

# 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 planning 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 as a 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 will 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.

## Adapter le fichier de configuration principal

Tout le travail de la virtualisation s'effectue dans le fichier de configuration principal de Moodle qui est toujours le premier fichier invoqué par toute "porte d'entrée" de Moodle.

Les modifications portent sur :

- Le déROUTAGE de virtualisation
- La mise en place du déROUTAGE pour les scripts CLI compatibles
- Des options de comportement de la plate-forme dans le cadre de réseaux de plates-formes.

le fichier `vconfig.php` défini précédemment effectue la boucle de reconfiguration virtuelle permettant de changer dynamiquement la base de données et le dossier de fichiers actif.

### Dérouter pour la virtualisation

Cette inclusion doit être placée juste après l'appel au "setup.php" de votre fichier de configuration.

```
... {position du déROUTAGE CLI} ...
```

```
require(</chemin/vers/monmoodle>/local/vmoodle/vconfig.php');
```

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

### Mette en place le déROUTAGE CLI

Le déROUTAGE des scripts CLI permet de jouer une partie du fichier de configuration, puis rendre la main au script CLI appelant, recevoir l'hôte cible dans les paramètres de ligne de commande, puis revenir enfin rejouer la configuration complète sur la nouvelle configuration virtualisée.

Ajoutez la séquence suivante avant votre appel à `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;
}
```

```
}
```

## Clefs de configuration techniques ayant un impact sur le fonctionnement de VMoodle

```
$CFG->mainhostprefix = 'http://physique.monmoodle';
```

Cette clef permet d'expliciter quel est le Moodle principal à tout moment. Ceci est utile pour certains composants associés à la virtualisation qui doivent reconnaître, une fois leur configuration acquise, s'il sont le Moodle principal ou s'ils sont virtuels.

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

Cette clef permet de forcer le bloc [User Mnet Hosts \(Mes hôtes du réseau\)](#) à laisser exceptionnellement passer les administrateurs principaux locaux (compte admin) à travers le réseau. Un exemple courant d'utilisation est après l'appel à `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;
}
```

Dans cette configuration, seul l'administrateur local de la plate-forme principal pourra visiter les autres plates-formes. Les administrateurs locaux des Moodle virtuels se verront afficher un refus de circuler. La raison de cette précaution est que la circulation des utilisateurs réplique les comptes dans les plates-formes visitées. Ceci peut conduire à de nombreuses confusions que de détenir dans toutes les plates-formes les administrateurs locaux de toutes les autres.

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

Cette clef permet de modifier le comportement de la pile MNET en autorisant des comptes externes (réseau) à être administrateurs de site localement.

## 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/> - **Documentation Moodle ActiveProLearn**

Permanent link:  
<https://docsen.activeprolearn.com/doku.php?id=local:vmoodle:install&rev=1519216703>

Last update: **2026/01/13 07:58**

