BookStack

All KB articles related to BookStack

- BookStack Templates
    - [Standard] Template
    - [How-To] Template

- BookStack Installation
    - [How-To] Install Bookstack on Debian 12

- BookStack Configuration
    - [How-To] Configure BookStack to Use SMTP Email

BookStack Templates

BookStack Templates for ease of use when creating new articles.

[Standard] Template

Document Title

Document description.

Purpose

Purpose description.

Standard Name

Standard explained: insert lists, tables, etc.

Add callouts as needed.

[How-To] Template

Purpose

Purpose description.

Prerequisites

List of prerequisites:

- Root user or sudo user
- Debian 12 LXC or VM

Instructions

Section 1

steps for section 1

BookStack Installation

All BookStack installation docs

[How-To] Install Bookstack on Debian 12

Prerequisites

Before commencing, ensure that you have the following:

- A Debian 12 server.
- A non-root user with sudo administrator privileges.
- A domain name pointed to the server IP address.

Installing Dependencies

BookStack is a wiki platform written in PHP and MySQL/MariaDB. At this time, it is fully supported with PHP 8.2. As for this stage, you will install dependencies for BookStack, which includes LAMP Stack (Apache2, MariaDB, PHP) packages, Git, and Composer.

Update and refresh your Debian package index using the following command.

```
sudo apt update
```

Once the repository is updated, install package dependencies for BookStack by executing the following apt install command. With this command, you will install the LAMP Stack package, Composer, and Git.

```
sudo apt install apache2 mariadb-server composer curl php php-xml libapache2-mod-php php-fpm php-curl php-m
```

Type y to proceed with the installation.

After dependencies are installed, verify each dependency by executing the following command.

Verify the apache2 service via the systemctl command below to ensure that the service is running and enabled.

```
sudo systemctl is-enabled apache2
sudo systemctl status apache2
```

The following output will confirm that apache2 service is running and enabled.

Now verify the mariadb service using the command below.

```
sudo systemctl is-enabled mariadb
sudo systemctl status mariadb
```

The output should display that the mariadb service is running and enabled on your system.

Next, verify the PHP version and list enabled extensions using the command below.

```
php -v
php -m
```

Based on the following output, PHP 8.2 is installed on your system.

Lastly, verify the Composer using the following command. This will locate the Composer binary file and check its version.

```
which composer
sudo -u www-data composer -v
```

You should see the Composer **2.5.5** is installed at *usr/bin/composer*.

Configuring PHP

After installing dependencies, the next step is to configure PHP installation by making changes to the php.ini file on your Debian machine.

Open the default PHP configuration */etc/php/8.2/apache2/php.ini* using the following nano editor command.

```
sudo nano /etc/php/8.2/apache2/php.ini
```

Change some default settings with the following configuration. Be sure to adjust the *date.timezone* and *memory_limit* parameters with your server environment.

```
date.timezone = Europe/Amsterdam
memory_limit = 512M
```

Save the file and exit the editor when finished.

Now run the following systemctl command to restart the apache2 service and apply the changes you've made.

```
sudo systemctl restart apache2
```

Configuring MariaDB Server

Now that PHP is configured, the next step you will configure the MariaDB server installation using the *mariadb-secure-installation* utility, which allows you to set up basic security for MariaDB. Then, you will also create a new database and user that will be used by BookStack.

Execute the following *mariadb-secure-installlation* command to secure your MariaDB installation.

```
sudo mariadb-secure-installation
```

During the process, you will be asked about some configurations. Type Y to agree and apply the new changes, or type n for No to reject the configuration.

- For the default MariaDB server installation without a root password, press ENTER when asked about the password.
- The local authentication for MariaDB root users is secured by default, input 'n' when asked to change the authentication method to 'unix_socket'.
- Input 'Y' to create a new MariaDB root password. Then, input the strong password for your MariaDB root user and repeat.
- When asked to disable remote authentication for the MariaDB root user, input 'Y' to agree.
- The default MariaDB server installation comes with the database 'test' and allows an anonymous user to access it. Input 'Y' for both settings to remove the default database 'test' and remove anonymous privilege.
- Lastly, input 'Y' to confirm reloading table privileges.

After configuring MariaDB, you will create a new MariaDB database and user that will be used by BookStack via the mariadb client command.

Log in to the MariaDB server using the following mariadb command. Input your MariaDB root password when prompted.

```
sudo mariadb -u root -p
```

Now run the following queries to create a new database and user for BookStack. In this example, you will create a new database and user **bookstack**, with the password **p4ssword**.

```
CREATE DATABASE bookstack;
CREATE USER bookstack@localhost IDENTIFIED BY 'p4ssword';
GRANT ALL ON bookstack.* TO bookstack@localhost WITH GRANT OPTION;
FLUSH PRIVILEGES;
```

Next, run the following query to verify the privileges for the user **bookstack**.

```
SHOW GRANTS FOR bookstack@localhost;
```

You should see the user **bookstack** is allowed to access and manage the database **bookstack** that will be used by BookStack.

Type quit to exit from the MariaDB Server.

Downloading Bookstack

Now that you've configured the PHP and MariaDB server, you're ready to install BookStack. In this section, you will install BookStack via Git and Composer.

First, create new directories */var/www/.config* and */var/www/.cache* that will be used by Composer for storing dependencies cache and configuration. Then, you must also change the ownership of both directories to the user *www-data*.

```
mkdir -p /var/www/{.config,.cache}
sudo chown -R www-data /var/www/{.config,.cache}
```

Move to the */var/www* directory and download BookStack source code to the **bookstack** directory. Your BookStack installation directory should be **/var/www/bookstack**.

```
cd /var/www
git clone https://github.com/BookStackApp/BookStack.git --branch release --single-branch bookstack
```

Now change the ownership of */var/www/bookstack* directory to user *www-data* and move into it.

```
sudo chown -R www-data:www-data /var/www/bookstack
cd /var/www/bookstack
```

Next, copy the **.env.example** file to **.env** and open the new file using the following nano editor command.

```
sudo -u www-data cp .env.example .env
sudo -u www-data nano .env
```

Input your domain name to the **APP_URL** option and input your MariaDB database details to the **DB_** settings below.

```
# The application URL
APP_URL=http://bookstack.hwdomain.io

# Database values also need to be modified
# If you follow previous command, you only need to add your DB_PASSWORD
DB_DATABASE=bookstack
DB_USERNAME=bookstack
DB_PASSWORD=p4ssword
```

Save and close the file when you're done.

Next, run the following command to install PHP dependencies via Composer.

```
sudo -u www-data composer install --no-dev --no-plugins
```

During the installation, the following output will be displayed.

After PHP dependencies are installed, execute the following command to generate the secret key and migrate the database.

```
sudo -u www-data php artisan key:generate --no-interaction --force
sudo -u www-data php artisan migrate --no-interaction --force
```

The secret key on the *.env* file will be updated, and below is the process during the database migration.

After everything is finished, run the following command to change the ownership of some directories to the *www-data* user and ensure those directories are writeable.

```
sudo chown www-data:www-data -R bootstrap/cache public/uploads storage
sudo chmod u+rw bootstrap/cache public/uploads storage
```

Lastly, run the command below to change the permission of the .env file to 640. This will ensure only user www-data will be able to make changes to the file.

```
sudo chmod -R 640 /var/www/bookstack/.env
```

Configuring Apache2 Virtual Host

In the following section, you will create a new Apache2 virtual host configuration that will be used to run BookStack. So, you must ensure that you've prepared your domain name and ensure it is pointed to the server IP address.

Before creating a virtual host configuration, execute the following command to enable the rewrite module on Apache2.

```
sudo a2enmod rewrite
```

Now create a new virtual host configuration */etc/apache2/sites-available/bookstack.conf* using the following nano editor command.

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

Insert the following configuration and be sure to change the domain name within the ServerName option. The following example will be using the domain **bookstack.hwdomain.io** for the BookStack installation.

```
<VirtualHost *:80>
    ServerName bookstack.hwdomain.io
    ServerAdmin webmaster@localhost
    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}/bookstack-error.log
    CustomLog ${APACHE_LOG_DIR}/bookstack-access.log combined
</VirtualHost>
```

Save and close the file when you're done.

Next, run the following command to activate the virtual host file bookstack.conf and verify your Apache2 syntax.

```
sudo a2ensite bookstack.conf
sudo apachectl configtest
```

If you use proper syntax, the output "**Syntax OK**" will be printed out to your terminal.

Lastly, run the systemctl command below to restart the aapche2 service and apply the changes that you've made.

```
sudo systemctl restart apache2
```

Securing Bookstack with SSL/TLS Certificates

In this guide, you will secure BookStack with SSL/TLS certificates, which can be generated from Letsencrypt via the Certbot tool.

Install Certbot and Certbot Apache plugin via the apt install command below. Type y to confirm and proceed with the installation.

```
sudo apt install certbot python3-certbot-apache
```

Now run the certbot command below to generate SSL/TLS certificates and secure your BookStack installation. Be sure to change the domain name and email address within the following command.

```
sudo certbot --apache --agree-tos --redirect --hsts --staple-ocsp --email admin@hwdomain.io -d bookstack.hwdoma
```

After the process is finished, your SSL/TLS certificates will be available in the */etc/letsencrypt/live/bookstack.hwdomain.io* directory. Also, your BookStack installation will automatically configured with HTTPS, which is configured via the Certbot Apache plugin.

Accessing Bookstack

At this point, everything is configured and your BookStack installation is finished. You can now access your BookStack installation.

Launch your web browser and visit the domain name of your BookStack installation, such as

https://bookstack.hwdomain.io/, and you will be redirected to the BookStack login page.

Input the default email address **admin@admin.com** and the password is **password**, then click **Login**.

If successful, you should see the BookStack dashboard like the following:

Conclusion

To wrap up, you've successfully installed BookStack on Debian 12 server step-by-step. You've installed BookStack via Git, Composer, and with LAMP Stack (Apache2, MariaDB, and PHP) packages. You've also secured BookStack with SSL/TLS certificates from Letsencrypt. From here, you can now populate your information and build your own documentation.

Online Reference: https://www.howtoforge.com/how-to-install-bookstack-on-debian-12/

BookStack Configuration

All docs related to BookStack configuration

[How-To] Configure BookStack to Use SMTP Email

Configure BookStack to use SMTP for email

Learn how to configure BookStack wiki to use SMTP for email sending.

Prerequisites:

- Login into the BookStack service over SSH.
- Have your SMTP credentials ready.

Step 1: Open the BookStack .env file

Open the BookStack .env file using the following command.

```
nano /var/www/bookstack/.env
```

Step 2: Find the default email settings

The email settings start at the 'MAIL_DRIVER' line.

BookStack default email settings
Image not found or type unknown

Step 3: Update the email settings

Update the email settings to reflect your SMTP service. Here are the default settings.

```
# Mail system to use
# Can be 'smtp' or 'sendmail'
MAIL_DRIVER=smtp

# Mail sender details
MAIL_FROM_NAME="BookStack"
MAIL_FROM=bookstack@example.com

# SMTP mail options
# These settings can be checked using the "Send a Test Email"
# feature found in the "Settings > Maintenance" area of the system.
MAIL_HOST=localhost
MAIL_PORT=1025
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
```

Here is an example using the K&T SMTP service.

```
# Mail system to use
# Can be 'smtp' or 'sendmail'
MAIL_DRIVER=smtp

# Mail sender details
MAIL_FROM_NAME="MY Wiki"
MAIL_FROM=bookstack@mydomain.com

# SMTP mail options
# These settings can be checked using the "Send a Test Email"
# feature found in the "Settings > Maintenance" area of the system.
MAIL_HOST=smtp.example.com
MAIL_PORT=587
MAIL_USERNAME=bookstack@mydomain.com
MAIL_PASSWORD=mysupersecurepassword
```

```
MAIL_ENCRYPTION=tls
```

When finished, restart the apache2 service:

```
systemctl restart apache2
```

Step 4: Test email sending

BookStack has a email test function under the maintenance area. Login as an administrative user, and navigate to Settings > Maintenance.

BookStack send email test

Click the **SEND TEST EMAIL** link and watch for the success or error message. The message will show up on the top corner of the browser. It should be green and confirm email sending. If not, review the error message and adjust settings for your SMTP service.

This article applies to BookStack version 24+