

# Prérequis pour la formation

Sur l'environnement Microsoft Windows

Glenn ROLLAND / glenux@glenux.net

## Installation des outils

### Installer Chocolatey

Chocolatey permet de faciliter l'installation de logiciels et fournir une pré-configuration notamment pour l'usage depuis la ligne de commande.

#### Procédure d'installation

1. Ouvrir <https://chocolatey.org/install> avec votre navigateur ;
2. Copier la ligne de code pour `cmd.exe` ;
3. Depuis le menu démarrer, lancer `cmd.exe` en tant qu'administrateur ;
4. Coller la ligne de code copiée.
5. Attendre le message de succès de Chocolatey.

Ensuite fermer la fenêtre du shell, car `cmd.exe` doit redémarrer pour prendre les changements en compte.

## Installation des outils

### Installer Git

Git permet de versionner les différentes code source et configuration. Il fournit également une version de `Bash` pour Microsoft Windows qui sera utile pour les commandes suivantes.

### Procédure d'installation

1. Depuis le menu démarrer, lancer `cmd.exe` en tant qu'administrateur ;
2. Taper ensuite la commande suivante :

```
choco install git
```

## Installation des outils

### Installer Virtualbox

Virtualbox est le gestionnaire de virtualisation pour lesquels les exemples du cours ont été préparées :

### Procédure d'installation

1. Depuis le menu démarrer, lancer `cmd.exe` en tant qu'administrateur ;
2. Taper ensuite la commande suivante :

```
choco install virtualbox
```

**Important : il est important de désactiver les modules (drivers) spécifiques à Microsoft Hyper-V et ceux VMWare** qui pourraient empêcher le bon fonctionnement de la virtualisation (selon ce qui est installé sur la machine).

## Installation des outils

### Installer Vagrant

Vagrant est un outil de gestion de machines virtuelles. Il nous permettra de charger automatiquement des images virtuelles avec différents systèmes (et paramètres) et d'accéder aux machines virtuelles comme si elle était en réseau via SSH.

### Procédure d'installation

1. Depuis le menu démarrer, lancer `cmd.exe` en tant qu'administrateur ;
2. Taper ensuite la commande suivante :

```
choco install vagrant
```

## Pour vérifier le bon fonctionnement

### Préparer une VM

- Créer un dossier `cours-test` sur le bureau.
- Télécharger le fichier suivant  
<https://bit.ly/3ephXn6>
- Mettre le fichier dans le dossier sous le nom `Vagrantfile` (sans extension !)
- Ouvrir le dossier `cours-test`
- Cliquer sur le fond du dossier, avec le bouton droit de la souris, et choisir `Git Bash here...` dans le menu
- Un terminal s'ouvre, avec un shell `Bash`

## Pour vérifier le bon fonctionnement

### Lancer la VM

1. Taper ensuite la commande :

```
vagrant up
```

2. Un certain nombre de lignes s'affichent...

```
~/minimal
$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'debian/buster64'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'debian/buster64' version '10.0.0' is up to date...
==> default: Setting the name of the VM: minimal_default_1586955960808_72100
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
default: Adapter 1: nat
==> default: Forwarding ports...
default: 80 (guest) => 8880 (host) (adapter 1)
default: 22 (guest) => 2222 (host) (adapter 1)
==> default: Running 'pre-boot' VM customizations...
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
default: SSH address: 127.0.0.1:2222
default: SSH username: vagrant
default: SSH auth method: private key
default:
default: Vagrant insecure key detected. Vagrant will automatically replace
default: this with a newly generated keypair for better security.

[...]

default: Enabling module authz_core.
default: Enabling module authz_host.
default: Enabling module authn_core.
default: Enabling module auth_basic.
default: Enabling module access_compat.
default: Enabling module authn_file.
default: Enabling module authz_user.
default: Enabling module alias.
default: Enabling module dir.
default: Enabling module autoindex.
default: Enabling module env.
default: Enabling module mime.
default: Enabling module negotiation.
default: Enabling module setenvif.
default: Enabling module filter.
default: Enabling module deflate.
default: Enabling module status.
default: Enabling module reqtimeout.
default: Enabling conf charset.
default: Enabling conf localized-error-pages.
default: Enabling conf other-vhosts-access-log.
default: Enabling conf security.
default: Enabling conf serve-cgi-bin.
default: Enabling site 000-default.
default: Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib.
default: Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.ser
default: Processing triggers for systemd (241-5) ...
default: Processing triggers for man-db (2.8.5-2) ...
default: Processing triggers for libc-bin (2.28-10) ...

==> default: Machine 'default' has a post 'vagrant up' message. This is a message
==> default: from the creator of the Vagrantfile, and not from Vagrant itself:
==> default:
==> default: Vanilla Debian box. See https://app.vagrantup.com/debian for help and bug reports
~/minimal
$
```

## Pour vérifier le bon fonctionnement

### Se connecter

1. Ouvrir le lien <http://127.0.0.1:8880> dans un navigateur.
2. Cela affiche le texte \_« C'est bon, ça fonctionne ! » \_





**Merci pour la préparation !**

Pour toute question :

**Glenn ROLLAND**

Email : [glenux@glenux.net](mailto:glenux@glenux.net)

Tel : +33 6 739 839 56