Stored XSS in listar_despachos.php

Moderate **nilsonLazarin** published **GHSA-mccp-8446-phw5** last week

Package

No package listed

Affected versions

<= 3.6.9

Patched versions

3.6.10

Description

Summary

A Stored Cross-Site Scripting (XSS) vulnerability allows an authenticated user to inject malicious JavaScript via the “Destinatário” field. The payload is stored and later executed when viewing the dispatch page, impacting other users.

Details

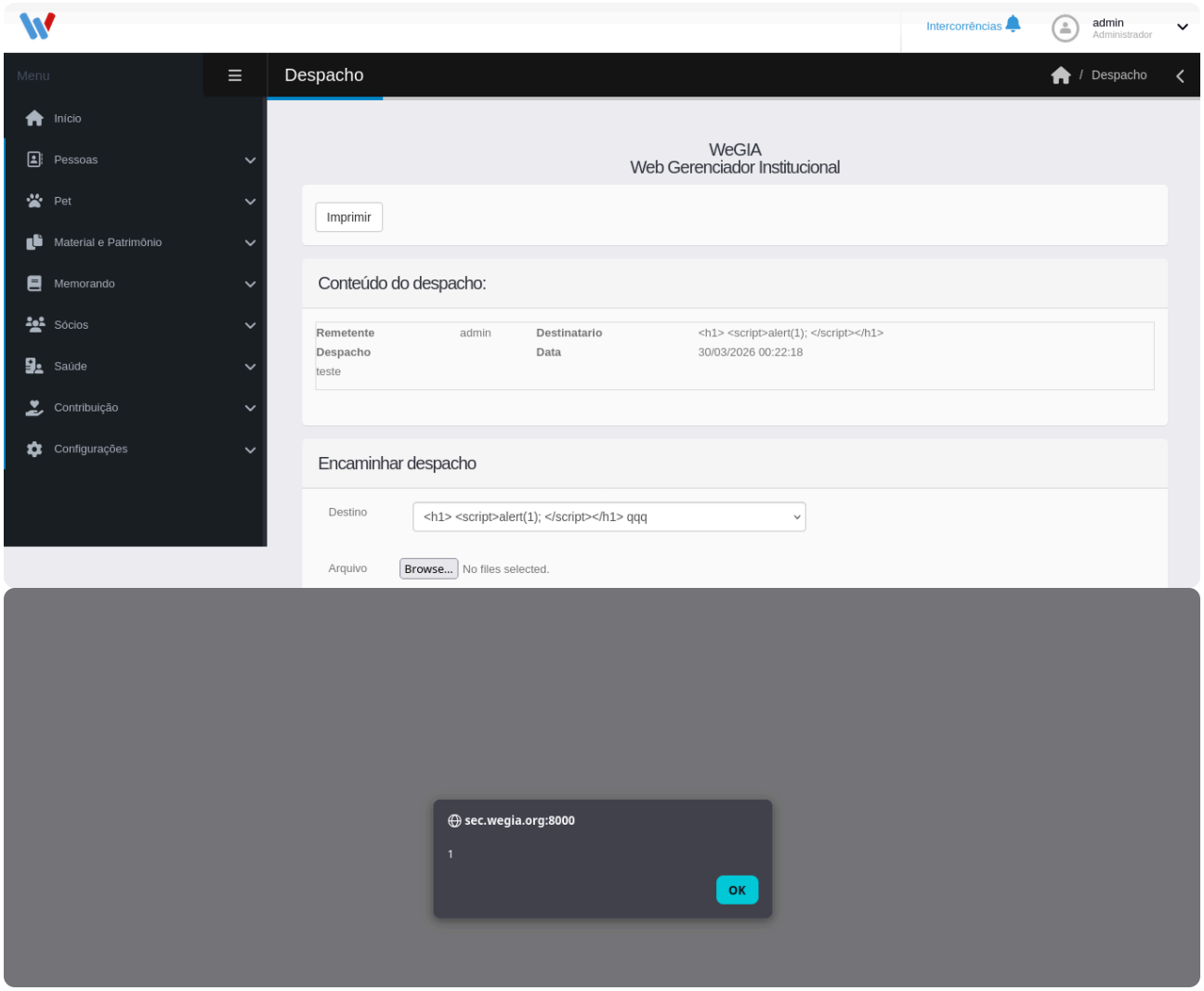
The application fails to properly sanitize or escape the “Destinatário” field, which is populated with user-controlled data (nome do usuário). When a despacho is created using a maliciously crafted name containing HTML/JavaScript, this value is stored in the system.

During the rendering of the dispatch listing page, the application inserts this data into the DOM using .html(), causing the browser to interpret and execute the injected code.

This results in a Stored XSS vulnerability due to improper output encoding of user-controlled data.

PoC

1. Alter the name of a user (or create one) with the following payload: `<h1> <script>alert(1); </script></h1>`
2. Create a despacho selecting this user as “Destinatário”
3. Access the page that lists or displays the despacho
4. Observe that the payload is executed in the browser



Impact

This is a Stored XSS vulnerability affecting any user who accesses the despacho page.

An attacker can:

Execute arbitrary JavaScript in victims' browsers
Steal session data
Perform actions on behalf of other users
Potentially compromise privileged accounts

The impact is higher if administrators or privileged users access the affected page.

Remediation

The error occours on this snippet:

```
.append($("<td colspan=4 id=texto" + item.id + ">") .html(item.texto));
```

The correct way is to avoid interpreting HTML by using .text(item.text), which inserts the content as plain text and prevents script execution. If it is necessary to allow HTML, the content must be sanitized beforehand with an appropriate library, such as DOMPurify, before being inserted with .html().

Credits

Thiago Escarrone

Severity

Moderate 6.8 / 10

CVSS v3 base metrics

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

Learn more about base metrics

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

CVE ID

CVE-2026-40284

Weaknesses

▶ CWE-79

Credits

 **ThiagoEscarrone** Reporter

 **GabrielPintoSouza** Remediation developer