

Déploiement d'un site web sur un ensemble de machines virtuelles avec Proxmox

Olivier Martin

Janvier 2024

Premier TD de virtualisation pour le module R4.08, BUT Informatique deuxième année.

Travail en binôme en salle de machines.

Compétence travaillée : Installer, configurer, mettre à disposition, maintenir en conditions opérationnelles des infrastructures, des services et des réseaux et optimiser le système informatique d'une organisation – niveau 2 : Déployer des services dans une architecture réseau – Utiliser des serveurs et des services réseaux virtualisés.

Pré-requis :

- Utilisation de la ligne de commande Unix (shell Bash)
- Notions de virtualisation
- Notions sur les réseaux TCP/IP
- Architecture client-serveur
- MySQL : droits des utilisateurs

Matériel ou logiciels utilisés :

- Accès au serveur Proxmox de l'Université depuis un navigateur web.
- Client SSH, par exemple PuTTY ou MobaXterm (optionnel)

Table des matières

1	Présentation du TD	2
1.1	Déroulement	2
1.2	Choix de Wordpress	2
1.3	Organisation du temps	3
2	Création du patron pour Ubuntu Server	3
3	Installation de Wordpress sur une VM	3
3.1	Étapes du déploiement	4
3.1.1	Installation des dépendances	4
3.1.2	Installation du programme de Wordpress	5

3.1.3	Configuration du serveur Apache pour Wordpress	5
3.1.4	Configuration de la base de données MySQL	6
3.1.5	Paramétrage de Wordpress pour la base de données	6
3.1.6	Lancement et configuration de Wordpress	7
3.1.7	Tests et rédaction du premier article	8
3.2	Migration sur site vers un autre serveur Proxmox	9
4	Déploiement du site sur plusieurs VM	10
4.1	Motivation	10
4.2	Étapes	11
4.2.1	Modifications à apporter à r408-apache	11
4.2.2	Instanciation et paramétrage du serveur r408-sql	12
4.2.3	Installation et configuration de MySQL sur vw-sql	13
4.2.4	Extraction des données SQL du serveur r408-web	13
4.2.5	Création des données SQL sur le nouveau serveur	14
4.3	Configuration de Wordpress pour deux serveurs	14
4.3.1	Création du réseau privé	14
4.3.2	Configuration de Wordpress et de MySQL, essais	15
5	Conclusion	16

1 Présentation du TD

1.1 Déroutement

Dans ce TD vous allez déployer un site web dynamique dont le programme est en langage PHP et les données sont enregistrées sur une base de données MySQL.

Dans un premier temps (section 2) vous allez créer un patron (*template*) qui vous servira à instancier des machines virtuelles (VM) sans refaire la phase d'installation .

Puis, à la section 3 vous créez une instance virtualisées de Ubuntu Server et y déployez le site Wordpress.

À la section 4 vous réutiliserez le travail de la section précédente pour déployer le site sur deux VM qui seront reliées par un réseau dédié.

1.2 Choix de Wordpress

Le logiciel à déployer, Wordpress, est un système de gestion de contenu (CMS - *Content Management System*) en ligne. Il permet à chacun de diffuser des articles sur le web sans connaissance technique particulière. Le site dispose d'une partie accessible sans connexion, qui permet au public de consulter les articles, d'une partie administration qui permet de le configurer (création de comptes utilisateurs, installation d'extensions, etc.), et d'une partie pour les auteurs qui peuvent rédiger leurs articles et les mettre en page avec une interface graphique.

Ce qui nous intéresse pour ce TD ce n'est pas sa fonctionnalité mais les technologies utilisés

par Wordpress : pages web en HTML 5 (avec CSS pour le style et Javascript pour le code exécuté sur le navigateur), code côté serveur en PHP, et données enregistrées sur une base SQL.

L'interpréteur PHP est appelé directement par le serveur web, il doit donc être installé sur le même serveur. Par contre la base de données n'est pas nécessairement hébergée par la même machine que l'interpréteur PHP. La communication entre PHP et SQL se fait par réseau TCP/IP, nous verrons le cas où les deux éléments sont déployés sur la même machine et le cas où ils sont sur deux VM différentes.

1.3 Organisation du temps

Au début de chaque TP ou TD on vous demande généralement de lire l'ensemble des instructions avant de commencer le travail. Pour ce TD oubliez ces conseils ! En effet la phase d'installation peut prendre un peu de temps et vous aurez tout loisir de lire le sujet en détail pendant que le système s'installe.

Commencez donc par créer le *template* (en français "patron") pour les VM utilisées en suivant les instructions de la section 2, puis lisez les instructions des sections 3 et 4 avant de déployer le système en suivant ces instructions.

Cette organisation du temps vous permettra de terminer le TD dans les deux heures imparties.

2 Création du patron pour Ubuntu Server

Les instructions pour créer un patron qui vous servira à instancier rapidement plusieurs VM sur le même modèle sont données dans le document "Création d'un patron pour machine virtuelle avec Proxmox". Suivez les instructions de ce document et créez un patron, puis reprenez les instructions du présent document.

3 Installation de Wordpress sur une VM

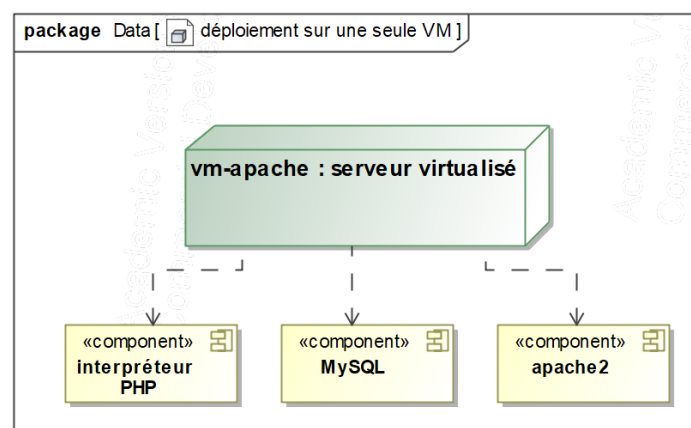


FIGURE 1 – Déploiement du site Wordpress sur une seule VM

Le diagramme UML de déploiement donné à la figure 1 décrit le système que vous allez déployer dans cette section. Notez qu'il n'y a qu'une seule VM qui supporte les différents modules logiciels composant le système.

À propos des noms de machine : Dans les instructions qui suivent on vous demande de créer une VM et de lui attribuer un nom. Pour s'y retrouver sur le serveur Proxmox parmi les VM créées par les étudiants, il faut que ce nom soit unique. Vous signerez donc le nom de la machine avec vos initiales, en vérifiant qu'aucun autre étudiant du groupe n'a les mêmes initiales que vous.

Par exemple si vous vous appelez Olivier Martin, la VM pour cette partie doit s'appeler r408-web suivi de la signature -om, son nom complet est donc r408-web-om. Dans le reste du document on l'appellera simplement r408-web, mais n'oubliez pas que pour vous le nom comporte toujours une signature.

3.1 Étapes du déploiement

Les étapes suivantes sont adaptées du document <https://ubuntu.com/tutorials/install-and-configure-wordpress> sur le site officiel de Ubuntu. Sauf indications contraires elles s'exécutent dans un shell sur le serveur. Connectez-vous au serveur Ubuntu soit à travers l'interface de Proxmox (déconseillé), soit par une connexion SSH (fortement conseillé car il va y avoir beaucoup de copier-coller à faire). Votre utilisateur doit disposer des droits de sudo.

Les différentes instructions doivent s'exécuter sans erreur. Si ce n'est pas le cas vérifiez les commandes que vous avez tapées (attention le copier-coller depuis un fichier PDF crée parfois des surprises) et essayez de résoudre les problèmes par vous-même avant d'appeler l'enseignant chargé de TD à l'aide. Dans tous les cas ne restez pas à attendre que le problème se résolve tout seul, ça n'arrivera pas.

3.1.1 Installation des dépendances

Installez Apache2, PHP et les modules nécessaires à Wordpress avec la commande suivante :

```
sudo apt update && sudo apt install apache2 \  
ghostscript libapache2-mod-php \  
mysql-server php php-bcmath php-curl \  
php-imagick php-intl php-json \  
php-mbstring php-mysql php-xml php-zip
```

Le *backslash* '\' à la fin d'une ligne indique au shell que l'instruction n'est pas terminée. Vous pouvez formater cette commande sur une seule ligne si vous le souhaitez.

La double esperluette '&&' entre les deux appels à sudo indique que le deuxième appel ne doit être fait que si le premier réussit. Notez que l'évaluation paresseuse de l'opérateur '&&' en C++ suit le même principe.

Détail des paquets Ubuntu installés par cet appel :

- apache2 : le serveur Apache, qui implémente le protocole HTTP

- `ghostscript` : un utilitaire pour produire des fichiers PDF, appelé par le code PHP.
- `mysql-server` : serveur de base de données relationnelles. Ce paquet installe aussi le client en ligne de commande.
- `php` : langage de programmation interprété, conçu pour la génération de pages web dynamiques.
- `libapache2-mod-php` : module pour Apache qui lui permet d'exécuter des commandes en PHP.
- `php-mysql` : module pour le langage PHP qui permet de se connecter et d'envoyer des requêtes à un serveur MySQL.
- Les autres paquets en `php-*` sont des modules pour l'interpréteur PHP, nous ne les détaillerons pas.

Après cette étape vous pouvez vous connecter à la machine depuis un navigateur web, vous obtenez alors une page par défaut de Apache.

3.1.2 Installation du programme de Wordpress

Les commandes shell suivantes créent un répertoire pour les données du serveur web, puis téléchargent la dernière version du programme Wordpress et les décompressent directement dans ce répertoire. L'utilisation des tubes '|' (ou *pipes*) évitent la création d'un fichier temporaire.

```
sudo mkdir -p /srv/www
sudo chown www-data: /srv/www
curl https://wordpress.org/latest.tar.gz | \
  sudo -u www-data tar zx -C /srv/www
```

Avant de passer à l'étape suivante vérifiez que le répertoire `/srv/www` contient un répertoire nommé `wordpress` qui contient, entre autres, des fichiers `*.php`

3.1.3 Configuration du serveur Apache pour Wordpress

Créez, en tant que root et avec l'éditeur de votre choix, un fichier `/etc/apache2/sites-available/wordpress.conf` qui doit contenir le texte suivant :

```
<VirtualHost *:80>
  DocumentRoot /srv/www/wordpress
  <Directory /srv/www/wordpress>
    Options FollowSymLinks
    AllowOverride Limit Options FileInfo
    DirectoryIndex index.php
    Require all granted
  </Directory>
  <Directory /srv/www/wordpress/wp-content>
    Options FollowSymLinks
    Require all granted
  </Directory>
</VirtualHost>
```

Vérifiez les majuscules et minuscules dans ce fichier avant de sauvegarder, puis activez le site avec `sudo a2ensite wordpress`

Activez les URL raccourcies (sans le `index.php?` ou équivalent) : `sudo a2enmod rewrite`

Désactivez le site par défaut : `sudo a2dissite 000-default`

Redémarrez Apache avec `sudo service apache2 reload` pour prendre en compte les modifications. Si vous vous connectez au serveur avec un navigateur web, vous devez maintenant avoir un message d'erreur car ni Wordpress ni la base de données ne sont configurés.

3.1.4 Configuration de la base de données MySQL

Dans cette étape vous allez vous connecter à la base de données locale au serveur avec `sudo mysql`.

Les commandes qui suivent sont à exécuter dans le client en ligne de commande pour MySQL que vous venez de démarrer ; ce ne sont pas des commandes shell. Après chaque commande le client doit afficher un message indiquant qu'elle a réussi, par exemple "Query OK, 1 row affected (0,00 sec)".

Ne passez pas à la commande suivante tant que la commande en cours n'a pas été prise en compte correctement.

1. Création de la base "wordpress" sur le serveur :

```
CREATE DATABASE wordpress;
```

2. Création du compte utilisé par Wordpress pour se connecter à la base. Choisissez un mot de passe, notez-le, et remplacez la chaîne `<your-password>` par ce mot de passe dans la requête suivante :

```
CREATE USER wordpress@'%' IDENTIFIED BY '<your-password>';
```

3. Attribution de droits à l'utilisateur précédemment créé. L'utilisateur dispose de la plupart des droits sur la base "wordpress", y compris celui de créer des tables car pour l'instant la base est vide :

```
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,ALTER ON wordpress.* TO  
wordpress@'%' ;
```

4. Les droits suivants ne sont pas nécessaires pour Wordpress, mais nous en aurons besoin plus tard dans le TD (avec `mysqldump`) alors nous les donnons maintenant à l'utilisateur :

```
GRANT SHOW VIEW, TRIGGER, LOCK TABLES ON wordpress.* TO wordpress@'%' ;  
GRANT PROCESS ON *.* TO wordpress@'%' ;
```

5. Prise en compte immédiate des droits donnés à l'utilisateur :

```
FLUSH PRIVILEGES;
```

Vous pouvez maintenant quitter le client MySQL avec `quit;`

3.1.5 Paramétrage de Wordpress pour la base de données

Commencez par créer le fichier de configuration `wp-config.php` en recopiant le fichier donné en exemple :

```
sudo -u www-data cp /srv/www/wordpress/wp-config-sample.php \  
/srv/www/wordpress/wp-config.php
```

Ouvrez ce nouveau fichier dans un éditeur de texte de votre choix, en tant que root sinon vous ne pourrez pas sauvegarder. Apportez les modifications suivantes :

- Recherchez le texte `database_name_here` et remplacez-le par `wordpress`
- Recherchez le texte `username_name_here` et remplacez-le par `wordpress`
- Recherchez le texte `password_here` et remplacez-le par le mot de passe choisi à l'étape 3.1.4

Toujours dans le fichier `wp-config.php`, recherchez le texte :

```
define( 'AUTH_KEY',          'put your unique phrase here' );  
define( 'SECURE_AUTH_KEY',   'put your unique phrase here' );  
define( 'LOGGED_IN_KEY',     'put your unique phrase here' );  
define( 'NONCE_KEY',        'put your unique phrase here' );  
define( 'AUTH_SALT',        'put your unique phrase here' );  
define( 'SECURE_AUTH_SALT',  'put your unique phrase here' );  
define( 'LOGGED_IN_SALT',    'put your unique phrase here' );  
define( 'NONCE_SALT',       'put your unique phrase here' );
```

Vous allez supprimer ces lignes et les remplacer par le texte généré aléatoirement par la page <https://api.wordpress.org/secret-key/1.1/salt/>

Si vous êtes connecté à la VM par l'interface de Proxmox, ça risque d'être un peu laborieux ! Dans ce cas, pour éviter de tout recopier à la main, vous pouvez télécharger la page avec `curl` et l'enregistrer sur votre disque local¹ puis ouvrir les deux fichiers dans votre éditeur et copier-coller de l'un à l'autre. Si votre éditeur ne permet pas l'ouverture simultanée de deux fichiers, vous pouvez ajouter le contenu de `salt.txt` à la fin de `wp-config.php`², en ayant pris soin de fermer `wp-config.php` dans l'éditeur. Il y a toujours une solution...

N'oubliez pas d'enregistrer `wp-config.php` avant de passer à l'étape suivante.

3.1.6 Lancement et configuration de Wordpress

Vous pouvez maintenant lancer un navigateur web sur votre station de travail et saisir l'adresse du serveur. Une page de configuration de Wordpress doit s'afficher, qui vous demande de saisir la langue pour la configuration. Procédez ensuite à la configuration (voir figure 2) du site en répondant aux questions. Notez le mot de passe de l'administrateur, il vous servira plus tard.

Après avoir validé la configuration, vous êtes redirigé vers une page de connexion. Saisissez les identifiants que vous avez créé dans la page précédente et validez, vous allez être redirigé vers la page d'administration du site, appelée *dashboard*.

Ce premier démarrage peut prendre un peu de temps car le programme crée les tables sur le serveur MySQL dans la base "wordpress" et y insère les premières données. Vous pouvez le

1. Par exemple avec `curl https://api.wordpress.org/secret-key/1.1/salt/ > ~/salt.txt`

2. La commande doit ressembler à `cat ~/salt.txt | sudo tee -a /srv/www/wordpress/wp-config.php`

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Do not worry, you can always change these settings later.

Site Title

Username
 Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password
Strong
Important: You will need this password to log in. Please store it in a secure location.

Your Email
 Double-check your email address before continuing.

Search engine visibility Discourage search engines from indexing this site
 It is up to search engines to honor this request.

FIGURE 2 – Paramétrage de Wordpress

vérifier en vous y connectant et en exécutant la requête :
SHOW TABLES FROM wordpress;

Après quelques instants la page d'administration s'ouvre, comme à la figure 3.

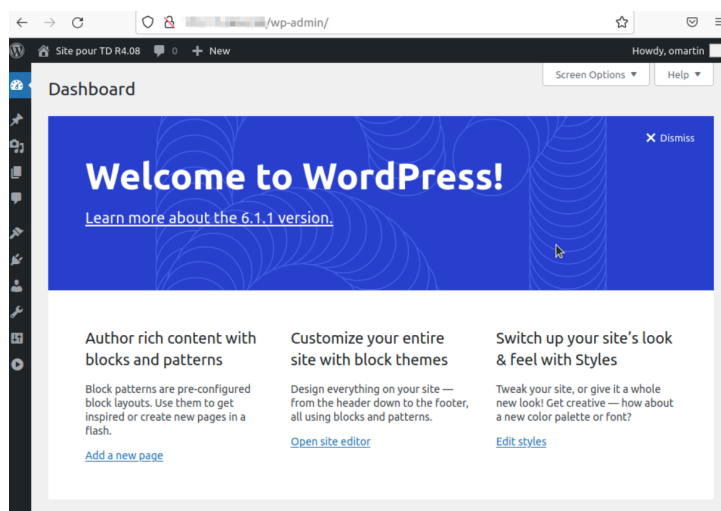


FIGURE 3 – Page d'administration de Wordpress

3.1.7 Tests et rédaction du premier article

Cliquez sur "Add New" du menu "Posts" (figure 4) pour créer un nouvel article.

Vous pouvez maintenant rédiger un article avec une interface qui ressemble à un traitement de texte. Cliquez sur "Publish" pour publier l'article (figure 5)

Il est possible de basculer entre la page d'administration et la consultation du site en cliquant sur "Dashboard" (figure 6) ou "Visit Site" (figure 7)

Cette fonctionnalité, et l'ensemble du *dashboard* ne sont disponibles que parce que vous êtes

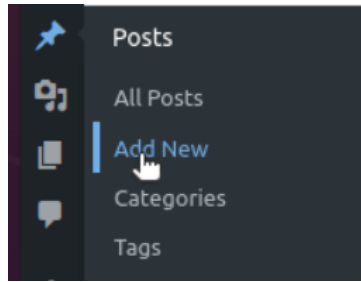


FIGURE 4 – Menu de création d'un nouvel article

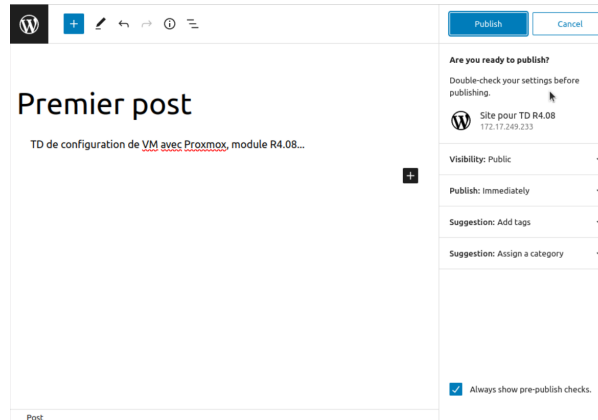


FIGURE 5 – Rédaction et publication d'un article

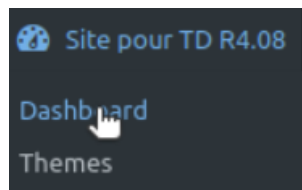


FIGURE 6 – Basculement vers le mode administration

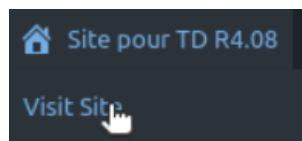


FIGURE 7 – Basculement vers le mode consultation du site

connecté en tant qu'administrateur. Déconnectez-vous avec le menu en haut à droite de la page. Vous pouvez toujours accéder au site mais les menus ne s'affichent plus.

3.2 Migration sur site vers un autre serveur Proxmox

Un des intérêts de la virtualisation est la possibilité de migrer une VM d'un hôte à un autre de manière transparente pour le programme exécuté, et potentiellement pour l'utilisateur.

Proxmox permet de migrer une VM d'un serveur à un autre, à condition que les deux serveurs soient regroupés sous le même *cluster*. Le service Proxmox de l'université est composé de quatre serveurs, appelés pve1, pve2, pve3 et pve4. Nous allons faire migrer la machine

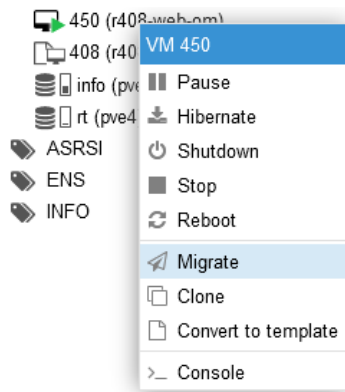


FIGURE 8 – Menu de migration d'une VM

précédemment créée d'un serveur à un autre.

Un clic droit sur le nom de la VM ouvre le menu contextuel (figure 8). Cliquez sur "Migrate".

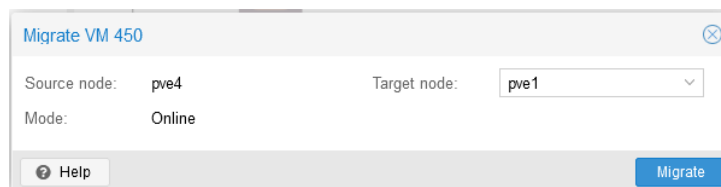


FIGURE 9 – Paramètres de la migration

Le menu de migration s'ouvre (figure 9). Choisissez un serveur cible différent du serveur source et cliquez sur "Migrate". Une boîte de dialogue s'ouvre et affiche des informations sur la migration.

Si la migration est réussie la VM est dans l'arborescence du nouveau serveur et plus dans celle de l'ancien. L'icône avec un triangle vert indique qu'elle est en cours d'exécution.

Ouvrez le navigateur web et connectez-vous au site Wordpress, ou rafraichissez la page si elle était déjà ouverte. Le site doit s'afficher normalement. Il est hébergé par une nouvelle machine mais son adresse IP est restée la même.

4 Déploiement du site sur plusieurs VM

Dans cette deuxième partie nous allons héberger le serveur MySQL sur une autre VM que celle qui héberge déjà le serveur Apache. Le diagramme de déploiement du système est donné à la figure 10.

4.1 Motivation

Séparer le serveur web du serveur de bases de données offre de nombreux avantages en terme de sécurité et de fiabilité.

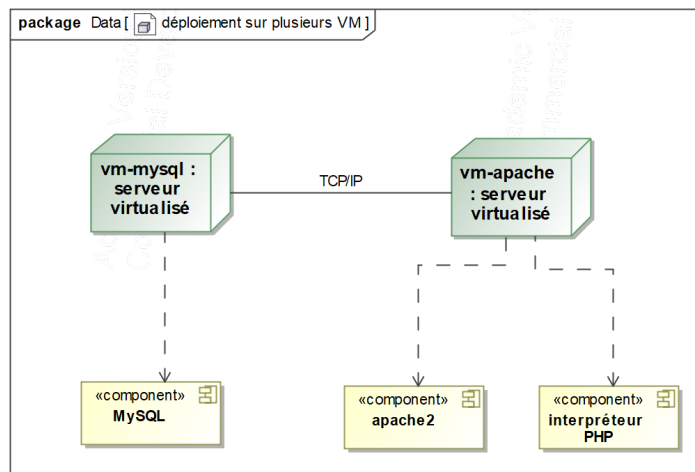


FIGURE 10 – Déploiement du site Wordpress sur deux VM

En entreprise on évite d'exposer inutilement les services à l'Internet. Les services auxquels les utilisateurs ont accès sont sur un réseau accessible publiquement, les services annexes sont sur un autre réseau qui ne communique pas avec le premier. Ainsi, si un service est piraté, les autres services ne le sont pas nécessairement.

C'est ce que nous essayons de reproduire ici de manière extrêmement simplifiée. Un vrai réseau d'entreprise utiliserait des notions de Zone Démilitarisées (DMZ), de proxy inversé, de pare-feu, etc. Rien de tout cela ici ; la sécurité informatique est un sujet à part entière et ce n'est pas celui de ce TD.

Nous nous contenterons donc de de créer deux réseaux : un réseau public sur lequel les utilisateurs ont accès au site Wordpress et un réseau privé pour faire communiquer Apache et MySQL.

4.2 Étapes

Suivez ces étapes pour configurer Wordpress comme sur le diagramme de déploiement de la figure 10.

Remarque : Les instructions sur les noms de machines données à la section 3 s'applique ici aussi. Si votre première VM s'appelle "r408-web-yz" alors la deuxième devra s'appeler "r408-sql-yz". Dans le texte du document on ne dira que "r408-web" ou "r408-sql".

4.2.1 Modifications à apporter à r408-apache

Le protocole Avahi, appelé aussi mDNS ou Bonjour, permet aux machines d'un même réseau de communiquer avec leur nom sans passer par un serveur DNS. Installez Avahi sur "r408-web" avec la commande :

```
sudo apt install avahi-daemon.
```

Éteignez ensuite la machine, puis ajoutez-lui une deuxième interface réseau : dans le menu de la VM, choisissez "Hardware" puis "Add" puis "Network Device" (figure 11).

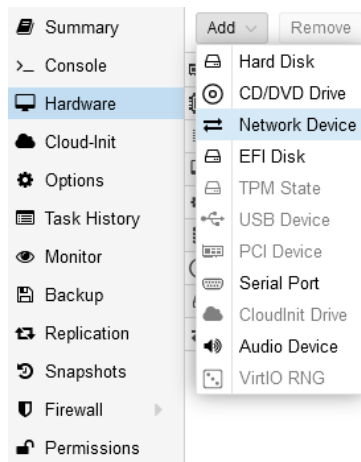


FIGURE 11 – Ajout d'une deuxième carte réseau à une VM

Proxmox vous demande les paramètres de la nouvelle interface. Gardez les paramètres par défaut sauf pour le pont : chaque binôme doit choisir un pont, *autre que vmbr0*, et attribuer ce pont à la nouvelle interface réseau. Voir figure 12.

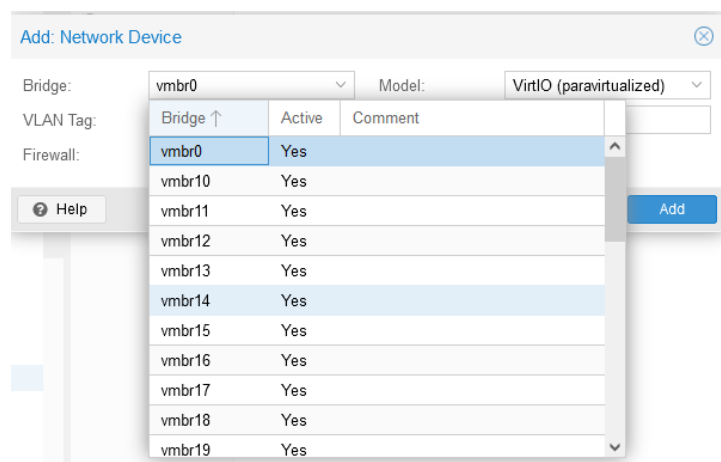


FIGURE 12 – Choix du réseau virtuel. Ne pas prendre vmbr0 !

Vous pouvez redémarrer la VM et vérifier avec la commande `ip a` que la nouvelle interface réseau est bien prise reconnue par le système. Nous la configurerons plus tard.

Faites migrer la VM vers le même serveur que le patron que vous avez créé à l'étape 2.

4.2.2 Instanciation et paramétrage du serveur r408-sql

Instanciez une nouvelle VM à partir du patron de l'étape 2. Nommez-la "r408-sql" (+ signature).

Avant de démarrer la VM, ajoutez-lui une deuxième carte réseau, comme vous l'avez fait pour "r408-web" et assignez-la au même pont (*bridge*) que la deuxième carte réseau de "r408-web".

Démarrez la machine et modifiez les fichiers `/etc/hosts` et `/etc/hostname` comme vous l'avez fait pour "r408-web". Redémarrez la machine.

Vérifiez que les deux VM ont des adresses IP différentes sur la première interface, que la deuxième interface n'est pas activée (`ip a` affiche "state DOWN") et que chaque VM répond aux `ping` envoyés par l'autre VM.

4.2.3 Installation et configuration de MySQL sur vw-sql

Installez le serveur MySQL et Avahi sur la nouvelle VM :

```
sudo apt install mysql-server avahi-daemon
```

Les deux machines peuvent désormais se contacter avec leur nom d'hôte, essayez `ping r408-sql.local` et `ping r408-web.local`. Si ça ne fonctionne pas c'est peut-être parce que l'extension ".local" est déjà utilisée sur le réseau ; dans ce cas vous utiliserez les adresse IP dans la suite du TD au lieu des noms complets.

Depuis "r408-sql", connectez vous au serveur (avec le compte root) et créez l'utilisateur "wordpress" comme vous l'avez fait sur "r408-web". Testez la connexion depuis "r408-sql" :

```
mysql -u wordpress -p
```

Saisissez le mot de passe. La connexion doit fonctionner.

Testez maintenant la connexion à la base de données depuis "wm-web" :

```
mysql -h r408-sql.local -u wordpress -p
```

Saisissez le mot de passe. Ça ne marche pas. MySQL n'accepte par défaut que des connexions depuis la machine locale, donc "r408-sql". Nous allons corriger ça.

Sur "r408-sql", en tant que root, éditez le fichier `/etc/mysql/mysql.conf/d/mysqld.cnf` et modifiez la ligne :

```
bind-address = 127.0.0.1
```

en

```
bind-address = 0.0.0.0
```

Redémarrez le serveur MySQL avec `sudo systemctl restart mysql` et testez à nouveau la connexion depuis "r408-web", ça doit maintenant marcher.

Vous allez maintenant migrer les données depuis "r408-web" vers "r408-sql".

4.2.4 Extraction des données SQL du serveur r408-web

La commande `mysqldump` extrait la structure d'une base MySQL et ses données et produit un fichier de commandes SQL pour les recréer.

Sur "r408-web", exécutez la commande :

```
mysqldump -u wordpress -p --databases wordpress > ~/wordpress.sql
```

Saisissez le mot de passe. La commande crée un fichier `wordpress.sql` dans le répertoire de votre utilisateur. Vérifiez que le contenu du fichier contient des commandes SQL et que sa taille fait quelques dizaines de ko.

4.2.5 Création des données SQL sur le nouveau serveur

Pour créer la base et toutes ses données sur "r408-sql" vous allez vous y connecter depuis "r408-web" car c'est cette dernière qui contient le fichier avec la structure et les données.

Depuis "r408-web" exécutez la commande :

```
mysql -u wordpress -p -h r408-sql.local < wordpress.sql
```

Remplacez le nom de la machine par son adresse IP si nécessaire. Saisissez le mot de passe, si la commande n'affiche pas d'erreur c'est qu'elle a réussi.

Vous pouvez tout de même vérifier que les tables sont présentes sur la nouvelle base en vous y connectant en tant que root ou wordpress, et en appelant par exemple la requête :

```
SHOW TABLES FROM wordpress;
```

Si cette commande affiche une liste d'une douzaine de noms de tables commençant par wp-* c'est bon.

Supprimez la base "wordpress" de "r408-web" en y exécutant la requête SQL :

```
DROP DATABASE wordpress;
```

4.3 Configuration de Wordpress pour deux serveurs

À l'étape 3.1.5 vous avez édité le fichier wp-config.php de "r408-web". Ouvrez à nouveau ce fichier avec un éditeur, en tant que root, et modifiez la ligne

```
define( 'DB_HOST', 'localhost' );
```

par

```
define( 'DB_HOST', 'r408-sql.local' );
```

N'oubliez pas la signature dans le nom r408-sql.local. Si avahi ne fonctionne pas, remplacez le nom de machine par l'adresse IP correspondante.

Sauvegardez le fichier. Il n'est pas nécessaire de redémarrer Apache

Ouvrez la page de Wordpress sur le navigateur, il doit fonctionner et l'article de test que vous avez créé à l'étape 3.1.7 doit être présent. Vous pouvez vous y connecter à l'adresse <http://r408-web.local/wp-admin/> et accéder au *dashboard*.

4.3.1 Création du réseau privé

Vous allez maintenant créer un réseau privé et y connecter la deuxième interface réseau de "r408-web" et "r408-sql". Par la suite nous nommerons cette interface ens19. Si la commande ip a vous indique un autre nom que ens19 vous devrez utiliser cet autre nom.

Il n'y a pas de service DHCP sur ens19, il faut attribuer des adresses manuellement. Voyez avec votre enseignant chargé de TD quelles sont les adresses que vous pouvez utiliser. Elles doivent être différentes pour chaque binôme. Dans les instructions qui suivent nous attribuons 10.10.10.27 à "r408-web" et 10.10.10.28 à "r408-sql". Vous adapterez les instructions aux adresses qui vous ont été attribuées.

En tant que root sur "r408-web" modifiez le fichier /etc/netplan/000-installer-config.yaml, la nouvelle version doit ressembler à ceci :

```
networks:
  ethernets:
    ens18:
      dhcp4: true
    ens19:
      addresses: [10.10.10.27/24]
  version: 2
```

Faites très attention à l'indentation. Remplacez ens18 et ens19 par les noms d'interface sur votre VM, remplacez l'adresse 10.10.10.27 par l'adresse IP choisie.

Attention : Si vous êtes connecté en SSH, le redémarrage du service réseau va vous déconnecter. Vous pouvez bien sûr attendre quelques secondes et vous reconnecter, mais si vous avez fait une erreur de connexion le serveur ne sera plus connecté au réseau du tout, et SSH ne fonctionnera plus. Dans ce cas utilisez la console de Proxmox pour remettre en route la connexion réseau.

Redémarrez le service réseau avec :

```
sudo netplan apply && sudo systemctl restart systemd-networkd
```

Reconnectez-vous par SSH si nécessaire et vérifiez avec `ip a` que l'adresse a bien été attribuée.

Recommencez toute l'opération sur "r408-sql" en adaptant l'adresse IP.

Vérifiez que les VM répondent au ping de l'une vers l'autre et inversement, avec les nouvelles adresses.

4.3.2 Configuration de Wordpress et de MySQL, essais

Sur "r408-web", modifiez wp-config pour utiliser la base de données à l'adresse privée de "r408-sql" ("10.10.10.28" avec les adresses données en exemple, à adapter à vos adresses)

Sur "r408-sql" modifiez le fichier mysql.cnf et remplacez la ligne :

```
bind-address = 0.0.0.0
```

par (en adaptant l'adresse) : `bind-address = 10.10.10.28`

Remplacez "10.10.10.28" par l'adresse IP de "r408-sql". Le serveur MySQL n'accepte désormais que des connexions sur son adresse privée.

Redémarrez MySQL avec `sudo systemctl restart mysql`. Testez ensuite le site Wordpress, il doit fonctionner normalement.

Les plus aventureux d'entre vous peuvent maintenant essayer de déconnecter "r408-sql" du premier réseau et vérifier que le site fonctionne toujours.

5 Conclusion

Dans ce TD nous avons créé un patron pour instancier rapidement des machines virtuelles, nous avons utilisé une instance pour y installer un service et nous avons testé la migration de cette machine vers un nouvel hyperviseur et vérifié que le service fonctionnait toujours. Nous avons ensuite créé une deuxième instance à partir du patron, sur laquelle nous avons migré la base de données. Nous avons fait fonctionner le service précédemment installé sur les deux VM connectées au réseau local. Enfin, nous avons créé un réseau privé dédié aux communications entre le serveur web et la base de données.

Ce document est mis à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International.