View Issue Details

ID	Project	Category	View Status	Date Submitted	Last Update
0037017	mantisbt	security	public	2026-04-12 08:24	2026-05-09 19:56
Reporter	siunam	**Assigned To**	dregad		
Priority	low	**Severity**	minor	**Reproducibility**	always
Status	closed	**Resolution**	fixed		
Product Version	2.28.1				
Target Version	2.28.2	**Fixed in Version**	2.28.2		

Summary: 0037017: CVE-2026-40598 : Potential Referer-Based Reflected HTML Injection / XSS in Tag Update Page

Description:

In `tag_update_page.php` line 106, `$t_redirect_page` 's value from the request's `Referer` header did not get HTML entity encoded:

```
1 $t_redirect_page = parse_url(
2     basename( $_SERVER["HTTP_REFERER"] ?? 'tag_view_page.php' ),
3     PHP_URL_PATH
4 );
5 // [...]
6 <input type="hidden" name="redirect" value="<?php echo $t_redirect_page ?>"/>
```

Unfortunately, this is not an exploitable reflected HTML Injection / XSS vulnerability. As per RFC 3986 section 2.3, all reserved characters in the be percent-encoded (URL encoded). Therefore, all modern browsers will automatically URL encode the request's URL path.

However, if the web server is configured with server-side caching, such as Varnish Cache, or the front-end and back-end servers are vulnerabl request smuggling, it could make this reflected HTML injection / XSS exploitable by poisoning the cache so that the cached response with the H injection / XSS payload can be served to the victim's browser.

Steps To Reproduce:

1. Create a new tag if there's no existing tags
2. Login as an administrator user
3. Send the following cURL command to confirm `$t_redirect_page` did not get HTML entity encoded. (Replace `<your_tag_id>` , `<your_admin_user_PHPSESSID>` , and `<your_admin_user_mantis_string_cookie>` with an existing tag ID, step 2's PHP session and `MANTIS_STRING_COOKIE` cookie)

```
1 curl -s http://localhost:8080/tag_update_page.php --get \
2 --data-urlencode 'tag_id=<your_tag_id>' \
3 --cookie 'PHPSESSID=<your_admin_user_PHPSESSID>; MANTIS_STRING_COOKIE=<your_admin_user_mantis_string_co
4 --header 'Referer: http://example.com/"><h1>HTML Injection' | grep 'redirect'
```

EDIT (dregad) break long line to avoid horizontal scrolling

Additional Information:

Patch

HTML entity encode `$t_redirect_page` by using function `string_display_line` :

```
1 <input type="hidden" name="redirect" value="<?php echo string_display_line( $t_redirect_page ) ?>"/>
```

Tags: No tags attached.

Activities

dregad
2026-04-12 13:13
developer ~0070977

Thanks for the detailed report.

I confirm the possibility of HTML injection following the given steps to reproduce.

Will open an advisory and request a CVE.

dregad
2026-04-12 13:56
developer ~0070980

Advisory https://github.com/mantisbt/mantisbt/security/advisories/GHSA-6jh4-47v2-4g37 created, CVE request sent.

Please review and comment, thanks.

dregad
2026-04-12 18:13
developer ~0070982

Patch ready for review in PR https://github.com/mantisbt/mantisbt-ghsa-6jh4-47v2-4g37/pull/1

siunam
2026-04-13 00:33
reporter ~0070993

> Patch ready for review in PR https://github.com/mantisbt/mantisbt-ghsa-6jh4-47v2-4g37/pull/1

I confirmed that the vulnerability can't be reproduced after the patch. LGTM!

siunam
2026-04-13 00:47
reporter ~0070995

For the CVSS score, I think it should be 5.3:
`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`. "User Interaction" should be "Passive".

dregad
2026-04-13 04:32
developer ~0071005

I updated the CVSS score as suggested. Thanks for the review and test.

 **dregad** 2026-04-17 02:46 developer ~0071014	CVE-2026-40598 assigned

Related Changesets

MantisBT: master-2.28 b1ebc577 2026-04-12 13:22 dregad Details Diff	Escape redirect page before display to prevent XSS While this is generally not directly actionable as modern browsers will URL-encode special characters, on some specific server configurations this could poison the cache, leading to HTML injection in the user's browser. Fixes ~~0037017~~, GHSA-6jh4-47v2-4g37	**Affected Issues** ~~0037017~~
	mod - tag_update_page.php	Diff File