4Shell verwundbar und hat zusätzlich noch ungepatchte Schwachstellen, die dank EOL-Status des Codes nicht gepatcht werden.[18]

Eine dieser Schwachstellen ist dieser Log4Shell-Schwachstelle sehr ähnlich.[19]

Dieses Werk ist lizenziert unter einer Creative Commons Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz.

[18] https://logging.apache.org/log4j/1.2/
[19] https://access.redhat.com/security/cve/CVE-2021-4104