Recent methods and strategies used in the market: **Log4j vulnerability**

Log4shell is a critical vulnerability in the widely-used logging tool **Log4j**, which is used by millions of computers worldwide running online services. A wide range of people, including organisations, governments and individuals are likely to be affected by it. Although fixes have been issued, they will still need to be implemented.

What's the issue?

Last week, a vulnerability was found in Log4j, an open-source logging library commonly used by apps and services across the internet. If left unfixed, attackers can break into systems, steal passwords and logins, extract data, and infect networks with malicious software.

Log4j is used worldwide across software applications and online services, and the vulnerability requires very little expertise to exploit. This makes Log4shell potentially the most severe computer vulnerability in years.

Apache Log4j2 versions 2.0-beta7 through 2.17.0 (excluding security fix releases 2.3.2 and 2.12.4) are vulnerable to a remote code execution (RCE) attack where an attacker with permission to modify the logging configuration file can construct a malicious configuration using a JDBC Appender with a data source referencing a JNDI URI which can execute remote code. This issue is fixed by limiting JNDI data source names to the java protocol in Log4j2 versions 2.17.1, 2.12.4, and 2.3.2.

Who is affected by this?

Almost all software will have some form of ability to log (for development, operational and security purposes), and Log4j is a very common component used for this.

For **individuals**, Log4j is almost certainly part of the devices and services you use online every day. The best thing you can do to protect yourself is make sure your devices and apps are as up to date as possible and continue to update them regularly, particularly over the next few weeks.

For **organisations**, it may not be immediately clear that your web servers, web applications, network devices and other software and hardware use Log4j.

What is Log4j?

Modern software can be large, powerful, and complex. Rather than a single author writing all the code themselves as was common decades ago, modern software creation will have large teams, and that software is increasingly made out of 'building blocks' pulled together by the team rather than entirely written from scratch.

A team is unlikely to spend weeks writing new code when they can use existing code immediately.

-Sadiq S.

Log4j is one of the many building blocks that are used in the creation of modern software. It is used by many organisations to do a common but vital job. We call this a ‘software library’.

Log4j is used by developers to keep track of what happens in their software applications or online services. It’s basically a huge journal of the activity of a system or application. This activity is called ‘logging’ and it’s used by developers to keep an eye out for problems for users.

In December 2021, a number of vulnerabilities were reported in Log4j:

- **CVE-2021-44228** - referred to as the "Log4shell" vulnerability, affects Log4j versions 2.0-beta9 to 2.14.1. It allows remote code execution and information disclosure if exploited.
- **CVE-2021-45046** - affects versions 2.0-beta9 to 2.15.0, excluding 2.12.2 and was originally reported as a Denial of Service when organisations are running a vulnerable non-standard configuration. Later research found that the same vulnerable configuration allowed a bypass of the mitigations to Log4shell, allowing remote code execution and information disclosure.
- **CVE-2021-45105** - affects Log4j versions from 2.0-beta9 to 2.16.0 – A similar denial of service issue to CVE-2021-45046 when organisations are running a vulnerable non-standard configuration.

An application is impacted by these vulnerabilities if it consumes untrusted user input and passes this to a vulnerable version of the Log4j logging library.

What if …

… I know we are using Log4j in applications developed in house?

Update to the latest version of Log4j (currently Log4j 2.17.0).

… I know Log4j is present in applications supplied by a third party?

Keep any such products updated to the latest version. More products may release patches over the next few days and weeks, and so organisations should make sure they’re checking for updates regularly.

… I don’t know if anything we use is using Log4j?

Ask your in-house developers and/or third-party suppliers. We have asked that developers of affected software communicate promptly with their customers to enable them to apply available mitigations or install updates. In turn, you should act promptly on any such communications from developers.

-Sadiq S.

What else can we do?

- Check your systems for the use of Log4j
- Check the list of vulnerable software
- Contact software vendors
- Set Web Application Firewall rules
- Check for scanning activity
- Check for exploitation
- Deploy protective network monitoring/blocking
- Discover unknown instances of Log4j within your organisation
- Install the latest updates immediately wherever Log4j is known to be used

Detection guidance

The following recommendations will help assist in detection of potential malicious activity trying to exploit the Log4j vulnerability.

- Organisations with detection and threat intelligence functions should ensure they're aware of the current payloads being delivered by exploitation attempts and searching for evidence of them.
- If your organisation is storing netflow data for your network's internet connections, or you have robust EDR coverage of servers, you should search for internally initiated LDAP, LDAPS, RMI and DNS connections to external destinations not seen before 10 December 2021. This may indicate exploitation and if detected, you should search the initiating host for the presence of Log4j. DNS queries by the server around the suspicious connection should also be reviewed as sensitive information could have been exfiltrated over DNS.
- YARA rules for a variety of scenarios are available should organisations have the tooling to query using them: Log4j RCE Exploitation Detection
- The log files for any services using affected Log4j versions could contain user-controlled strings. For example, "jndi:ldap".

How to mitigate Log4Shell:

Any company using Log4j version 2.14.1 or below -- or patched versions 2.15.0 and 2.16.0, which contain flaws -- is vulnerable to Log4Shell. The zero-day vulnerability has been patched by the Apache Logging Services Project.

-Sadiq S.

Enterprises have to deploy the security update -- Log4j version 2.17.0 -- but note that updating mission-critical applications takes time to ensure no functionality is broken or lost.

Another challenge with this flaw is it's not necessarily obvious where Log4j is being used. It could well be included in a third-party library or dependency, so systems can't be automatically assumed safe even if Java is not installed or not running in the process list. For example, the following could be true about Log4j:

- included in a Java Runtime Environment;
- runs only when triggered by another process, like cron; or
- used directly by a cloud service connected to the network.

If your company uses a software bill of materials (SBOM) for cybersecurity, check it for Log4j, and ensure anything using it upstream or downstream has patched the vulnerability. If your company needs help creating a cybersecurity SBOM, learn more about it here.

Be aware that the initial Log4shell fix was incomplete in certain nondefault configurations (CVE-2021-45046) and a denial-of-service vulnerability (CVE-2021-45105) has been fixed in version 2.17.0 for Java 8 users.

Technologies to mitigate the Log4j flaw:

The most effective way to block malicious requests targeting Log4j is with a web application firewall (WAF). WAFs can compare request data against rules indicating CVE-2021-44228.

Attackers will develop techniques and patterns to bypass these checks, however, so keep abreast of new developments and keep WAF rules up to date.

Note that obfuscation has already been detected, so signature-based detection alone won't be sufficient. Multilayered security is the only way to establish comprehensive protection against the numerous ways the Log4j vulnerability can be exploited. Monitor and inspect outbound traffic for signs of hosts responding to a Log4Shell packet or command-and-control callbacks.

-Sadiq S.