

**GOVERNMENT OF THE REPUBLIC
OF VANUATU**

PRIME MINISTER'S OFFICE

CERTVU
DEPARTMENT OF COMMUNICATIONS
& DIGITAL TRANSFORMATION

PM B 9108 Port Vila, Vanuatu

Tel: (678) 33380



**GOUVERNEMENT DE LA
REPUBLIQUE DU VANUATU**

BUREAU DU PREMIER MINISTRE

CERTVU

DEPARTMENT DE
COMMUNICATION ET DE
TRANSFORMATION NUMERIQUE

SPP 9108 Port Vila, Vanuatu

Tel: (678) 33380

8 September 2026

Advisory 269: Mail Mint (WPFunnels) handle_form_submission PHP Object Injection (CVE-2026-10196).

Release Date: 5th September 2026

Impact: **CRITICAL**

TLP: CLEAR

The Department of Communications and Digital Transformation (DCDT) through CERT Vanuatu (CERTVU), provides the following advisory.

This alert is relevant to Organizations, System/Network administrators, and marketing/website teams that operate a WordPress or WooCommerce site using the Mail Mint email marketing automation plugin by WPFunnels. This alert is intended to be understood by technical users and systems administrators.

What is it?

CVE-2026-10196 is a critical, unauthenticated PHP Object Injection vulnerability in Mail Mint, the same WordPress email marketing, newsletter, and automation plugin by WPFunnels addressed in CERTVU Advisory 257 (CVE-2026-84753). This finding was reported independently through the Wordfence bug bounty program (researcher "maru finder"), separately from Advisory 257's Patchstack-reported finding, and carries its own CVE identifier and disclosure record. The flaw lies in the plugin's handle_form_submission function: custom field submissions are deserialized without validating their origin or type, allowing an unauthenticated attacker to inject a malicious PHP object.

What are the systems affected?

The following version(s) are affected:

- Mail Mint through 1.31.0 - (Affected)

Mail Mint 1.31.1 and later - (Not affected, patched)

What does this mean?

Typical attack flow:

1. **Submit a crafted custom field value to the form submission handler**
 - An unauthenticated attacker sends a specially crafted, serialized PHP object as a custom field value in a request to a WordPress site running an affected version of Mail Mint, targeting the `handle_form_submission` function, which deserializes attacker-supplied input without validating its origin or type.
2. **Trigger a gadget chain during deserialization**
 - If the WordPress installation - core, another installed plugin, or the active theme – This contains a class with a magic method (such as `_destruct()` or `_wakeup()`) that performs a dangerous action when instantiated, PHP's deserialization process automatically invokes it, potentially resulting in arbitrary file deletion, denial of service, or remote code execution depending on the specific gadget chain available on that site.

Attack vectors:

- A network-based, unauthenticated request against any internet-reachable WordPress site running an affected version of the Mail Mint plugin.
- No user interaction, no privileges, and no special access conditions are required (CVSS AV:N/AC:L/PR:N/UI:N).

Successful exploitation may allow attackers to:

- Trigger arbitrary file deletion or a denial-of-service condition on the affected WordPress site, without ever holding a legitimate credential.
- Depending on the specific combination of plugins, themes, and WordPress core version installed on the site, potentially achieve remote code execution via an available deserialization gadget chain, and access subscriber/contact data managed by Mail Mint's email marketing functionality - the same consequences already described in Advisory 257.

Mitigation process

CERTVU recommends the following:

1. Apply the vendor patch without delay

- Update the Mail Mint plugin to version 1.31.1 or later through the WordPress admin dashboard (Plugins > Installed Plugins). This is the same update already recommended in Advisory 257- a single upgrade is expected to resolve both CVE-2026-84753 and this CVE, but confirm via the plugin's own changelog that the installed version includes the "arbitrary PHP object" fix.

2. If Advisory 257 has already been actioned, verify rather than assume coverage
3. Audit the estate for Mail Mint installations
4. Consider a Web Application Firewall (WAF) as an interim layer of defence
5. Review web-server access logs for suspicious activity

Report suspected compromise to CERTVU at incident@cert.gov.vu or on telephone (678) 33380.

Reference

1. <https://www.cve.org/CVERecord?id=CVE-2026-10196>
2. <https://wordpress.org/plugins/mail-mint/>