Reflected XSS on conversion filter function in beancount/fava

0

Valid Reported on Jul 28th 2022

Description

Fava v1.22 have a conversion filter function on income statement dashboard which allow user to perform XSS due to improper validation on filter conversion.

Proof of Concept

Navigate to Fava demo instance https://fava.pythonanywhere.com/example-beancount-file/income_statement/.
Filter on conversion type and add payload on the result.
Hover mouse cursor to bar chart (visualization) and XSS alert will pop up.

Endpoints

https://fava.pythonanywhere.com/huge-example-file/income_statement/?conversion=at_value
https://fava.pythonanywhere.com/example-with-budgets/income_statement/?conversion=units
https://fava.pythonanywhere.com/example-beancount-file/income_statement/?conversion=at_value

Payload

"><img src=a onerror=alert(document.domain)>

Screenshot POC

xss domain
xss

Impact

This vulnerability is capable of executing a malicious javascript code in web page

Occurrences

conversion.py L1-L120

CVE
CVE-2022-2589
(Published)

Vulnerability Type
CWE-79: Cross-site Scripting (XSS) - Reflected

Severity
Medium (6.9)

Registry
Pypi

Affected Version
1.22

Visibility
Public

Status
Fixed

Found by

din
@baharuddinzulkifli
pro

This report was seen 508 times.

We are processing your report and will contact the **beancount/fava** team within 24 hours.
4 months ago

We have contacted a member of the **beancount/fava** team and are waiting to hear back
4 months ago

din modified the report 4 months ago

A **beancount/fava** maintainer modified the Severity from High (7.6) to Medium (6.9)
4 months ago

A **beancount/fava** maintainer gave praise 4 months ago

Thanks for the report :) Since Fava URLs are dependent on the name of the underlying Beancount journal and the base URLs, which should be private and require a previous attack to be determined by the attacker, I've marked the attack complexity as "high"

The researcher's credibility has slightly increased as a result of the maintainer's thanks: +1

The researcher has received a minor penalty to their credibility for miscalculating the severity: -1

A **beancount/fava** maintainer validated this vulnerability 4 months ago

din has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

A **beancount/fava** maintainer marked this as fixed in **1.22.3** with commit **68bbb6** 4 months ago

The fix bounty has been dropped ✖

This vulnerability will not receive a CVE ✖

conversion.py#L1-L120 has been validated ✔

din 4 months ago Researcher

Thanks

Sign in to join this conversation