

Faire communiquer 3 Vms (Symfony, MariaDB, Windows Server)

Pré-requis :

VM Symfony :

- Avoir un serveur Debian avec une IP fixe
- SSH installé et configuré

Ce qui est nécessaire pour la VM Symfony :

- Apache2
- PHP
- Composer
- Symfony
- mariadb-client
- Git
- UFW (pour gérer les ports)

Tout d'abord :

Apache2

Mettre à jour la liste des paquets avec la commande : apt update

```
root@gsb-symfony:~# apt update
Atteint : 1 http://deb.debian.org/debian trixie InRelease
Atteint : 2 http://security.debian.org/debian-security trixie-security InRelease
Atteint : 3 http://deb.debian.org/debian trixie-updates InRelease
Tous les paquets sont à jour.
```

Installer apache2 : apt install apache2

```
root@gsb-symfony:~# apt install apache2
Installation de :
  apache2

Installation de dépendances :
  apache2-bin          libaprutil1t64      librtmp1
  apache2-data         libcurl4t64         libsasl2-2
  apache2-utils       libldap-common      libsasl2-modules
  libapr1t64           libldap2            libsasl2-modules-db
  libaprutil1-dbd-sqlite3 liblua5.4-0         libssh2-1t64
  libaprutil1-ldap     libnghttp3-9        ssl-cert

Paquets suggérés :
  apache2-doc          libsasl2-modules-gssapi-mit
  apache2-suexec-pristine | libsasl2-modules-gssapi-heimdal
  | apache2-suexec-custom libsasl2-modules-ldap
  ufw                  libsasl2-modules-otp
  www-browser         libsasl2-modules-sql

Sommaire :
  Mise à niveau de : 0. Installation de : 19 Supprimé : 0. Non mis à jour : 0
  Taille du téléchargement : 3 327 kB
  Espace nécessaire : 17,9 MB / 16,4 GB disponible
```

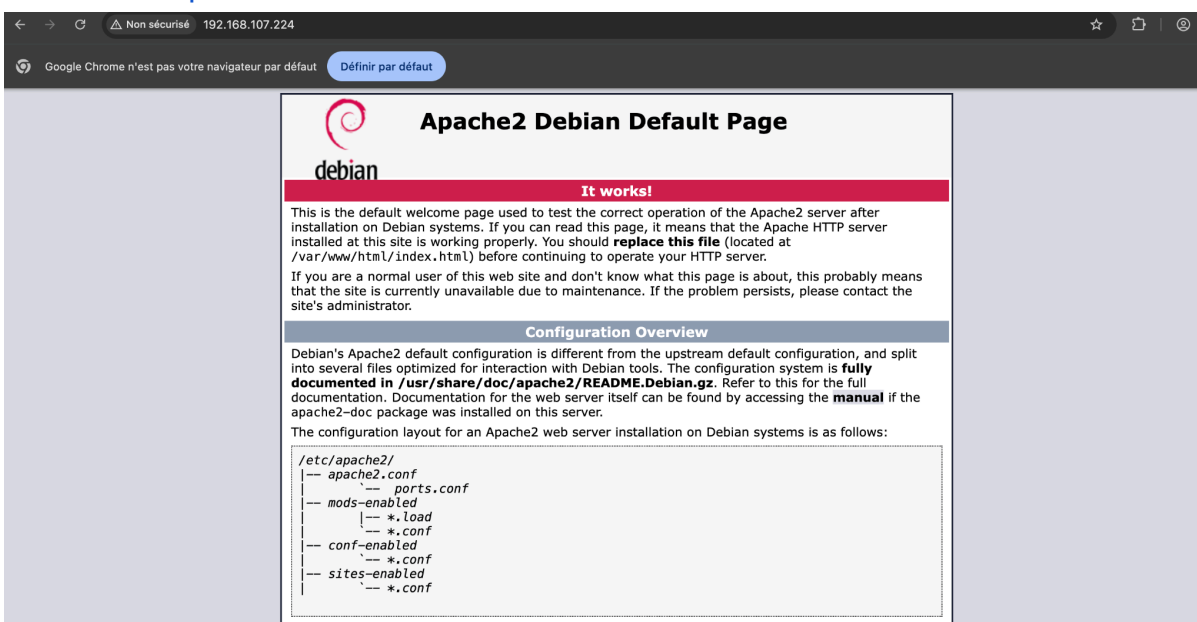
```
[root@gsb-symfony:~# systemctl status apache2
● apache2.service – The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset:
   Active: active (running) since Mon 2026-03-09 14:08:26 CET; 1min 24s ago
   Invocation: 9ae87083d59141f8be75511fd97966bb
   Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1916 (apache2)
   Tasks: 55 (limit: 4783)
   Memory: 5.6M (peak: 6M)
   CPU: 27ms
   CGroup: /system.slice/apache2.service
           └─1916 /usr/sbin/apache2 -k start
             └─1917 /usr/sbin/apache2 -k start
               └─1918 /usr/sbin/apache2 -k start

mars 09 14:08:26 gsb-symfony systemd[1]: Starting apache2.service – The Apache
mars 09 14:08:26 gsb-symfony apachectl[1915]: AH00558: apache2: Could not relia
mars 09 14:08:26 gsb-symfony systemd[1]: Started apache2.service – The Apache
lines 1-17/17 (END)
```

Démarrer Apache2 au démarrage de la machine : systemctl enable apache2

```
root@gsb-symfony:~# systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/sy
stemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
root@gsb-symfony:~#
```

vérifier que le service soit bien fonctionnel dans un navigateur web avec l'adresse IP attribuée. "<http://adresseIP>" :



PHP

Pour installer php avec toutes ses dépendances : `apt install php php-{cli,common,json,opcache,pdo,mysql,curl,gd,zip,intl,xml,xsl,bz2,ldap,dev,apcu,m bstring}`

```
autoconf-archive      gdb                   libheif-plugin-j2kenc
gnu-standards         gcc-doc              libheif-plugin-kvazaar
autoconf-doc         gdb-aarch64-linux-gnu libheif-plugin-rav1e
binutils-doc         gettext-doc          libheif-plugin-svtenc
gprofng-gui          libasprintf-dev      libtool-doc
binutils-gold         libgettextpo-dev     libssl-doc
cpp-doc              gnulib-110n          libstdc++-14-doc
gcc-14-locales       libc-devtools        gfortran
cpp-14-doc           glibc-doc            | fortran95-compiler
debtags              git                  gcj-jdk
dh-make              bzip2                 m4-doc
debian-keyring       rng-tools             make-doc
debian-tag2upload-keyring libgd-tools           ed
gcc-14-doc           libheif-plugin-ffmpegdec diffutils-doc
gcc-multilib         libheif-plugin-jpegdec dh-php
flex                 libheif-plugin-jpegenc libmail-box-perl
bison               libheif-plugin-j2kdec
```

Sommaire :
Mise à niveau de : 0. Installation de : 158 Supprimé : 0. Non mis à jour : 0
Taille du téléchargement : 113 MB
Espace nécessaire : 460 MB / 16,4 GB disponible

Continuer ? [O/n] █

Continuer avec "O"

```
root@gsb-symfony:~# php -v
PHP 8.4.16 (cli) (built: Dec 18 2025 21:19:25) (NTS)
Copyright (c) The PHP Group
Built by Debian
Zend Engine v4.4.16, Copyright (c) Zend Technologies
    with Zend OPcache v8.4.16, Copyright (c), by Zend Technologies
root@gsb-symfony:~# █
```

Mariadb-client

Pour installer le client mariadb : apt install mariadb-client

```
root@gsb-symfony:~# apt install mariadb-client
Installation de :
  mariadb-client

Installation de dépendances :
  libconfig-inifiles-perl  libgpm2      libterm-readkey-perl  mariadb-common
  libdbd-mariadb-perl     libmariadb3  liburing2             mysql-common
  libdbi-perl             libncurses6  mariadb-client-core

Paquets suggérés :
  libclone-perl  libnet-daemon-perl  gpm
  libmldbm-perl  libsql-statement-perl

Sommaire :
  Mise à niveau de : 0. Installation de : 12Supprimé : 0. Non mis à jour : 0
  Taille du téléchargement : 5 303 kB
  Espace nécessaire : 86,7 MB / 15,9 GB disponible

Continuer ? [O/n]
```

Continuer avec "O"

Pour vérifier l'installation de mariadb : mariadb --version

```
root@gsb-symfony:~# mariadb --version
mariadb from 11.8.3-MariaDB, client 15.2 for debian-linux-gnu (aarch64) using E
ditLine wrapper
root@gsb-symfony:~#
```

Composer

Pour installer Composer : wget https://getcomposer.org/installer -O composer-setup.php

```
root@gsb-symfony:~# wget https://getcomposer.org/installer -O composer-setup.php

--2026-03-09 14:20:58-- https://getcomposer.org/installer
Résolution de getcomposer.org (getcomposer.org)... 57.128.19.244, 2001:41d0:304:30
0::18ef
Connexion à getcomposer.org (getcomposer.org)|57.128.19.244|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 59524 (58K) [application/octet-stream]
Sauvegarde en : « composer-setup.php »

composer-setup.php 100%[=====>] 58,13K --.-KB/s ds 0,02s

2026-03-09 14:20:59 (3,29 MB/s) - « composer-setup.php » sauvegardé [59524/59524
]

root@gsb-symfony:~#
```

Puis php composer-setup.php --install-dir=/usr/local/bin --filename=composer

```

root@gsb-symfony:~# php composer-setup.php --install-dir=/usr/local/bin --filename=composer
All settings correct for using Composer
Downloading...

Composer (version 2.9.5) successfully installed to: /usr/local/bin/composer
Use it: php /usr/local/bin/composer

root@gsb-symfony:~# █

```

Symfony

Pour télécharger Symfony : `wget https://get.symfony.com/cli/installer -O - | bash`

```

root@gsb-symfony:~# wget https://get.symfony.com/cli/installer -O - | bash
--2026-03-09 14:22:53-- https://get.symfony.com/cli/installer
Résolution de get.symfony.com (get.symfony.com)... 18.164.52.7, 18.164.52.20, 18.164.52.17, ...
Connexion à get.symfony.com (get.symfony.com)|18.164.52.7|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 6100 (6,0K) [binary/octet-stream]
Sauvegarde en : « STDOUT »

-          100%[=====>] 5,96K --.-KB/s  ds 0s

2026-03-09 14:22:54 (556 MB/s) – envoi vers sortie standard [6100/6100]

Symfony CLI installer

Environment check
[*] wget is installed
[*] Tar is installed
[ ] Warning: Git will be needed.
[*] Your architecture (arm64) is supported

Download
Downloading https://github.com/symfony-cli/symfony-cli/releases/latest/download/symfony-cli_linux_arm64.tar.gz...

```

Si installé en root, déplacer le fichier avec la commande “`mv /root/.symfony5/bin/symfony /usr/local/bin/symfony`” sinon `mv /home/NOM_USER/.symfony5/bin/symfony /usr/local/bin/symfony`

```

The Symfony CLI was installed successfully!

Use it as a local file:
/root/.symfony5/bin/symfony

Or add the following line to your shell configuration file:
export PATH="$HOME/.symfony5/bin:$PATH"

Or install it globally on your system:
mv /root/.symfony5/bin/symfony /usr/local/bin/symfony

Then start a new shell and run 'symfony'
root@gsb-symfony:~# mv /root/.symfony5/bin/symfony /usr/local/bin/symfony
root@gsb-symfony:~# █

```

Git

Pour installer git : apt install git -y

```
root@gsb-symfony:~# apt install git -y
Installation de :
  git

Installation de dépendances :
  git-man          liberror-perl    libngtcp2-crypto-gnutls8
  libcurl3t64-gnutls  libngtcp2-16

Paquets suggérés :
  git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn

Sommaire :
  Mise à niveau de : 0. Installation de : 6Supprimé : 0. Non mis à jour : 0
  Taille du téléchargement : 11,4 MB
  Espace nécessaire : 54,6 MB / 15,8 GB disponible

Réception de : 1 http://deb.debian.org/debian trixie/main arm64 libngtcp2-16 arm64 1.11.0-1 [121 kB]
Réception de : 2 http://deb.debian.org/debian trixie/main arm64 libngtcp2-crypto-gnutls8 arm64 1.11.0-1 [28,2 kB]
Réception de : 3 http://deb.debian.org/debian trixie/main arm64 libcurl3t64-gnutls arm64 8.14.1-2+deb13u2 [354 kB]
Réception de : 4 http://deb.debian.org/debian trixie/main arm64 liberror-perl al
```

Configurer Git avec l'identité souhaitée

- Tout d'abord, le nom avec git config --global user.name "Ton Nom"
- Ensuite, l'email git config --global user.email "ton.email@example.com"

```
...
[root@gsb-symfony:~# git config --global user.name "GSB-Symfony"
[root@gsb-symfony:~# git config --global user.email "leape@orange.fr"
root@gsb-symfony:~# █
```

1. Générer une clé SSH (authentifier le Git à distance) avec la commande :
ssh-keygen -t ed25519 -C "ton.email@example.com"

```
[root@gsb-symfony:~# ssh-keygen -t ed25519 -C "leape@orange.fr"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519): █
```

Appuyer sur Entrée 3 fois pour valider le fichier dans lequel sera sauvegardé la clé

```
Generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519):
Enter passphrase for "/root/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_ed25519
Your public key has been saved in /root/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:yN+RJwLafEjaqdilqclXkD8UqiCvEW/peNxia8sKXRQ leape@orange.fr
The key's randomart image is:
+--[ED25519 256]--+
|
|+E.
|O++
|.=O.
|O *O..O .
|= +=..O S + .
|+*.O= . O +
|=+*O + . .
|oBB . .
|+==+
+-----[SHA256]-----+
root@gsb-symfony:~# █
```

Récupérer la clé avec : `cat ~/.ssh/id_ed25519.pub`

Il faut ensuite copier la clé dans son gitlab :

Your SSH keys [🔗](#) 6 Add new key

SSH Keys

Add an SSH key for secure access to GitLab. [Learn more.](#)

Key

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIAPIJora6KRXrn4HXCF81G4lt8pttdgHaekGPwejtzpKM leape@orange.fr
```

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'.

Title

Key titles are publicly visible.

Usage type

Expiration date

Optional but recommended. If set, key becomes invalid on the specified date.

(capture d'écran d'une autre procédure)

UFW

Installer ufw : `apt-get install ufw`

```
root@gsb-symfony:~# apt-get install ufw
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  iptables libip4tc2 libip6tc2 libnetfilter-contrack3 libnfnetlink0
Paquets suggérés :
  firewalld rsyslog
Les NOUVEAUX paquets suivants seront installés :
  iptables libip4tc2 libip6tc2 libnetfilter-contrack3 libnfnetlink0 ufw
0 mis à jour, 6 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 616 kB dans les archives.
Après cette opération, 9 961 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

Il faut ensuite activer ufw : `ufw enable`

```
root@gsb-symfony:~# ufw enable
Firewall is active and enabled on system startup
root@gsb-symfony:~# _
```

Il faut ensuite ouvrir les ports souhaités avec : `ufw allow {port}`
(80, 22, 443, 389)

- Port 80 - HTTP
- Port 443 - HTTPS
- Port 22 - SSH
- Port 389 - LDAP (pour la machine virtuelle windows)

```
root@gsb-symfony:~# ufw allow 80/tcp
Rule added
Rule added (v6)
root@gsb-symfony:~# ufw allow 22/tcp
Rule added
Rule added (v6)
root@gsb-symfony:~# ufw allow 443/tcp
Rule added
Rule added (v6)
root@gsb-symfony:~#
```

```
[root@gsb-symfony:/var/www/html# ufw allow 389/tcp
Rule added
Rule added (v6)
```

VM BDD AVEC :

- Avoir un serveur Debian avec une IP fixe
- SSH installé et configuré

Ce qui est nécessaire pour la VM de Base de données :

- MariaDB Serveur
- SSH
- UFW

Mariadb-server:

Pour installer : `mariadb-server`

```
root@gsb-bdd:~# apt install mariadb-server
Installation de :
  mariadb-server

Installation de dépendances :
galera-4          libgpm2          mariadb-plugin-provider-bzip2
gawk             libhtml-parser-perl  libncurses6     mariadb-plugin-provider-lz4
libcgi-fast-perl libhtml-tagset-perl  libnuma1        mariadb-plugin-provider-lzma
libcgi-pm-perl   libhtml-template-perl  libpcre2-posix3  mariadb-plugin-provider-lzo
libclone-perl    libhttp-date-perl    libsigsegv2     mariadb-plugin-provider-snapp
libconfig-inifiles-perl  libhttp-message-perl  libsnappy1v5    mariadb-server-core
libdbd-mariadb-perl  libio-compress-brotli-perl  libterm-readkey-perl  mysql-common
libdbi-perl       libio-html-perl      libtimedate-perl  psmisc
libencode-locale-perl  liblwp-mediatypes-perl  liburi-perl     pv
libfcgi-bin       liblzo2-2           mariadb-client   rsync
libfcgi-perl      libmariadb3         mariadb-client-core  socat
libfcgi0t64       libmpfr6            mariadb-common

Paquets suggérés :
gawk-doc          gpm              libmime-base32-perl  mariadb-test
libmldbm-perl     libdata-dump-perl  libregexp-ipv6-perl  netcat-openbsd
libnet-daemon-perl  libipc-sharedcache-perl  libwww-perl          doc-base
libsql-statement-perl  libbusiness-isbn-perl  mailx                python3-braceexpand

Sommaire :
Mise à niveau de : 0. Installation de : 48Supprimé : 0. Non mis à jour : 0
Taille du téléchargement : 20,7 MB
Espace nécessaire : 208 MB / 16,4 GB disponible

Continuer ? [O/n]
```

Continuer avec “O”

Préparer la configuration sécurisée du serveur : mariadb-secure-installation

```
root@gsb-bdd:~# mariadb-secure-installation

NOTE: MariaDB is secure by default in Debian. Running this script is
useless at best, and misleading at worst. This script will be
removed in a future MariaDB release in Debian. Please read
mariadb-server.README.Debian for details.

Enter root user password or leave blank:
Enter current password for root (enter for none):
```

Plusieurs questions sont posées :

- “Enter current password for root (enter for none):” Appuyer sur Entrée
- You already have your root account protected, so you can safely answer 'n'.
- Switch to unix socket authentication [Y/n] : Appuyer sur N car le compte root est déjà configuré
- You already have your root account protected, so you can safely answer 'n'.
Change the root password? [Y/n] : Appuyer sur N car le mot de passe root est déjà configuré
- By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them.
This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment. Remove anonymous users? [Y/n] : Appuyer sur Y pour

sécuriser la base de données en enlevant les utilisateurs anonymes (surtout en production)

- Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network. Disallow root login remotely? [Y/n] : Appuyer sur Y pour garder uniquement une connexion locale
- By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment. Remove test database and access to it? [Y/n] : Appuyer sur Y car aucun intérêt d'avoir une base de tests en production (pour un environnement de tests oui)
- Reloading the privilege tables will ensure that all changes made so far will take effect immediately. Reload privilege tables now? [Y/n] : Appuyer sur Y pour recharger les actions effectuées précédemment

Comme pour la VM Symfony il faut installer UFW puis ouvrir les ports 22 mais cette fois 3306 afin de gérer la base de données.

Importer le projet sur les 2 Vms :

On crée une base de données sur le serveur MariaDB (BDD)

> se connecter à Mariadb

mysql -u root -p

Puis :

```
MariaDB [(none)]> CREATE DATABASE gsb_bdd;
Query OK, 1 row affected (0,005 sec)

MariaDB [(none)]> CREATE USER 'gsb_user'@'192.168.107.224' IDENTIFIED BY 'motdepasse';
Query OK, 0 rows affected (0,011 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON gsb_bdd.* TO 'gsb_user'@'192.168.107.224';
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,004 sec)

MariaDB [(none)]> exit
```

Ensuite on modifie le fichier de conf pour accepter les accès externes :

nano /etc/mysql/mariadb.conf.d/50-server.cnf

On bind en 0.0.0.0 au lieu du local 127.0.0.1

```

#user                = mysql
pid-file              = /run/mysqld/mysqld.pid
basedir               = /usr
#datadir              = /var/lib/mysql
#tmpdir               = /tmp

# Broken reverse DNS slows down connections considerably and name resolve
# safe to skip if there are no "host by domain name" access grants
#skip-name-resolve

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address          = 0.0.0.0

# * Fine Tuning

```

On sauvegarde avec Ctrl + X puis "O"

ensuite on restart le système

systemctl restart mariadb

```

[root@gsb-bdd:~# systemctl restart mariadb
root@gsb-bdd:~# █

```

On vérifie bien que la machine écoute bien sur le port en extérieur après UFW

```

[root@gsb-bdd:~# systemctl restart mariadb
[root@gsb-bdd:~# ss -tlnp | grep 3306
LISTEN 0      80          0.0.0.0:3306      0.0.0.0:*       users:(("mariadb",pid=1123,fd=28))
root@gsb-bdd:~# █

```

Depuis la VM Symfony :

on se place dans le dossier /var/www/html avec cd

```

root@gsb-symfony:~# cd /var/www/html
root@gsb-symfony:/var/www/html# █

```

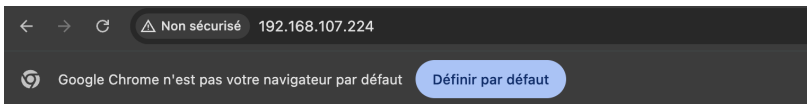
On supprime l'index d'Apache (la page "It works") :

rm /var/www/html/index.html

```

root@gsb-symfony:/var/www/html# rm /var/www/html/index.html
root@gsb-symfony:/var/www/html# █

```



Index of /

Name	Last modified	Size	Description
bin/	2026-03-11 12:17	-	
composer.json	2026-03-11 12:17	2.9K	
composer.lock	2026-03-11 12:17	315K	
config/	2026-03-11 12:17	-	
migrations/	2026-03-11 12:17	-	
public/	2026-03-11 12:17	-	
src/	2026-03-11 12:17	-	
symfony.lock	2026-03-11 12:17	6.7K	
templates/	2026-03-11 12:17	-	

Apache/2.4.66 (Debian) Server at 192.168.107.224 Port 80

On configure :

nano /etc/apache2/sites-available/symfony.conf. Cela crée le fichier de configuration clone le projet (avec une demande de connexion)

git clone -b ldap https://gitlab.com/Sunliaa/notes-de-frais-gsb.git . (je veux cloner ma branche ldap en particulier)

> Demande de connexion au compte gitlab

```
root@gsb-symfony:/var/www/html# git clone -b ldap https://gitlab.com/Sunliaa/notes-de-frais-gsb.git .
Clonage dans '.'...
Username for 'https://gitlab.com': leape@orange.fr
Password for 'https://leape%40orange.fr@gitlab.com':
remote: Enumerating objects: 1742, done.
remote: Counting objects: 100% (404/404), done.
remote: Compressing objects: 100% (395/395), done.
remote: Total 1742 (delta 259), reused 0 (delta 0), pack-reused 1338 (from 1)
Réception d'objets: 100% (1742/1742), 454.54 Kio | 3.50 Mio/s, fait.
Résolution des deltas: 100% (1196/1196), fait.
root@gsb-symfony:/var/www/html#
```

Lorsque cela est fait l'IP pointe sur le projet, on va configurer l'host

de configuration, que l'on va ajuster : nano /etc/apache2/sites-available/symfony.conf

```
<VirtualHost *:80>
    DocumentRoot /var/www/html/public

    <Directory /var/www/html/public>
        AllowOverride All
        Require all granted
        FallbackResource /index.php
    </Directory>
    SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
</VirtualHost>
```

Sauvegarder le fichier avec Ctrl + X puis O

```
root@gsb-symfony:/var/www/html# nano /etc/apache2/sites-available/symfony.conf  
root@gsb-symfony:/var/www/html# █
```

On applique ensuite la configuration puis on redémarre le service

a2ensite symfony.conf

a2dissite 000-default.conf

a2enmod rewrite

systemctl restart apache2

```
root@gsb-symfony:/var/www/html# a2ensite symfony.conf  
Enabling site symfony.  
To activate the new configuration, you need to run:  
  systemctl reload apache2  
root@gsb-symfony:/var/www/html# █
```

```
root@gsb-symfony:/var/www/html# a2dissite 000-default.conf  
Site 000-default disabled.  
To activate the new configuration, you need to run:  
  systemctl reload apache2  
root@gsb-symfony:/var/www/html# █
```

```
root@gsb-symfony:/var/www/html# a2enmod rewrite  
Module rewrite already enabled  
root@gsb-symfony:/var/www/html# █
```

```
root@gsb-symfony:/var/www/html# systemctl restart apache2
```

Avant que le projet soit fonctionnel il faut configurer l'environnement de Symfony et la base de données

Pour récupérer les dépendances :

composer install

```

- Installing api-platform/http-cache (v4.2.14): Extracting archive
- Installing api-platform/symfony (v4.2.14): Extracting archive
- Installing symfony/doctrine-bridge (v8.0.0): Extracting archive
- Installing symfony/dependency-injection (v8.0.5): Extracting archive
- Installing symfony/config (v8.0.0): Extracting archive
- Installing symfony/routing (v8.0.0): Extracting archive
- Installing symfony/cache-contracts (v3.6.0): Extracting archive
- Installing symfony/cache (v8.0.5): Extracting archive
- Installing symfony/framework-bundle (v8.0.0): Extracting archive
- Installing doctrine/sql-formatter (1.5.3): Extracting archive
- Installing doctrine/doctrine-bundle (3.1.0): Extracting archive
- Installing doctrine/data-fixtures (2.2.0): Extracting archive
- Installing doctrine/doctrine-fixtures-bundle (4.3.1): Extracting archive
- Installing symfony/stopwatch (v8.0.0): Extracting archive
- Installing doctrine/migrations (3.9.5): Extracting archive
- Installing doctrine/doctrine-migrations-bundle (4.0.0): Extracting archive
- Installing symfony/lock (v8.0.5): Extracting archive
- Installing symfony/expression-language (v8.0.4): Extracting archive
- Installing phpstan/phpdoc-parser (2.3.2): Extracting archive
- Installing phpdocumentor/reflection-common (2.2.0): Extracting archive
- Installing phpdocumentor/type-resolver (1.12.0): Extracting archive
- Installing nikic/php-parser (v5.7.0): Extracting archive
- Installing jolicode/automapper (dev-main 98069b1): Extracting archive
- Installing symfony/security-http (v8.0.0): Extracting archive
- Installing symfony/security-csrf (v8.0.0): Extracting archive
- Installing psr/clock (1.0.0): Extracting archive
- Installing symfony/clock (v8.0.0): Extracting archive
- Installing symfony/security-bundle (v8.0.0): Extracting archive
- Installing lcobucci/jwt (5.6.0): Extracting archive
- Installing lexik/jwt-authentication-bundle (v3.2.0): Extracting archive
- Installing nelmio/cors-bundle (2.6.1): Extracting archive
- Installing webmozart/assert (2.1.2): Extracting archive
- Installing phpdocumentor/reflection-docblock (5.6.6): Extracting archive
- Installing symfony/dotenv (v8.0.0): Extracting archive
- Installing symfony/polyfill-intl-icu (v1.33.0): Extracting archive
- Installing symfony/options-resolver (v8.0.0): Extracting archive
- Installing symfony/form (v8.0.0): Extracting archive
- Installing symfony/ldap (v8.0.6): Extracting archive
- Installing symfony/process (v8.0.0): Extracting archive
- Installing symfony/maker-bundle (v1.65.1): Extracting archive
- Installing symfony/polyfill-intl-idn (v1.33.0): Extracting archive
- Installing symfony/mime (v8.0.0): Extracting archive
- Installing twig/twig (v3.22.1): Extracting archive
- Installing symfony/twig-bridge (v8.0.0): Extracting archive
- Installing symfony/yaml (v8.0.0): Extracting archive
- Installing symfony/twig-bundle (v8.0.0): Extracting archive
- Installing twig/extra-bundle (v3.22.1): Extracting archive
- Installing symfony/intl (v8.0.1): Extracting archive
- Installing twig/intl-extra (v3.22.1): Extracting archive
Generating autoload files
76 packages you are using are looking for funding.
Use the `composer fund` command to find out more!

Run composer recipes at any time to see the status of your Symfony recipes.

Executing script cache:clear [OK]
Executing script assets:install public [OK]
root@gsb-symfony:/var/www/html#

```

On crée le fichier de configuration :

```
root@gsb-symfony:/var/www/html# nano .env.local
```

```

APP_ENV=prod
DATABASE_URL="mysql://[redacted]@192.168.107.225:3306/gsb_bdd"
JWT_SECRET_KEY=%kernel.project_dir%/config/jwt/private.pem
JWT_PUBLIC_KEY=%kernel.project_dir%/config/jwt/public.pem
JWT_PASSPHRASE=230ed7fc98a04ca5457833937063cfcf6265d3a756a239ae5463ab7adc701e21
LDAP_HOST=1
LDAP_PORT=389
LDAP_DN=CN=Administrateur,CN=Users,DC=gsb,DC=local
LDAP_BASE_DN=DC=gsb,DC=local

```

Ctrl + X puis O pour sauvegarder

(S la bdd n'a pas été créé sur le serveur, on créé la base de données configurée dans le serveur et dans le fichier
php bin/console doctrine:database:create)

Si c'est déjà fait on peut directement appliquer les migrations
php bin/console doctrine:migrations:migrate

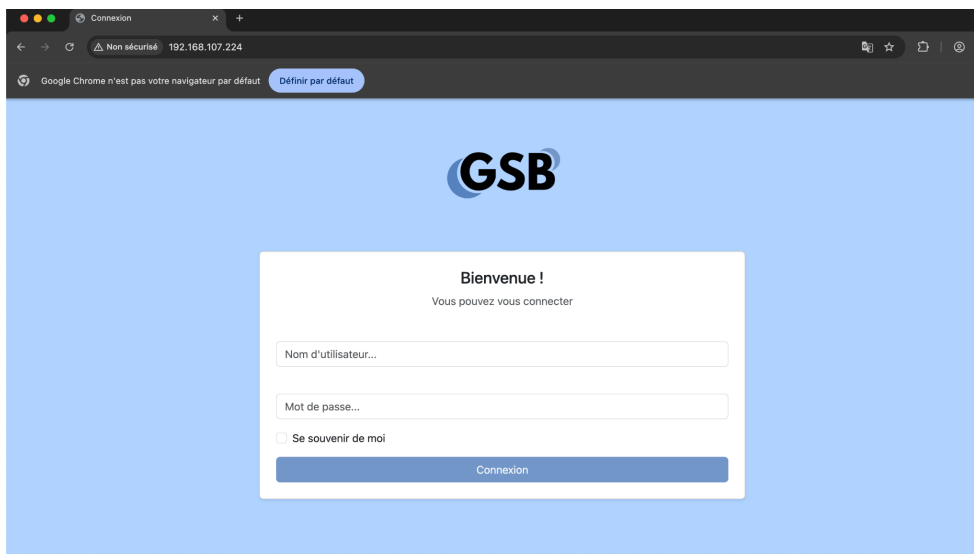
```
WARNING! You are about to execute a migration in database "gsb_bdd" that could result in schema changes and data loss. Are you sure you wish to continue? (yes/no) [yes]:  
>  
[notice] Migrating up to DoctrineMigrations\Version20260303124808  
[notice] finished in 927.7ms, used 16M memory, 43 migrations executed, 142 sql queries  
[OK] Successfully migrated to version: DoctrineMigrations\Version20260303124808  
root@gsb-symfony:/var/www/html#
```

Il manque maintenant les permissions d'écriture dans le dossier /var pour apache et l'utilisateur www-data :

chown -R www-data:www-data /var/www/html/var
chmod -R 775 /var/www/html/var

```
root@gsb-symfony:/var/www/html# chown -R www-data:www-data /var/www/html/var  
[root@gsb-symfony:/var/www/html# chmod -R 775 /var/www/html/var  
root@gsb-symfony:/var/www/html#
```

Le projet est en production



On n'oublie pas de vider le cache

php bin/console cache:clear --env=prod

```
root@gsb-symfony:/var/www/html# php bin/console cache:clear --env=prod  
// Clearing the cache for the prod environment with debug false  
[OK] Cache for the "prod" environment (debug=false) was successfully cleared.  
root@gsb-symfony:/var/www/html#
```