E-Arogya - Bug #91

Feature # 65 (New): Security Audit

[Security Audit] 26 -Cross-Site Request Forgery (CSRF)

17/04/2024 04:09 PM - Kalyan Battula

Status:	Closed	**Start date:**	17/04/2024
Priority:	Low	**Due date:**	
Assignee:	Vasudev Mamidi	**% Done:**	0%
Category:		**Estimated time:**	0:00 hour
Target version:	Security Audit	**Spent time:**	0:00 hour
Deployed In:		**Category:**	

Description

26- Cross-Site Request Forgery (CSRF)

CWE : CWE-352
Description :
Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they're currently authenticated. CSRF attacks specifically target state-changing requests, not theft of data, since the attacker has no way to see the response to the forged request. With a little help of social engineering (such as sending a link via email or chat), an attacker may trick the users of a web application into executing actions of the attacker's choosing. If the victim is a normal user, a successful CSRF attack can force the user to perform state changing requests like transferring funds, changing their email address, and so forth. If the victim is an administrative account, CSRF can compromise the entire web application.
Affected Path(s) :
https://his-core-domainservice.satragroup.in/master/hospital/location/deletelocationby-locationId/215449
*-Applicable to entire application
Impact :
Depending on the application, an attacker can mount any of the actions that can be done by the user such as adding a user, modifying content, deleting data. All the functionality that’s available to the victim can be used by the attacker. Only exception to this rule is a page that requires extra information that only the legitimate user can know (such as user’s password).
Recommendation :
Use anti-CSRF tokens:
The anti-CSRF token should be unique for each user session
The session should automatically expire after a suitable amount of time
The anti-CSRF token should be a cryptographically random value of significant length
The anti-CSRF token should be cryptographically secure, that is, generated by a strong pseudo-random number generator (PRNG) algorithm
The anti-CSRF token can be added as a hidden field for forms or within URLs (only necessary if GET requests cause state changes, that is, GET requests are not idempotent)
The server should reject the requested action if the anti-CSRF token fails validation
Implement Samesite cookie attribute and set its value to "Strict/Lax".
Reference: https://cheatsheetseries.owasp.org/cheatsheets/CrossSite_Request_Forgery_Prevention_Cheat_Sheet.html
Evidence/Proof Of Concept
Step 1: Csrf token not implemented as shown in below screenshot.

clipboard-202404171609-t7zh7.png

History

#1 - 24/04/2024 12:59 AM - Vasudev Mamidi

- Status changed from New to Resolved

#2 - 26/04/2024 12:02 AM - Karthik Daram

- Status changed from Resolved to In Progress

#3 - 02/05/2024 09:40 PM - Vasudev Mamidi

- Assignee set to Vasudev Mamidi

#4 - 03/05/2024 05:28 AM - Sivakanth Kesiraju

- Target version set to Security Audit

#5 - 15/05/2024 12:33 AM - Vasudev Mamidi

- Status changed from In Progress to Resolved

#6 - 30/09/2024 05:13 PM - Gautam Kumar

- Status changed from Resolved to Closed

Files

clipboard-202404171609-t7zh7.png	58 KB	17/04/2024	Kalyan Battula