

CRÉER UN SERVEUR D'HÉBERGEMENT MUTUALISÉ

Le TP de John expliqué pas à pas

Objectif du support

Comprendre ce que nous avons construit, pourquoi chaque service existe et comment les permissions, Apache, FTP, MariaDB et WordPress travaillent ensemble.

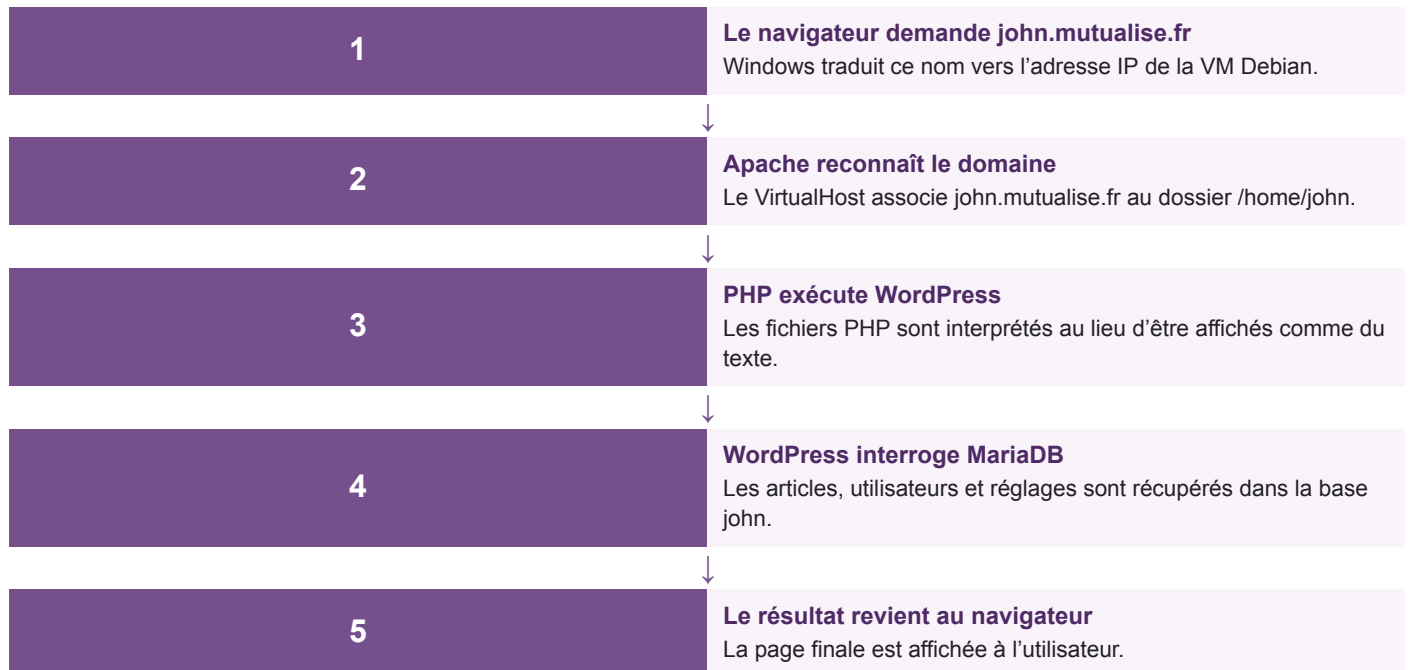
Service	Rôle
Apache2	Affiche le site web
PHP	Exécute WordPress
MariaDB	Stocke les données
phpMyAdmin	Administre MariaDB dans le navigateur
ProFTPD	Reçoit les fichiers envoyés en FTP
WordPress	Permet de créer et gérer le site

Dossier étudié : /home/john • Domaine : john.mutualise.fr

Sommaire

- 1. Vue d'ensemble du serveur
- 2. Hébergement mutualisé et environnement multi-utilisateurs
- 3. Les comptes utilisés dans le TP
- 4. Le dossier /home/john et les permissions Linux
- 5. Apache2 et le VirtualHost
- 6. Le fichier hosts, curl et les services
- 7. FTP avec ProFTPD et FileZilla
- 8. MariaDB et phpMyAdmin
- 9. PHP et installation de WordPress
- 10. Sauvegardes et vérifications
- 11. Méthode de mémorisation et mini-quiz

Le fil rouge du TP



1. Vue d'ensemble du serveur

Nous avons transformé une machine Debian en petit serveur d'hébergement web. Plusieurs services coopèrent : aucun ne remplace les autres.

Composant	Question à laquelle il répond	Ce qu'il fait
Apache2	Qui doit recevoir la requête web ?	Écoute le port 80 et sert le bon site.
PHP	Comment exécuter WordPress ?	Interprète le code PHP.
MariaDB	Où stocker les données ?	Conserve articles, comptes et réglages.
phpMyAdmin	Comment gérer la base graphiquement ?	Fournit une interface web.
ProFTPD	Comment déposer les fichiers ?	Permet les transferts FTP.
WordPress	Comment administrer le contenu ?	Fournit le CMS et /wp-admin.

À retenir

Apache affiche, PHP exécute, MariaDB mémorise, ProFTPD transfère et WordPress organise le contenu.

Les paramètres de John

Élément	Valeur
IP de la VM	192.168.190.130
Nom du site	john.mutualise.fr
Compte Linux / FTP	john
Dossier personnel et web	/home/john
Base de données	john
Compte WordPress	john_admin (exemple)

2. Hébergement mutualisé et Linux multi-utilisateurs

Un serveur mutualisé héberge plusieurs clients sur une même machine. Les services sont partagés, mais chaque client possède son espace.

Client	Utilisateur	Dossier	Domaine	Base
John	john	/home/john	john.mutualise.fr	john
Alice (exemple)	alice	/home/alice	alice.mutualise.fr	alice

Principe de mutualisation

Une seule machine, plusieurs clients, des espaces séparés et des services communs.

Pourquoi Linux convient bien ?

- Chaque processus est lancé sous l'identité d'un utilisateur et d'un groupe.
- Chaque fichier appartient à un utilisateur propriétaire et à un groupe propriétaire.
- Les permissions permettent de décider qui peut lire, écrire ou traverser un dossier.
- Les comptes système, comme www-data, exécutent les services sans utiliser root.

root : le compte administrateur

- root possède l'UID 0 et les droits les plus élevés.
- Il sert aux tâches administratives : installation, configuration, permissions, services.
- Dans le TP, nous passons en root avec :

su -

Attention

root n'est pas un compte de travail ordinaire. Une erreur lancée en root peut modifier ou supprimer tout le système.

3. Les comptes utilisés dans le TP

Le mot « john » apparaît plusieurs fois, mais il ne désigne pas toujours le même compte. C'est une source classique de confusion.

Compte	Système	Utilité	Exemple d'accès
john	Linux	Propriétaire des fichiers et compte FTP	Connexion FTP
www-data	Linux système	Identité utilisée par Apache	Lecture du site
john@localhost	MariaDB	Connexion de WordPress à la base	Base john
john_admin	WordPress	Administration du CMS	/wp-admin

Création du compte Linux

```
adduser --home /home/john john
```

Cette commande crée le compte john et fixe son dossier personnel à /home/john.

Vérifier le compte

```
getent passwd john
```

Lecture simple

Le compte Linux donne une identité au client. Le compte MariaDB donne une identité à WordPress. Le compte WordPress donne une identité à l'administrateur du site.

4. Le dossier /home/john et les permissions Linux

/home/john joue trois rôles à la fois : dossier personnel de John, racine FTP et racine du site web.

Chemin	Rôle
/home/john	Racine du client
/home/john/index.php	Point d'entrée de WordPress
/home/john/wp-admin	Administration WordPress
/home/john/wp-content	Thèmes, extensions, médias
/home/john/wp-config.php	Connexion à MariaDB et réglages sensibles

r, w et x

Droit	Fichier	Dossier
r — read	Lire le contenu	Lister le contenu
w — write	Modifier le fichier	Créer, supprimer ou renommer des éléments
x — execute	Exécuter le fichier	Entrer ou traverser le dossier

Les trois catégories

Lettre	Catégorie	Dans notre TP
u	Utilisateur propriétaire	john
g	Groupe propriétaire	www-data
o	Autres utilisateurs	Aucun accès prévu

Changer le propriétaire

```
chown -R john:www-data /home/john
```

Partie	Signification
chown	Change le propriétaire
-R	Applique récursivement
john	Utilisateur propriétaire
www-data	Groupe propriétaire
/home/john	Dossier concerné

Permissions numériques

Valeur	Droits
0	---
1	--X
2	-W-
3	-WX
4	r--

Valeur	Droits
5	r-x
6	rw-
7	rwX

Méthode

On additionne : r = 4, w = 2, x = 1. Exemple : 6 = 4 + 2 = rw-.

Le 2 devant 750 : le SGID

```
chmod 2750 /home/john
```

Chiffre	Signification
2	Active le SGID sur le dossier
7	john : rwx
5	www-data : r-x
0	autres : ---

Sur un dossier, le SGID aide les nouveaux éléments à conserver le groupe du dossier parent, ici www-data.

Permissions finales du TP

```
chown -R john:www-data /home/john
find /home/john -type d -exec chmod 2750 {} \;
find /home/john -type f -exec chmod 0640 {} \;
find /home/john/wp-content -type d -exec chmod 2770 {} \;
find /home/john/wp-content -type f -exec chmod 0660 {} \;
chmod 0640 /home/john/wp-config.php
```

Zone	Pourquoi ces droits ?
Dossiers du site	John gère, Apache lit et traverse.
Fichiers du site	John lit/écrit, Apache lit.
wp-content	Apache doit aussi écrire pour les médias, thèmes et extensions.
wp-config.php	Apache doit lire, mais les autres utilisateurs ne doivent pas y accéder.

Piège classique

Ne pas donner 777 partout. Cela masque les problèmes de droits et ouvre inutilement l'accès en écriture.

5. Apache2 et le VirtualHost

Apache reçoit les requêtes HTTP. Le VirtualHost lui indique quel dossier utiliser lorsqu'un domaine précis est demandé.

`/etc/apache2/sites-available/john.mutualise.fr.conf`

```
<VirtualHost *:80>
    ServerName john.mutualise.fr
    DocumentRoot /home/john

    <Directory /home/john>
        Options FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/john_error.log
    CustomLog ${APACHE_LOG_DIR}/john_access.log combined
</VirtualHost>
```

Directive	Rôle
<code><VirtualHost *:80></code>	Écoute le HTTP sur toutes les interfaces.
<code>ServerName</code>	Nom qui déclenche ce site.
<code>DocumentRoot</code>	Dossier contenant les fichiers.
<code><Directory></code>	Règles Apache appliquées au dossier.
<code>AllowOverride All</code>	Autorise le fichier .htaccess de WordPress.
<code>Require all granted</code>	Autorise l'accès au site.
<code>ErrorLog / CustomLog</code>	Fichiers utiles pour le dépannage.

Activer et tester Apache

```
a2ensite john.mutualise.fr.conf
a2enmod rewrite
apache2ctl configtest
systemctl restart apache2
systemctl status apache2 --no-pager
apache2ctl -S
```

Résultat indispensable

apache2ctl configtest doit afficher Syntax OK avant le redémarrage.

Erreur fréquente

Modifier un fichier Apache sans redémarrer ou recharger le service : la nouvelle configuration n'est alors pas prise en compte.

6. Fichier hosts, curl et gestion des services

Pourquoi modifier le fichier hosts ?

Le domaine john.mutualise.fr n'existe pas réellement dans le DNS public. Windows doit donc apprendre localement quelle IP utiliser.

```
C:\Windows\System32\drivers\etc\hosts
```

```
192.168.190.130 john.mutualise.fr
```

PowerShell Windows

```
ipconfig /flushdns
ping john.mutualise.fr
```

Interprétation

Le fichier hosts remplace temporairement un véritable serveur DNS pour ce poste Windows.

Tester le VirtualHost avec curl

```
curl -H "Host: john.mutualise.fr" http://127.0.0.1
```

Élément	Signification
curl	Effectue une requête web.
-H	Ajoute un en-tête HTTP.
Host: john.mutualise.fr	Indique quel VirtualHost est demandé.
127.0.0.1	Le serveur se teste lui-même.

Comprendre systemctl

Commande	Action
systemctl start apache2	Démarre Apache.
systemctl stop apache2	Arrête Apache.
systemctl restart apache2	Redémarre et relit la configuration.
systemctl reload apache2	Recharge la configuration sans arrêt complet.
systemctl status apache2	Affiche l'état du service.
systemctl enable --now apache2	Active au démarrage et lance immédiatement.

7. FTP avec ProFTPD et FileZilla

ProFTPD est le serveur FTP installé sur Debian. FileZilla Client est l'outil graphique utilisé sous Windows.

Limiter John à son dossier

```
/etc/proftpd/proftpd.conf
```

```
DefaultRoot ~
```

Le symbole ~ désigne le dossier personnel de l'utilisateur connecté. Pour John : ~ = /home/john.

Conséquence

Dans FileZilla, John voit « / », mais ce « / » correspond réellement à /home/john. Il ne peut pas parcourir tout le serveur.

Paramètres FileZilla

Champ	Valeur
Protocole	FTP
Hôte	john.mutualise.fr
Port	21
Utilisateur	john
Mot de passe	johnpwd

Dans FileZilla : la partie gauche représente Windows ; la partie droite représente Debian.

Sécurité

Le FTP simple n'est pas chiffré. Il est acceptable pour ce TP local, mais pas pour un serveur exposé à Internet.

Test en ligne de commande

```
ftp 192.168.190.130
pwd
ls
binary
put test-john.txt
bye
```

8. MariaDB et phpMyAdmin

WordPress ne stocke pas les articles et les comptes dans de simples fichiers : il utilise une base de données.

Créer la base et l'utilisateur

```
CREATE DATABASE john
CHARACTER SET utf8mb4
COLLATE utf8mb4_unicode_ci;

CREATE USER 'john'@'localhost'
IDENTIFIED BY 'johnpwd';

GRANT ALL PRIVILEGES
ON john.*
TO 'john'@'localhost';

SHOW GRANTS FOR 'john'@'localhost';
```

Élément	Explication
utf8mb4	Gère correctement les caractères Unicode et les emojis.
'john'@'localhost'	Compte autorisé depuis la machine locale.
john.*	Toutes les tables de la base john uniquement.
SHOW GRANTS	Affiche les droits du compte.

Principe de moindre privilège

Le compte MariaDB de John reçoit les droits sur sa base, pas sur toutes les bases du serveur.

phpMyAdmin

phpMyAdmin est une interface web pour administrer MariaDB. Ce n'est ni une base de données ni un service séparé.

Élément	Rôle
MariaDB	Moteur de base de données.
phpMyAdmin	Interface graphique pour gérer MariaDB.

<http://john.mutualise.fr/phpmyadmin>

9. PHP et installation de WordPress

Extensions PHP installées

```
apt install php-mysql php-curl php-gd php-mbstring php-xml php-zip
```

Extension	Utilité principale
php-mysql	Connexion entre PHP et MariaDB.
php-curl	Communications HTTP.
php-gd	Traitement des images.
php-mbstring	Gestion avancée des caractères.
php-xml	Traitement XML.
php-zip	Gestion des archives ZIP.

Après installation des extensions

```
systemctl restart apache2
```

Envoyer WordPress au bon endroit

Correct	Incorrect
/home/john/index.php	/home/john/wordpress/index.php
/home/john/wp-admin	/home/john/wordpress/wp-admin
/home/john/wp-content	/home/john/wordpress/wp-content

À retenir

Il fallait envoyer le contenu du dossier WordPress, et non le dossier WordPress lui-même.

Connexion de WordPress à MariaDB

Champ WordPress	Valeur
Nom de la base	john
Identifiant	john
Mot de passe	johnpwd
Adresse de la base	localhost
Préfixe	wp_

WordPress crée ensuite des tables telles que wp_users, wp_posts, wp_options et wp_comments.

Pourquoi wp-content est particulier ?

- Il contient les thèmes et les extensions.
- Il reçoit les médias envoyés dans la bibliothèque WordPress.
- Apache doit donc pouvoir y écrire, contrairement au reste du site qui peut être plus strict.

10. Sauvegardes et vérifications finales

Une sauvegarde WordPress complète comprend toujours deux éléments : les fichiers et la base de données.

Sauvegarder les fichiers

```
tar -czf /root/john-wordpress-$(date +%F).tar.gz -C /home john
```

Option	Signification
-c	Créer une archive.
-z	Compresser avec gzip.
-f	Indiquer le fichier de destination.
\$(date +%F)	Ajouter la date au nom.
-C /home	Se placer dans /home avant d'archiver john.

Sauvegarder la base

```
mariadb-dump -u john -p john > /root/john-bdd-$(date +%F).sql
```

Partie	Signification
mariadb-dump	Exporte la base.
-u john	Utilise le compte john.
-p	Demande le mot de passe.
john	Nom de la base.
> fichier.sql	Écrit le résultat dans un fichier.

Checklist finale

- ☐ Apache2, MariaDB et ProFTPD sont actifs.
- ☐ getent passwd john affiche /home/john.
- ☐ Le VirtualHost pointe vers /home/john.
- ☐ apache2ctl configtest affiche Syntax OK.
- ☐ john.mutualise.fr répond depuis Windows.
- ☐ FileZilla envoie les fichiers dans /home/john.
- ☐ phpMyAdmin affiche la base john.
- ☐ WordPress et /wp-admin fonctionnent.
- ☐ Les permissions finales sont appliquées.
- ☐ Les fichiers et la base sont sauvegardés.

11. Dépannage, mémorisation et mini-quiz

Ordre de dépannage

1	Vérifier le service systemctl status apache2 / mariadb / proftpd
2	Vérifier la configuration apache2ctl configtest et apache2ctl -S
3	Vérifier le chemin DocumentRoot et présence des fichiers dans /home/john
4	Vérifier les droits ls -ld, ls -l, chown et chmod
5	Tester localement curl avec l'en-tête Host
6	Tester depuis Windows fichier hosts, ping, navigateur, FileZilla
7	Lire les journaux john_error.log ou les messages de systemctl status

Familles de commandes à retenir

Commande	Idée à retenir
apt	Installer des logiciels.
systemctl	Gérer les services.
adduser	Créer un utilisateur.
chown	Changer propriétaire et groupe.
chmod	Changer les permissions.
ls -l	Voir les droits et propriétaires.
nano	Modifier un fichier.
a2ensite	Activer un site Apache.
a2enmod	Activer un module Apache.
apache2ctl	Vérifier Apache.
mariadb	Ouvrir la console SQL.
curl	Tester une page web.
find	Appliquer une action à plusieurs éléments.
tar	Archiver les fichiers.
mariadb-dump	Sauvegarder une base.

Bonne stratégie

Ne cherche pas à retenir toutes les lignes immédiatement. Retiens d'abord la logique : utilisateur → dossier → permissions → FTP → Apache → base → WordPress → sauvegardes.

Mini-quiz**Question : Pourquoi Apache a-t-il besoin du groupe www-data ?**

Réponse : Parce que le processus Apache utilise ce compte système pour lire et parfois écrire dans les fichiers du site.

Question : Pourquoi /home/john possède-t-il le SGID ?

Réponse : Pour aider les nouveaux fichiers et dossiers à conserver le groupe www-data.

Question : Quelle différence entre MariaDB et phpMyAdmin ?

Réponse : MariaDB stocke les données ; phpMyAdmin est une interface web pour les administrer.

Question : Pourquoi modifier le fichier hosts Windows ?

Réponse : Pour associer localement john.mutualise.fr à l'adresse IP de la VM Debian.

Question : Pourquoi sauvegarder à la fois les fichiers et la base ?

Réponse : Parce qu'un site WordPress dépend des deux : les fichiers seuls ne contiennent pas les articles, comptes et réglages de la base.

Résumé en une phrase**Le TP de John**

Nous avons créé un utilisateur isolé, donné à Apache les droits nécessaires, relié un domaine à /home/john, installé un accès FTP, une base MariaDB et WordPress, puis sécurisé et sauvegardé l'ensemble.