

# VMoodle (local) : Installation



## Introduction

**Warning** : Setting up a virtualized array of moodle assumes you have a good knowledge of moodle administration and installation, and a correct background in LAMP application administration.

ActiveProlearn can provide full support in helping building, setting up and training exploitation staff to master the architecture.

## Installation

What you basically need to virtualize a Moodle installation is:

- Install your moodle using a non virtualized codebase as a main moodle instance.
- Add/Install the VMoodle plugin.
- Add/Install VMoodle block and report plugins.
- Configure a local/vmoodle/vconfig.php on the local/vmoodle/vconfig-dist.php base.
- Plug in to your main configuration script the VMoodle trap sequence.
- Report some patches in the /mnet implementation if you are planningn using MNET and want to get MNET stable over time.

## Components installation

- Download the zips from the repositories (local\_vmoodle, block\_vmoodle and report\_vmoodle) and extract respectively the archives in '/local', '/blocks' and '/report' directory in your moodle.
- Browse to the Administration Notifications to terminate installation.

After plugins have been installed, your moodle is still NOT capable to answer to virtual domains. Your virtualized installation keeps working a s standard moodle.

Note that we DO NOT support any more the pre-28 version.

## Virtualisation file Configuration

This file is located in `/local/vmoodle` as `vconfig.php` and is responsible of the configuration switch. The configuration settings allow you to point where the vmoodle definition register is located. Usually you wil use the database of your primary installation to store the main register. It is also

possible to move this register to another DB instance as long as the name of the table `{PREFIX}_local_vmoodle` is conserved.

A template file is provided as `<moodleroot>/local/vmoodle/vconfig-dist.php`. Copy this file as `vconfig.php` and change adequate values inside. The `vconfig.php` could be relocated anywhere in your server as long as the main `config.php` is able to include it.

In most case, you will simply have to report your main `dbsettings` in this file.

## Update the main configuration file of Moodle

The virtualisation trap need to be set in the master configuration of Moodle as any entrance page of moodle will call it at startup.

Updates are needed inorder to:

- hook the configuration switch for standard use
- add special conditionnality for CLI script (VMoodle enabled scripts)
- Configure some network related behaviour when using MNET cross-services.

Hooking the `vconfig.php` will be the point where the current configuration (working database and moodledata volume) will switch.

### Hooking the virtual configuration switch

The main point about this hooking is that it **MUST** be placed before calling the `setup.php` include in your `config.php` file.

```
... {CLI VMoodle Hook} ...
```

```
require(</path/to/moodleinstall>/local/vmoodle/vconfig.php');
```

```
require_once(dirname(__FILE__) . '/lib/setup.php');
```

### Adding the CLI hook

#### What issue it solves:

Standard cli script just play the operations loading the standard configuration. VMoodle switches the configuration dynamically analysing the current HTTP required hostname or URL. In CLI mode you do NOT have this information in the execution environment and the admin needs to provide it in some way.

The CLI VMoodle hooking allows the following process to occur:

- The CLI script starts and loads an incomplete moodle configuration (just getting the vmoodle register db description),
- The script receives the command line parameters in which a `--host` additional option has been

added.

- The script reloads the full configuration file till the end, activating the virtualisation switch.

Add the following code to your `config.php` file before including `vconfig.php`:

```
// this fragment will trap the CLI scripts trying to work for a virtual
node, and
// needing booting a first elementary configuration based on main config
if (isset($CLI_VMOODLE_PRECHECK) && $CLI_VMOODLE_PRECHECK == true) {
    $CLI_VMOODLE_PRECHECK = false;
    return;
}
```

**Important** : Only adapted CLI script will be able to process data in virtual submoodles. Most of the standard admin scripts have been adapted and are provided in the `/local/vmoodle/cli` directory. In additions we added for each standard script a bulk script for launching each script on the complete register.

Note : use the `--help` command-line attribute to see the options.

## Routing the instance domains to the single moodle install

As the physical install is unique, you will have to configure your apache or nginx virtual hosts to converge to this single location. The easiest way to do so is:

- In apache : Adding `ServerAlias` clauses to the `VHost` definition
- On nginx : Adding `servername` additional instructions to the server structure routing the moodle root.

If you have one single domain for all your moodles and your web server configuration and DNS supports wildcard, you can wildcard the subdomain as `*.mymoodles.edu` for convenience.

## Global configuration keys having an influence on the VMoodle behaviour

```
$CFG->mainhostprefix = 'http://someprefixthatmatchsmymainmoodle';
```

This key allows any instance to know which host matches the master install of moodle. We use this for some features such as f.e. identifying who is the “master network admin account” of the whole plant.

```
$CFG->forced_plugins_settings['user_mnet_hosts']['admin_override'] = true;
```

This key will allow to force the [User Mnet Hosts](#) blocks to let the local admins go through the network links. Usually we will forbid local subadmins to roam, but the master global admin should be able to. A sample of this settign is possible in `vconfig.php` :

```
$CFG->forced_plugins_settings['user_mnet_hosts']['admin_override'] = false;
if (preg_match('#'.$CFG->mainhostprefix.'#', $CFG->wwwroot)) {
    $CFG->forced_plugins_settings['user_mnet_hosts']['admin_override'] =
```

```
true;  
}
```

Using this configuration, the master moodle admin will be allowed to roam to any submoodle. The local subadmins will not. As roaming to another moodle creates an image of the user account, multiplying the admin account could lead to some confusing states.

```
$CFG->mnetsiteadmins = true;
```

This key allows remote users to become site administrators of the local moodle. This is forbidden by standard Mnet rules but VMoodle implementation propose patches to override.

## Mise en oeuvre du cron virtualisé

### Mise en place à partir de cron simples

Un plate-forme Moodle fonctionne avec une tâche cron associée qui effectue tout un ensemble de tâches automatiques et cadencées sur les données de Moodle.

La mise en oeuvre de plates-formes virtualisées nécessite également de mettre en oeuvre ces tâches pour chacune des instances virtuelles qui sont créées dans la virtualisation.

En HTTP, vous pouvez enregistrer vos tâches cron sur les domaines d'exploitation propres de vos instances, comme vous le feriez pour des plates-formes standard. Par contre, pour une mise en oeuvre par script serveur utiliser le script standard `/admin/cli/cron.php` n'est pas possible, car ce type d'appel ne fournit aucune informations permettant au basculement virtuel de se faire.

Pour obtenir un basculement effectif de l'instance de service, utilisez à la place le script :

```
/local/vmoodle/cli/cron.php --host=http://my_virtual.mymoodledomain.com
```

en précisant explicitement la racine wwwroot que vous voulez adresser.

### Utilisation de l'ordonnanceur de cron

VMoodle fournit en plus un ordonnanceur de cron qui est lancé sur l'installation principale, et est capable de faire "tourner" automatiquement le déclenchement des crons virtuels de toutes les instances. Vous pouvez utiliser ce script en mode HTTP ou en mode script :

```
wget http://main.mymoodledomain.com/local/vmoodle/vcron.php
```

ou

```
sudo -uwww-data php /root/to/moodle/local/vmoodle/cli/vcron.php
```

La fréquence de rotation de l'ordonnanceur de cron doit être très rapide afin de permettre que chaque site virtuel soit activé suffisamment souvent.

[back to VMoodle component index](#) - [Back to plugins](#) - [Back to catalogue](#)

From:

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

Permanent link:

<https://docsen.activeprolearn.com/doku.php?id=local:vmoodle:install&rev=1519227063>

Last update: **2024/04/04 15:50**

