

Code Analysis Summary

10/25/2023

[Commit](#): 223

Static Application Security Testing (SAST)

Summary: The analysis has identified 25 code issues in the project, categorized as follows:

- 5 **High** severity issues
- 3 **Medium** severity issues
- 17 **Low** severity issues

High Severity Issues:

Info: Unsensitized input from cookies flows into the return value of total_data, pie_data, cube_data, meter_data, where it is used to render an HTML page returned to the user. This may result in a Cross-Site Scripting attack (XSS)

- bokeh.py (line 14, 20, 26, 32, 38)

Medium Severity Issues:

Info: Running the application in debug mode (debug flag is set to True in run) is a security risk if the application is accessible by untrusted parties.

- server.py (line 40)

Info: Unsensitized input from remote resources is utilized to dynamically construct the HTML page on the client side, potentially resulting in DOM Based Cross-Site Scripting (DOMXSS) attacks.

- single-page-index.html (line 1002, 1019)

Low Severity Issues:

Info: Sensitive cookies in HTTPS sessions lack the 'Secure' attribute, making them susceptible to man-in-the-middle attacks.

- session.py (line 116)

Info: SQL Injection vulnerabilities exist due to unsensitized input from HTTP parameters flowing into SQL queries, potentially leading to SQL Injection attacks.

- app.py (line 132, 162, 193, 197, 133, 163, 194, 198)

Info: Cookies are missing the 'HttpOnly' flag, exposing them to malicious client-side code.

- session.py (line 116)

Info: Hardcoded credentials are found in the code, which is considered a poor security practice.

- app.py (line 122, 152, 183, 255)

Info: Unsensitized input from HTTP parameters flows into functions responsible for rendering HTML pages, potentially resulting in Cross-Site Scripting (XSS) attacks.

- app.py (line 149, 178, 251)

Docker Analysis

The analysis was an on the following Docker Images using Docker Scout and Snyk:
After closer assessment, the vulnerabilities were not relevant and patches are not out.

nginx:latest and src-web:latest

▼

debian/zlib 1:1.2.13.dfsg-1

1 C 0 H

▼

CVE-2023-45853 [🔗](#)

9.8 C


MiniZip in zlib through 1.3 has an integer overflow and resultant heap-based buffer overflow in zipOpenNewFileInZip4_64 via a long filename, comment, or extra field. NOTE: MiniZip is not a supported part of the zlib product.

CVSS Score: 9.8

Affected range: >=1:1.2.13.dfsg-1

Fix version: Not yet available

Publish date: 2023-10-18

 **snyk**

Test container images for vulnerabilities

nginx:latest

Test image

Settings

Tested 149 dependencies, found 74 vulnerabilities.

1 Critical

2 High

4 Medium

67 Low

C **zlib/zlib1g** Integer Overflow or Wraparound

From: curl@7.88.1-10+deb12u4 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx@1.25.3-1-bookworm > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-njs@1.25.3+0.8.2-1-bookworm > zlib/zlib1g@1:1.2.13.dfsg-1

From: util-linux/utlil-linux@2.38.1-5+b1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: apt@2.6.1 > apt/libapt-pkg6.0@2.6.1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: apt@2.6.1 > gnupg2/gpgv@2.2.40-1.1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: curl@7.88.1-10+deb12u4 > curl/libcurl4@7.88.1-10+deb12u4 > zlib/zlib1g@1:1.2.13.dfsg-1

From: dash@0.5.12-2 > dpkg@1.21.22 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-njs@1.25.3+0.8.2-1-bookworm > libxml2@2.9.14+dfsg-1.3-deb12u1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: curl@7.88.1-10+deb12u4 > curl/libcurl4@7.88.1-10+deb12u4 > libssh2/libssh2@1:1.10.0-3+b1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: curl@7.88.1-10+deb12u4 > curl/libcurl4@7.88.1-10+deb12u4 > rtmpdump/librtmp@2.4+20151223.gitfa8646d.1-2+b2 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-image-filter@1.25.3-1-bookworm > libgd2/libgd3@2.3.3-9 > libheif/libheif1@1.15.1-1 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-image-filter@1.25.3-1-bookworm > libgd2/libgd3@2.3.3-9 > tiff/libtiff@4.5.0-6 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-image-filter@1.25.3-1-bookworm > libgd2/libgd3@2.3.3-9 > fontconfig/libfontconfig@2.14.1-4 > freetype/libfreetype@2.12.1+dfsg-5 > zlib/zlib1g@1:1.2.13.dfsg-1

From: nginx-module-image-filter@1.25.3-1-bookworm > libgd2/libgd3@2.3.3-9 > fontconfig/libfontconfig@2.14.1-4 > freetype/libfreetype@2.12.1+dfsg-5 > libpng1.6/libpng16-16@1.6.39-2 > zlib/zlib1g@1:1.2.13.dfsg-1

[Learn more about this vulnerability](#)

<https://security.snyk.io/vuln/SNYK-DEBIAN12-ZLIB-6008963>

mariadb:latest

▼

stdlib 1.16.7

3 C 35 H

▼

CVE-2023-24540 [🔗](#)

9.8 C



Not all valid JavaScript whitespace characters are considered to be whitespace. Templates containing whitespace characters outside of the character set `"\t\n\r\u0020\u2028\u2029"` in JavaScript contexts that also contain actions may not be properly sanitized during execution.

CVSS Score: 9.8

Affected range: <1.19.9

Fix version: 1.19.9



Publish date: 2023-05-05

 CVE-2023-24538 

9.8 C

Templates do not properly consider backticks (`) as Javascript string delimiters, and do not escape them as expected. Backticks are used, since ES6, for JS template literals. If a template contains a Go template action within a Javascript template literal, the contents of the action can be used to terminate the literal, injecting arbitrary Javascript code into the Go template. As ES6 template literals are rather complex, and themselves can do string interpolation, the decision was made to simply disallow Go template actions from being used inside of them (e.g. "var a = `{{.}}`"), since there is no obviously safe way to allow this behavior. This takes the same approach as github.com/google/safehtml. With fix, Template.Parse returns an Error when it encounters templates like this, with an ErrorCode of value 12. This ErrorCode is currently unexported, but will be exported in the release of Go 1.21. Users who rely on the previous behavior can re-enable it using the GODEBUG flag jshtmlinterp=1, with the caveat that backticks will now be escaped. This should be used with caution.


CVSS Score:	9.8
Affected range:	<1.19.8
Fix version:	1.19.8
Publish date:	2023-04-05

 CVE-2022-23806 

9.1 C

Some big.Int values that are not valid field elements (negative or overflowing) might cause Curve.IsOnCurve to incorrectly return true. Operating on those values may cause a panic or an invalid curve operation. Note that Unmarshal will never return such values.

CVSS Score:	9.1
Affected range:	<1.16.14
Fix version:	1.16.14
Publish date:	2022-05-23

mariadb:latest 

Tested 153 dependencies, found 19 vulnerabilities.

0	Critical	0	High	5	Medium	14	Low
---	----------	---	------	---	--------	----	-----