GIF poster fetch bypasses traversal scrubbing and exposes local files through public media URLs

High **DanielnetoDotCom** published **GHSA-f4f9-627c-jh33** 2 weeks ago

Package

WWBN/AVideo

Affected versions

<=26.0

Patched versions

None

Description

Summary

`objects/aVideoEncoderReceiveImage.json.php` allowed an authenticated uploader to fetch attacker-controlled same-origin `/videos/...` URLs, bypass traversal scrubbing, and expose server-local files through the GIF poster storage path.

The vulnerable GIF branch could be abused to read local files such as `/etc/passwd` or application source files and republish those bytes through a normal public GIF media URL.

Details

The vulnerable chain was:

1. `objects/aVideoEncoderReceiveImage.json.php` accepted attacker-controlled `downloadURL_gifimage`
2. traversal scrubbing used `str_replace('../', '', ...)`, which was bypassable with overlapping input such as `..../`
3. same-origin `/videos/...` URLs were accepted
4. `url_get_contents()` and `try_get_contents_from_local()` resolved the request into a local filesystem read

5. the fetched bytes were written into the GIF destination
6. invalid GIF cleanup used the wrong variable, so the non-image payload remained on disk

This made the GIF poster path a local file disclosure primitive with public retrieval.

Proof of concept

1. Log in as an uploader and create an owned video row through the normal encoder flow.
2. Send:

```
POST /objects/aVideoEncoderReceiveImage.json.php
downloadURL_gifimage=https://localhost/videos/..../////..../////..../////..../////..../////..../////pa
```

3. Query:

```
GET /objects/videos.json.php?showAll=1
```

4. Recover the generated GIF URL from `videosURL.gif.url`.
5. Download that GIF URL.
6. Observe that the body matches the target local file, such as `/etc/passwd`, byte-for-byte.

Impact

An authenticated uploader can read server-local files and republish them through a public GIF media URL by supplying a crafted same-origin `/videos/...` path to `downloadURL_gifimage`. Because traversal scrubbing was bypassable and the fetched bytes were written to the GIF destination without effective invalid-image cleanup, successful exploitation allows disclosure of files such as `/etc/passwd`, readable application source code, or deployment-specific configuration accessible to the application.

Recommended fix

- Reject any remote image URL whose decoded path contains traversal markers
- Do not allow attacker-controlled same-origin `/videos/...` fetches to resolve into local file reads
- Constrain any local shortcut path handling with `realpath()` and strict base-directory allowlists
- Validate GIF content before saving it into public media storage
- Ensure invalid-image cleanup checks the correct destination path

Severity

High 7.6 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	Low
User interaction	None
Scope	Unchanged
Confidentiality	High
Integrity	Low
Availability	Low

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:L

CVE ID

CVE-2026-39369

Weaknesses

▶ CWE-22

Credits

 **threalwinky** Reporter