Moodle Load Monitor: Installation and configuration

This guide describes the installation of the `tool_moodleload` tool on the Moodle site **to be tested**, the preparation of the technical account used by the MoodleLoad engine and the connection to the MoodleLoad front end.

The component is at *alpha* maturity. Do not install it on a production site: it exposes system metrics, authorises load tests and creates a technical account with extended rights.

Prerequisites

- Moodle 4.3 or higher (tested on 4.5).
- PHP 8.1 or higher.
- A Linux server with a readable `/proc` file system, for CPU and memory measurements. On Windows, only some measurements are available (no load average).
- Web services enabled in Moodle, with the REST protocol.
- A MoodleLoad front end reachable from the site (URL, and service token supplied by MoodleLoad) for the status view and for launching tests.

Installation

1. Copy the `moodleload` folder into the `admin/tool` directory of your Moodle installation (final path: `admin/tool/moodleload`).
2. Log in as administrator and go to *Site administration / Notifications* to complete the installation.
3. Enable web services if they are not: *Site administration / Advanced features / Enable web services*.
4. Enable the REST protocol: *Site administration / Server / Web services / Manage protocols*.
5. Clear the caches (*Site administration / Development / Purge caches*).

Steps 3 and 4 are not done by the plugin: without them, the technical account's token is refused.

What the installation creates

The installation automatically prepares the account used by the MoodleLoad engine. It is replayed at upgrade on sites installed before this feature, and can be re-run safely.

Item	Detail
Web service	*Moodle Load Monitor* (short name `moodleload_monitor`), reserved for authorised users, with file upload enabled and downloads disabled.

Item	Detail
User	`moodleload` (“Moodle Load Test Engine”), manual authentication, unknown random password, fictitious address `moodleload@moodleload.invalid`, emails disabled. A pre-existing `moodleload` account is kept.
Authorisation	The account is added to the authorised users of the service.
Roles	**Manager** role at system level (the engine manages courses, categories and users), and a small dedicated *Moodle Load web service* role (short name `moodleloadws`) that only carries `webservice/rest:use`. Without it, the token would be refused for a user who is not a site administrator.
Token	A permanent token on the service, valid for **3 months**. It is displayed nowhere.
Course category	A top-level category *Load Tests*, named according to the site language. It serves as the default destination for the restored test courses.

The `moodleload` account has management rights: do not assign it any other role and monitor its use.

Retrieving the token

The token is read in *Site administration / Server / Web services / Manage tokens*, on the line of the `moodleload` user and the *Moodle Load Monitor* service. It is to be given to the MoodleLoad front end, never published nor sent by unencrypted email.

The token is not renewed automatically. After three months, create a new one for the same user and the same service, and update the front end.

Capabilities

Capability	Context	Default	Role
`tool/moodleload:readmetrics`	System	Manager	Read the host metrics (dashboard and `tool_moodleload_get_metrics` web service)
`tool/moodleload:setuptests`	System	Manager	Prepare the test material (file upload, course restore, bulk user creation)

The engine status view (`view.php`) is reserved for site administrators (`moodle/site:config`).

Configuration

Path: *Site administration / Plugins / Administration tools / Moodle Load Monitor settings*.

Measurements

- **CPU measurement cache lifetime** (`cache_ttl`, 5 seconds by default): how long the CPU measurement is kept in Moodle's universal cache. A shorter value gives more precise gaps but reads `/proc` more often.

Load test authorisation

MoodleLoad only launches a test on this site after a signed handshake with the plugin (web service `tool_moodleload_authorize_load`). These settings limit the site's exposure:

- **Allow load tests** (`allowload`): unticked, MoodleLoad refuses any new test and stops those in progress at their next check.
- **Maximum load** (`maxload`): maximum number of simultaneous user sessions. 0 for no limit.
- **Allowed hours** (`allowedhours`): time ranges, in server time, for example `20-6` or `12-14,20-23`. A range that ends before it starts crosses midnight. Empty for any hour.

Connection to the MoodleLoad front end

- **MoodleLoad front end URL** (`frontendurl`): base URL without trailing slash, for example `https://load.example.com`.
- **Web service token** (`token`): token supplied by MoodleLoad to query the status of the engine and jobs. It must be generated for the front end's *Moodle Load Monitoring API* service: a token of the *Moodle Load Collector API* service is refused. Do not confuse it with the token of the `moodleload` account of this site.
- **Maximum request timeout** (`timeout`, 10 seconds by default): how long to wait for the front end's response.

Front end HTTP authentication (optional)

To be filled in only if the front end is itself protected by HTTP authentication (a pre-production behind a `.htpasswd`, for example). Leave the user empty otherwise.

- **HTTP user** and **HTTP password**.
- **Method**: *Basic* (default), *Digest* or *Automatic* (negotiated by curl, at the cost of an extra request).

If the front end is on an internal network, allow its host and port in Moodle's security settings (`curlsecurityblockedhosts` and `curlsecurityallowedport`).

Checking the installation

1. **Dashboard**: open the tool page (*Site administration / Server* or plugin link): the host metrics are displayed and refreshed. The first CPU reading is empty (value −1): two measurements about 5 seconds apart are needed.
2. **Web service**: with the `moodleload` account's token, call the measurement function. The response contains the timestamp, the CPU, the memory and the load average.

```
GET
/webservice/rest/server.php?wstoken=<TOKEN>&wsfunction=tool_moodleload_get_metrics&moodlewsrestformat=json
```

1. **Status view**: *Site administration / Development / MoodleLoad load tester: engine and injector status*: the engine status and the token owner's jobs are displayed if the front end URL and

token are correct.

Troubleshooting

Symptom	Probable cause
Token refused	Web services or REST not enabled; token expired (3 months); `moodleload` user not authorised on the service.
CPU response at −1	First call: call again after a few seconds.
No memory / CPU measurements	Non-Linux host or unreadable `/proc`.
Status view: HTTP 401	Front end HTTP authentication: check user, password and method.
Status view: host blocked	Front end on an internal network: adjust `curlsecurityblockedhosts` and `curlsecurityallowedport`.
Status view: token refused by the front end	Token generated for the wrong front end service (Collector instead of Monitoring API).
Test functions missing after an upgrade	Visit the administration notifications (the service definition is updated), then purge the caches.

Upgrade and uninstallation

After each update of the folder, go through the administration notifications then purge the caches. Updates that modify the service definition (addition of functions, file upload) require it.

At uninstallation, the plugin removes the tokens, the service authorisation, the role assignments, the dedicated role, the *Load Tests* category if it is empty, the uploaded test files and the `moodleload` account if it created it itself. A pre-existing account is not deleted: only what the plugin added to it is removed.

Back to the component index - Back to plugins index - Back to catalogue

From:
https://docsen.activeprolearn.com/ - **Documentation Moodle ActiveProLearn**

Permanent link:
https://docsen.activeprolearn.com/doku.php?id=tool:moodleload:install

Last update: **2026/11/13 20:37**