SQl Injection in was found in “/DMS/dms/admin/add-directory.php” in PHPGurukul Directory Management System Project in PHP v2.0 allows remote attackers to execute arbitrary code via “fullname” POST request parameter.

* Official Website URL

<https://phpgurukul.com/directory-management-system-using-php-and-mysql/>

* Affected Product Name:Directory Management System Using PHP and MySQL

|  |  |
| --- | --- |
| Affected Vendor | Phpgurukul |
| Affected Code File | add-directory.php |
| Affected Parameter | fullname |
| Method | POST |
| Type | Time-based blind |
| Version | V 2.0 |

**Steps to Reproduce:**

1. **Log in to the Admin Panel:**

* Open the admin login page.
* Enter your credentials and sign

A screenshot of a computer

AI-generated content may be incorrect.

1. **Go to DMS Admin Dashboard section:**

* Navigate to the “Directory” section.
* Click the “Add Directory” section.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Move to the Add Directory Section**

* Add some details like: - Full Name, Profession, Email address, Mobile Number, City, and Address.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Intercept the Request:**

* Open Burp Suite and set your browser to use its proxy (127.0.0.1:8080).
* Go to the **Proxy** tab and turn on **the Interceptor**.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Modify the Request:**

* Capture the request when updating directory details.
* Send it to the Burp Suite Repeater.
* Modify the **fullname** parameter by injecting this payload:

'%2b(select\*from(select(sleep(20)))a)%2b'



1. **Send the Modified Request:**

* Forward the modified request in the Burp Suite Repeater.
* Observe the delay in the response time.
* The server will delay its response by 20 seconds, confirming the successful execution of the SLEEP () function, indicating a time-based SQL injection vulnerability.

A screenshot of a computer

AI-generated content may be incorrect.

**Recommended Mitigations:**

**Mitigation:** Implement proper input validation, output encoding, and Content Security Policy (CSP) to prevent malicious HTML injection.

**Impact**

* Data Theft: Unauthorized access to sensitive user or system data in the database.
* Data Manipulation: Modification or erasure of data, which destroys the integrity of data.
* Reconnaissance: Enumeration of the database structure, such as tables, columns, and schemas, for further exploitation.

**Recommended Mitigations:**

[SQL Injection Prevention - OWASP Cheat Sheet Series](https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html)