0-DAY NO CVE HIGH PUBLIC

Moodle, enrol_database concatenates course idnumber

Moodle 5.2.3 enrol_database still builds external-DB SQL with db_addslashes and string concat. auth_db was parameterized in MSA-26-0005. An editing teacher sets course idnumber. Cron interpolates it. Sink is the enrol DB, not core $DB. No CVE yet.

NAME **Moodle, enrol_database concatenates course idnumber**	TYPE **0-day analysis**
CVE **n/a**	CVE RISK **high**
DISCLOSURE STATUS **public**	VENDOR **Moodle**
AFFECTED **Moodle LMS 5.2.3 tag v5.2.3 (344232c) with enrol_database enabled. Default local map is course idnumber. Gadget lab of db_get_sql / db_addslashes.**	PUBLISHED **12 Oct 2026**
UPDATED **12 Oct 2026**	TAGS **0-day, moodle, sqli, enrol-database, cwe-89, authenticated**

TL;DR

Lab files and proof-of-concept **here**.

01 auth_db got a placeholder. enrol_database got a slash

I am @abraxas_null. The proof of concept is on GitHub: **abraxas/moodle-enrol-database-sqli** (loopback client). The lab stack is `lab/`: Dockerfile, docker-compose.yml, run.sh, poc.py, index.php. Authorized lab only. It talks to loopback.

This is **Moodle LMS 5.2.3** (`344232c15336c71b80f9aca8359ce0e0a9f3d116`). No CVE yet. CWE-89. 8.1 High (`CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N`). Remote SQLi on the **external enrolment database**. Plugin must be on. Moodle core `$DB` is not the sink. Not OS LPE.

MSA-26-0005 / MDL-88138 parameterized `auth_db`: `Execute($sql, [$extusername])`. `enrol_database` still slashes and concatenates. Same class of external DB. Same decade of `db_addslashes`. One plugin got the patch. The other kept the helper they already distrusted.

Same pin: Blackboard img file read, grades feedback IDOR, hidden grade items.

02 Why this plugin

Moodle talks to other databases when an institution already has a student-information system. `enrol_database` is that bridge. Cron reads a remote table and enrols people. The local map defaults to course **idnumber**, which an editing teacher can set. That is a SQL parameter wearing a course-settings costume.

I only wanted Critical or High. SQLi on an external enrol DB with teacher write is High. FILE / INTO OUTFILE would need that DB role. I did not demonstrate it and I am not inflating it to OS RCE.

03 Pinning v5.2.3

Same pin as the Blackboard leftover. Tag `v5.2.3`, commit `344232c`, build 20260914. MSA-26-0005 is in this tag. That is the point. The advisory closed `auth_db`. I opened the other plugin that still had `db_addslashes`.

04 The skip list

Do not re-file `auth_db`. Do not re-file MSA-26-0031 question-bank WS SQLi. Do not re-file MSA-26-0042 profile availability. Those are parameterized or whitelisted on this tag.

The rest of the MSA-26 list from the Blackboard post applies. I am not repeating it as if it were a new hunt.

05 Surfaces that returned nothing

Core `$DB` xmldb and moodle_database on 5.2.3 did not give me a teacher SQLi. Report builder audience is `is_subclass_of`. Question WS is parameterized. I was not looking for "Moodle has SQL." I was looking for the plugin MSA-26-0005 forgot.

`enrol_database` was that plugin.

06 The concat

`public/enrol/database/lib.php` `db_get_sql`:

PHP Copy

```
protected function db_get_sql($table, array $conditions, array $fields, $distinc
    $fields = $fields ? implode(',', $fields) : "*";
    $where = array();
```

```
    if ($conditions) {
        foreach ($conditions as $key=>$value) {
            $value = $this->db_encode($this->db_addslashes($value));

            $where[] = "$key = '$value'";
        }
    }
```

`db_addslashes` when `dbsybasequoting` is 0 (the default):

PHP Copy

```
$text = str_replace("'", "''", $text);
```

A backslash is not doubled. MySQL sees `\'` as an escaped quote and the string closes. Cron interpolates the mapped course field:

PHP Copy

```
$sql = $this->db_get_sql($table, array($coursefield=>$course->mapping), $sqlfield
if ($rs = $extdb->Execute($sql)) {
```

`sync_user_enrolments` does the same with `$user->$localuserfield`. Student path only if an admin maps `localuserfield=email`. Default path is teacher idnumber.

`auth_db` after MDL-88138:

PHP Copy

```
$sql = "SELECT $select
          FROM {$this->config->table}
         WHERE {$this->config->fielduser} = ?";

if ($rs = $authdb->Execute($sql, [$extusername])) {
```

That is the fix they already wrote. It is not in this file.

07 The lab

Faithful gadget of the pin builders. php:8.3-cli serves the copied `db_get_sql` / `db_addslashes` / `db_encode`. mysql:8.0 holds two enrol rows. Loopback `:19570`. Compose project `moodle-enrol-database-sqli`. Not a full Moodle PHP app. `how=gadget`.

Plugin off issued no SQL. Clean idnumber `LABCOURSE` returned one row (`teacher-local`). `LABCOURSE\' OR 1=1 --` returned two rows, including `MOODLE-ENROL-DATABASE-SQLI-WITNESS` from another course. The auth_db-style placeholder query returned zero rows and the SQL contained `?`.

PLAIN TEXT Copy

```
SUCCESS moodle-enrol-database-sqli how=gadget plugin-off=no-query clean-rows=1 ir
```

PLAIN TEXT Copy

```
python3 moodle-enrol-database-sqli-Abraxas-Labs.py
```

or `cd lab && ./run.sh`. Target only `http://127.0.0.1:19570`. Stack tears down after the run.

08 Wrong turns, in the order I made them

Treating Moodle core `$DB` as the sink. The leftover is `$extdb->Execute($sql)` on the enrol plugin connection. Core xmldb is a different hunt.

Calling this OS LPE because FILE exists as a MySQL privilege. FILE / INTO OUTFILE needs that DB role. Not demonstrated. Not this leftover.

Student email as the default map. Admin has to change `localuserfield`. Default is course idnumber. Teacher.

Plugin-off as SUCCESS. Plugin-off is the negative. No query. If the gadget still talks to MySQL with the plugin disabled, the lab is wrong.

Stacked queries. This mysqli gadget is single-statement. Extra rows are the oracle. Not a second statement.

A reverse shell. Theatre. The oracle is two rows where one course should have one, plus the witness userid.

09 What an attacker can do

An editing teacher on a site with **enrol_database** enabled injects SQL on the **external enrolment database** by setting the course **idnumber**. Cron `sync_enrolments` then runs that SQL.

Class: SQLi. **Reach:** Remote.

PRIVILEGE	METHOD / SINK
Splice a SQL fragment into the external enrol SELECT	Cron `sync_enrolments` → `db_get_sql` → `$extdb->Execute($sql)`
Read other courses' enrol rows on that external DB	Same SELECT after the string closes

PRIVILEGE	METHOD / SINK
Skip the query when the plugin is off	Same gadget, `plugin=off`
Hold the same payload as a bound parameter	auth_db leftover of MSA-26-0005: `Execute($sql, [$value])`
Enrol / unenrol / role-assign Moodle users named in extra rows	After cron applies `requestedenrols`
FILE / INTO OUTFILE as the enrol-DB user	Same sink if that account has FILE
Student-controlled path	Only if admin maps `localuserfield=email`

Closed: Moodle core `$DB`; plugin disabled; unauthenticated access; OS shell; stacked queries. Default Moodle without enrol_database is not this bug.

10 The fix

Copy auth_db's parameterized queries (MDL-88138). Pass the mapped course / user value as `Execute($sql, [$value])`. Do not concatenate `db_addslashes` into `'$value'`. Add a test that a quote in course idnumber does not return rows from other courses.

I am not going to print an idnumber you can paste at someone else's course settings. The missing placeholder is the useful part. If you own the site, run the proof of concept against loopback.

11 References

- Proof of concept: abraxas/moodle-enrol-database-sqli · moodle-enrol-database-sqli-Abraxas-Labs.py
- Lab: `lab/` · Dockerfile · docker-compose.yml · run.sh · poc.py · index.php
- @abraxas_null · github.com/abraxas · abraxaslabs.tech · abraxas.null@proton.me
- CWE-89
- Tree: moodle/moodle v5.2.3 (`344232c`) · `lib.php`
- Nearby patched (not this plugin): MSA-26-0005 / MDL-88138 `auth_db`
- Same pin: moodle-blackboard-img · moodle-grades-feedback-idor · moodle-grades-gradeitems

Abraxas Labs

Independent security research by abraxas.

Analysis of 0-days, n-days, malware, and the

leftovers that still matter.

GitHub @abraxas_null abraxas.null@proton.me abraxaslabs.tech

Home Research Open Source Reading Room About GitHub @abraxas_null Console

Independent research. Private analysis by arrangement. No ads. abraxas.null@proton.me · For defensive awareness and authorized testing only.

© 2026 Abraxas Labs