

# Installation

- [installation bookstack](#)
- [SSL](#)
- [Rendu PDF](#)
- [Automysqlbackup](#)
- [Installation redis \(cache\)](#)

# installation bookstack

## Processus d'installation de BookStack sur Debian

---

### Mise à jour du système

Avant de commencer, il est important de mettre à jour votre système Debian.

```
sudo apt update -y
```

```
sudo apt upgrade -y
```

---

### Installation des dépendances

Pour que BookStack fonctionne correctement, il est nécessaire d'installer les paquets suivants :

```
sudo apt-get -y install lsb-release ca-certificates curl
```

```
sudo curl -sSLo /usr/share/keyrings/deb.sury.org-php.gpg https://packages.sury.org/php/apt.gpg
```

```
sudo sh -c 'echo "deb [signed-by=/usr/share/keyrings/deb.sury.org-php.gpg] https://packages.sury.org/php/$(lsb_release -sc) main" > /etc/apt/sources.list.d/php.list'
```

```
sudo apt install -y php8.1 php8.1-{curl,gd,ldap,mbstring,mcrypt,mysql,xm1}
```

```
sudo apt install -y default-mysql-server
```

```
sudo apt install -y git
```

---

### Installation de Composer

Composer est un gestionnaire de dépendances PHP qui est utilisé par BookStack.

```
sudo php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
```

```
sudo php -r "if (hash_file('sha384', 'composer-setup.php') ===  
'55ce33d7678c5a611085589f1f3ddf8b3c52d662cd01d4ba75c0ee0459970c2200a51f492d557530c71c15d8dba01eae') { echo  
'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
```

```
sudo php composer-setup.php
```

```
sudo php -r "unlink('composer-setup.php');"
```

```
sudo mv composer.phar /usr/local/bin/composer
```

---

# Installation d'Apache

Apache est le serveur web utilisé pour exécuter BookStack.

```
sudo apt install -y apache2
```

## Configuration mysql

Execution de sql server et configuration de la BDD

```
mysql -u root
```

```
create database bookstack;
```

```
create user bookstack@localhost identified by 'password';
```

```
grant all privileges on bookstack.* to bookstack@localhost;
```

```
flush privileges;
```

## Téléchargement et installation de BookStack

Il est maintenant temps de télécharger et d'installer BookStack !

```
cd /var/www/
```

```
sudo git clone https://github.com/BookStackApp/BookStack.git --branch release --single-branch
```

```
cd BookStack
```

```
sudo apt install php8.1-zip
```

```
sudo composer install --no-dev
```

```
sudo cp .env.example .env
```

```
sudo php artisan key:generate
```

Vous devez maintenant éditer le fichier .env pour y ajouter les informations de connexion à la base de données :

```
sudo nano .env
```

Recherchez les lignes suivantes :

```
DB_DATABASE=homestead
DB_USERNAME=homestead
DB_PASSWORD=secret
```

Et remplacez-les par les informations de connexion à votre base de données MySQL :

```
DB_DATABASE=bookstack
DB_USERNAME=bookstackuser
DB_PASSWORD=StrongPassword
```

Sauvegardez et fermez le fichier.

Appliquez les autorisations nécessaires :

```
sudo chown -R www-data:www-data bootstrap/cache public/uploads/ storage
```

Enfin, lancez la commande suivante pour générer une clé :

```
php artisan key:generate
```

# Configuration d'Apache

Il ne reste plus qu'à configurer Apache pour qu'il puisse exécuter BookStack.

```
sudo nano /etc/apache2/sites-available/bookstack.conf
```

Insérez le bloc de code suivant :

```
<VirtualHost *:80>

# This is a simple example of an Apache VirtualHost configuration
# file that could be used with BookStack.
# This assumes mod_php has been installed and is loaded.
#
# Change the "docs.example.com" usage in the "ServerName" directive
# to be your web domain for BookStack.
#
# Change the "/var/www/bookstack/public/", used twice below, to the
# location of the "public" folder within your BookStack installation.
#
# This configuration is only for HTTP, Not HTTPS.
# For HTTPS we recommend using https://certbot.eff.org/

ServerName docs.example.com
DocumentRoot /var/www/bookstack/public/

<Directory /var/www/bookstack/public/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
    <IfModule mod_rewrite.c>
        <IfModule mod_negotiation.c>
            Options -MultiViews -Indexes
        </IfModule>

        RewriteEngine On

        # Handle Authorization Header
        RewriteCond %{HTTP:Authorization} .
        RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

        # Redirect Trailing Slashes If Not A Folder...
        RewriteCond %{REQUEST_FILENAME} !-d
        RewriteCond %{REQUEST_URI} (.+)/$
        RewriteRule ^ %1 [L,R=301]

        # Handle Front Controller...
        RewriteCond %{REQUEST_FILENAME} !-d
        RewriteCond %{REQUEST_FILENAME} !-f
        RewriteRule ^ index.php [L]
    </IfModule>
</Directory>

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
```

Remplacez les valeurs de ServerName par vos propres informations, puis sauvegardez et fermez le fichier.

Activez le site et redémarrez Apache :

```
sudo a2ensite bookstack.conf
```

```
sudo sa2enmod rewrite
```

```
sudo a2enmod php8.1
```

```
sudo rm 000-default.conf
```

```
sudo systemctl restart apache2
```

Vous pouvez maintenant accéder à BookStack en ouvrant votre navigateur et en entrant l'URL suivante :

```
http://example.com/
```

# SSL

## 1. Installer snapd :

Tout d'abord, mettez à jour votre système et installez snapd en exécutant les commandes suivantes dans un terminal :

```
$ sudo apt update  
$ sudo apt install snapd
```

## 2. Assurez-vous que votre version de snapd est à jour

Ensuite, mettez à jour la version de snapd en exécutant la commande suivante :

```
$ sudo snap install core  
$ sudo snap refresh core
```

## 3. Installer certbot

Pour installer certbot, exécutez la commande suivante :

```
$ sudo snap install --classic certbot
```

## 4. Préparer la commande certbot

Créez un lien symbolique pour la commande certbot afin de pouvoir y accéder plus facilement :

```
$ sudo ln -s /snap/bin/certbot /usr/bin/certbot
```

## 5. Installer le certificat

Pour installer le certificat, exécutez la commande suivante :

```
$ sudo certbot --apache
```

Suivez les instructions à l'écran pour configurer et installer le certificat SSL.

## 6. Renouvellement du certificat

Pour renouveler le certificat SSL, exécutez la commande suivante :

```
$ sudo certbot renew --dry-run
```

Assurez-vous de renouveler votre certificat avant son expiration pour éviter toute interruption de service.

# Rendu PDF

Choisir la bonne version : <https://wkhtmltopdf.org/downloads.html>

Ensuite, en tant que SUDO :

Télécharger le paquet en faisant la commande suivante :

```
wget https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6.1-2/wkhtmltox_0.12.6.1-2.bullseye_amd64.deb
```

En faisant attention de mettre la bonne version dans l'url (ici debian 11)

Il faut ensuite installer le paquet avec dpkg :

```
dpkg -i wkhtmltox_0.12.6.1-2.bullseye_amd64.deb
```

Vérifier l'installation avec :

```
wkhtmltopdf --version
```

Et voilà, c'est installé!!!

# Automysqlbackup

## Installation du module

```
apt install -y automysqlbackup
```

Se déplacer ensuite sur /etc/default/

et modifier le fichier de config de automysqlbackup :

```
vi automysqlbackup
```

Une fois modifier faite un test avec :

```
/usr/sbin/automysqlbackup
```

Pour voir le résultat aller dans :

```
cd /var/lib/automysqlbackup/
```

et trouver votre sauvegarde si elle a fonctionné.

## Automatisation avec CRON

Vérifier que automysqlbackup fonctionne correctement sinon suivre mon tuto sur mon autre page :).

Pour l'automatisation, il faut créer un script :

```
sudo vi /usr/local/bin/mysqlbackup.sh
```

y insérer ce texte :

```
#!/bin/bash
/usr/sbin/automysqlbackup
```

Une fois le fichier crée, donner les droits exécutions :

```
sudo chmod +x /usr/local/bin/mysqlbackup.sh
```

Ensuite, il faut accéder a Cron pour insérer la ligne permettant de l'automatiser :

```
sudo crontab -e
```

Et y insérer la ligne suivante pour automatiser la sauvegarde tous les dimanches à minuit.

```
0 0 * * 0 /usr/local/bin/mysqlbackup.sh
```



# Installation redis (cache)

## Installation & configuration

Il faut installer redis dans un 1er temps :

```
sudo apt install redis-server -y
```

Ensuite, il faut rajouter au .env de votre site en modifiant les données si besoin :

```
CACHE_DRIVER=redis  
SESSION_DRIVER=redis  
# Example of using a single local Redis server  
REDIS_SERVERS=127.0.0.1:6379:0  
SESSION_LIFETIME=240
```

Puis pour finir, il faut effectuer une commande pour monitorer le tout.

```
redis-cli monitor
```