Bookstack

- Start from YAML / Docker application
- Alter url
- Run in an Iframe
- Solving problems
    - Error starting docker after update
    - Error starting MariaDB after update
    - SMTP: response code "235" but got code "535"
- Run your own Bookstack docker
- Update Database Links to relative paths

Start from YAML / Docker application

It is possible to quickly setup a docker instance.

- This will created them into an isolated network - if you want to use an nginx passthrough, you'll need to put them into the same network as nginx container
- Be aware, that the environment variable ***APP_URL*** is the target URL used through the docker, and cannot easily be changed later.

```
---
version: "2"
services:
  bookstack:
    image: lscr.io/linuxserver/bookstack
    container_name: bookstack
    environment:
      - PUID=1000
      - PGID=1000
      - APP_URL=http://wiki.mydomain.com
      - DB_HOST=192.x.x.x
      - DB_PORT=13456
      - DB_USER=***REDACTED
      - DB_PASS=***REDACTED
      - DB_DATABASE=***REDACTED
    volumes:
      - ./bookstackdata:/config
    ports:
      - 7890:80
    restart: unless-stopped
    depends_on:
      - bookstack_db
  bookstack_db:
    image: lscr.io/linuxserver/mariadb
    container_name: bookstack_db
```

```
environment:
  - PUID=1000
  - PGID=1000
  - MYSQL_ROOT_PASSWORD=987393TGFSGFDKS
  - TZ=Europe/Brussels
  - MYSQL_DATABASE=***REDACTED
  - MYSQL_USER=***REDACTED
  - MYSQL_PASSWORD=***REDACTED
volumes:
  - ./bookstackdbdata:/config
ports:
  - 123456:3306
restart: unless-stopped
```

Alter url

Bookstack reads the environment variable "APP_URL", which unfortunately cannot be altered permanently in Container station.

So, while you've started with a wrong URL, you will end up trying to set it straight.

You can set it your environment variable manually with ***export APP_URL=xxxxx***- but it will only work until the next startup of your docker.

Instead of redeploying a container or app, you can edit it manually.

1) To change the values in the database, use this command:

```
# Searches for  and replaces it with
php artisan bookstack:update-url

# Example:
php artisan bookstack:update-url http://docs.example.com https://demo.bookstackapp.com
```

2) to change the paths in the application, connect to the docker via terminal, and in the path, edit app.php
note: in the file it looks like the environment variable 'APP_URL' is being read.

```
/app/www/app/Config#vi app.php
```

```
    'recycle_bin_lifetime' => env('RECYCLE_BIN_LIFETIME', 30),

    // The limit for all uploaded files, including images and attac
    'upload_limit' => env('FILE_UPLOAD_SIZE_LIMIT', 50),

    // Allow <script> tags to entered within page content.
    // <script> tags are escaped by default.
    // Even when overridden the WYSIWYG editor may still escape scr
    'allow_content_scripts' => env('ALLOW_CONTENT_SCRIPTS', false),

    // Allow server-side fetches to be performed to potentially unk
    // and user-provided locations. Primarily used in exports when
    // in externally referenced assets.
    'allow_untrusted_server_fetching' => env('ALLOW_UNTRUSTED_SERVE

    // Override the default behaviour for allowing crawlers to craw
    // May be ignored if view has be overridden or modified.
    // Defaults to null since, if not set, 'app-public' status used
    'allow_robots' => env('ALLOW_ROBOTS', null),

    // Application Base URL, Used by laravel in development command
    // and used by BookStack in URL generation.
    'url' => 'http://wiki.sophior.com', //env('APP_URL'  '') === 'ht

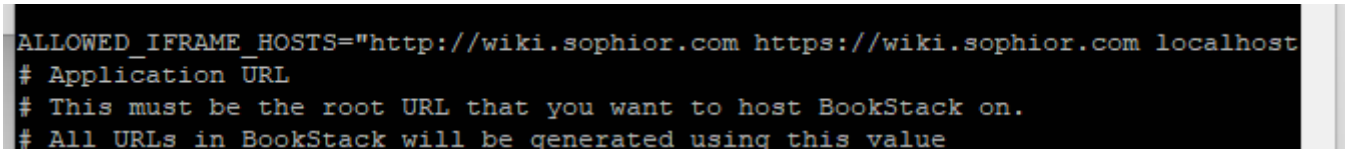
    // A list of hosts that BookStack can be iframed within.
    // Space separated if multiple. BookStack host domain is auto-i
    'iframe_hosts' => env('ALLOWED_IFRAME_HOSTS', null),

    // A list of sources/hostnames that can be loaded within iframe
    // Space separated if multiple. BookStack host domain is auto-i
```

Run in an Iframe

If for some reason you want to run the bookstack in an iframe, you have to modify the .env file to set the allowed urls.
This change is instantly and doesn't require the docker or service to be restarted.

```
cd /config/www
vi .env
```



Solving problems

Solving problems

Error starting docker after update

After an update I came across this error

```
chmod: changing permissions of '/config/nginx/site-confs': Bad address
chmod: changing permissions of '/config/nginx/site-confs/default.conf.sample': Bad address
chmod: changing permissions of '/config/nginx/site-confs/default.conf': Bad address
chmod: changing permissions of '/config/nginx/nginx.conf.sample': Bad address
chmod: changing permissions of '/config/nginx/ssl.conf.sample': Bad address
chmod: changing permissions of '/config/nginx/nginx.conf': Bad address
chmod: changing permissions of '/config/nginx/ssl.conf': Bad address
chmod: changing permissions of '/config/nginx/dhparams.pem': Bad address
chmod: changing permissions of '/config/nginx/resolver.conf': Bad address
chmod: changing permissions of '/config/nginx/worker_processes.conf': Bad address
chmod: changing permissions of '/etc/logrotate.d/acpid': Bad address
chmod: changing permissions of '/etc/logrotate.d/nginx': Bad address
chmod: changing permissions of '/etc/logrotate.d/php-fpm': Bad address
chmod: changing permissions of '/etc/logrotate.d/php-fpm83': Bad address
App Key found - setting variable for seds
```

It was solved by going into the container

```
[~] # docker exec -it 52e39ca076d9 /bin/bash
user@52e39ca076d9:/config/nginx/site-confs# chmod +777 default.conf
user@52e39ca076d9:/config/nginx# chmod +777 nginx.conf
```

Restart container

Solving problems

Error starting MariaDB after update

After an update I saw this in my docker station

```
   Brought to you by linuxserver.io
———————————————————————————————————

To support LSIO projects visit:
https://www.linuxserver.io/donate/

———————————————————————————————————
GID/UID
———————————————————————————————————

User UID:    1000
User GID:    1000
———————————————————————————————————
Linuxserver.io version: 10.11.8-r0-ls155
Build-date: 2024-09-10T07:43:39+00:00
———————————————————————————————————

chmod: changing permissions of '/var/run/mysqld/mysqld.pid': Bad address
s6-rc: warning: unable to start service init-mariadb-config: command exited 1
[migrations] started
[migrations] no migrations found
usermod: no changes
```

Again, ssh into the docker and in this case, removing the mysqld.pid solved the problem:

```
[~] # docker exec -it e2de9fa5c446 /bin/bash
user@e2de9fa5c446:/# cd /var/run
user@e2de9fa5c446:/var/run# cd mysqld
user@e2de9fa5c446:/var/run/mysqld# ls
mysqld.pid  mysqld.sock
user@e2de9fa5c446:/var/run/mysqld# rm mysqld.pid
```

Solving problems

SMTP: response code "235" but got code "535"

This is probably more a PHP error, but I came across it in the context of bookstacks.

When configuring the .env file to connect to a mail server, and testing the email, you might get this error:

```
unexpected response code "235" but got code "535", with message "535 Error: authentication failed".".
Authenticator "PLAIN" returned "Expected response code "235" but got code "535
```

The problem goes away when you put the password in quotes a such:

```
MAIL_PASSWORD="thIsIsYour873#password"
```

Run your own Bookstack docker

After being required to move my server and the LinuxServer BookStack image stopped working. At the same time I've been obligated to migrate my containers.

Delving deeper, I ended up making my own bookstack docker image - which turned out to be quite easy but not as easy as I had hoped. But I believe this path is the easiest to maintain and comprehend, in the end.

It's not "to the letter perfect" but these are the coarse steps to achieve the migratin. The Dockerfile is the largest hurdle, that is pasted below for your convenience.

What we have to achieve in this page:

- Create a container with MariaDB
- Create a docker image that has an installation of Bookstack
  - Requires PHP-FPM
  - Requires Lavarel
  - Requires GIT to pull the latest version

In the next page we will:

- Inspect the existing installation
- Copy the old database to the new one
  - Perform some maintenance tasks to update links in the old data
- Copy the old files to the new container

Backup your files

In the linuxserver there are some recusirve links to different folders, which makes it quite confusing. You could use "ls -l" to discover which are linked. There are all in /var/config somewhere.

Initially, you can start to discover and search for your data by opening an interactive shell onto the docker (this would be likely, to SSH into the QNAP NAS, and executing commands as such:)

```
docker ps
docker exec -i 87652832 /bin/bash
cd /var/config
ls -l
cd /var/www/public
ls -l
```

once you have found your files with going through your directories and are confident you've found your uploaded files:

```
docker ps
docker cp 9YI22323:/var/config/www/config/BOOKSTACK_APP_KEY ./BOOKSTACK_APP_KEY
docker cp 9YI22323:/var/config/www/images ./config/images
docker cp 9YI22323:/var/config/www/files ./config/files

docker cp  9YI22323:/var/www/.env ./.env
```

Create your dockerfile

Don't forget, that we are using a docker with linuxserver, and extracting the images and uploads. Then, recreating a fresh docker, and coying over the files for the image. In my case, I'm working in a folder with a subfolder WikiExport where my files reside. And the database already exists.

I copy them over, to test my docker image locally. And in a deployment stage, we will publish the image and bind these folders to the host. And link the .env file with the host.
So we can easily move them in the future or edit them without digging into the container!

For a new fresh docker, these lines should be uncommented:

WORKDIR /var/www/BookStack
RUN php artisan key:generate
RUN php artisan migrate

```
FROM ubuntu:latest AS base

ENV DEBIAN_FRONTEND noninteractive
# from here https://dev.to/adnanbabakan/dockerizing-laravel-10-ubuntu-image-php-82-fpm-nginx-318p

# Install dependencies
RUN apt update
RUN apt install -y software-properties-common
RUN add-apt-repository -y ppa:ondrej/php
RUN apt update
RUN apt install -y php8.2\
    php8.2-cli\
    php8.2-common\
    php8.2-fpm\
    php8.2-mysql\
    php8.2-zip\
    php8.2-gd\
    php8.2-mbstring\
    php8.2-curl\
    php8.2-xml\
    php8.2-bcmath\
    php8.2-pdo

# Install php-fpm
RUN apt install -y php8.2-fpm php8.2-cli

# Install composer
RUN apt install -y curl
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer

# Install nodejs
RUN apt install -y ca-certificates gnupg
RUN mkdir -p /etc/apt/keyrings
RUN curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | gpg --dearmor -o
/etc/apt/keyrings/nodesource.gpg
ENV NODE_MAJOR 20
RUN echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg]
https://deb.nodesource.com/node_$NODE_MAJOR.x nodistro main" | tee /etc/apt/sources.list.d/nodesource.list
RUN apt update
RUN apt install -y nodejs
```

```
RUN apt install -y git
# Install nginx
RUN apt install -y nginx
RUN echo "\
    server {\n\
        listen 80;\n\
        listen [::]:80;\n\
        root /var/www/BookStack/public;\n\
        add_header X-Frame-Options \"SAMEORIGIN\";\n\
        add_header X-Content-Type-Options \"nosniff\";\n\
        index index.php;\n\
        charset utf-8;\n\
        location / {\n\
            try_files \$uri \$uri/ /index.php?\$query_string;\n\
        }\n\
        location = /favicon.ico { access_log off; log_not_found off; }\n\
        location = /robots.txt  { access_log off; log_not_found off; }\n\
        error_page 404 /index.php;\n\
        location ~ \.php$ {\n\
            fastcgi_pass unix:/run/php/php8.2-fpm.sock;\n\
            fastcgi_param SCRIPT_FILENAME \$realpath_root\$fastcgi_script_name;\n\
            include fastcgi_params;\n\
        }\n\
        location ~ /\.(?!well-known).* {\n\
            deny all;\n\
        }\n\
    }\n" > /etc/nginx/sites-available/default

RUN echo "\
    #!/bin/sh\n\
    echo \"Starting services...\"\n\
    service php8.2-fpm start\n\
    nginx -g \"daemon off;\" &\n\
    echo \"Ready.\"\n\
    tail -s 1 /var/log/nginx/*.log -f\n\
    " > /var/start.sh


RUN apt install -y vim
```

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

# RUN composer install
RUN composer install --no-dev
COPY ./.env .
COPY ./BOOKSTACK_APP_KEY.txt .
#COPY ./WikiExport/config/www/images /var/www/BookStack/public/uploads

# Set the bookstack folders and files to be owned by the user barry and have the group www-data
RUN chown -R www-data:www-data /var/www/BookStack

# Set all bookstack files and folders to be readable, writable & executable by the user (barry) and
# readable & executable by the group and everyone else
RUN chmod -R 755 /var/www/BookStack

# For the listed directories, grant the group (www-data) write-access
RUN chmod -R 775 /var/www/BookStack/storage /var/www/BookStack/bootstrap/cache
/var/www/BookStack/public/uploads

# Limit the .env file to only be readable by the user and group, and only writable by the user.
RUN chmod 640 /var/www/BookStack/.env

#RUN mkdir /config

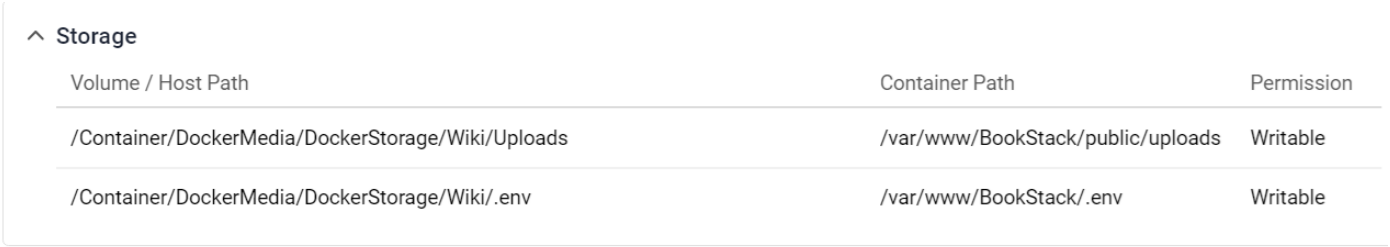
#/var/www/BookStack/public/uploads#

COPY ./WikiExport/config/www/uploads/images /var/www/BookStack/public/uploads/images
COPY ./WikiExport/config/www/files /var/www/BookStack/public/uploads/files
#COPY ./WikiExport/app/www/



# WORKDIR /var/www/BookStack
# RUN php artisan key:generate
# RUN php artisan migrate
EXPOSE 80
CMD ["sh", "/var/start.sh"]
```

So we can run the new container on a host, with a bound path and manipulate the configuration or load/balance/mirror or backup our files without doing all this above work digging into the docker again.



Storage

Volume / Host Path	Container Path	Permission
/Container/DockerMedia/DockerStorage/Wiki/Uploads	/var/www/BookStack/public/uploads	Writable
/Container/DockerMedia/DockerStorage/Wiki/.env	/var/www/BookStack/.env	Writable

Makig paths relative

To avoid some URL dependent code, you can open MySQL and execute simular queries, to change the absolute paths into relative paths.

```
SET SQL_SAFE_UPDATES = 0;
UPDATE  bookstackapp.images SET URL= REPLACE(REPLACE(url, 'http://wiki.sophior.com',
''),'https://wiki.sophior.com','')
UPDATE bookstackapp.settings set value = REPLACE(value, 'https://wiki.sophior.com', '') where value like
'%wiki%'
UPDATE bookstackapp.pages set html = replace(html, 'http://wiki.sophior.com', '') where HTML LIKE '%wiki%';
UPDATE bookstackapp.pages set html = replace(html, 'https://wiki.sophior.com', '') where HTML LIKE '%wiki%';
```

Mixed content error

I have found that it works quite okay if you run it via an internal IP and exposed port.

But if you have an NGINX in front, forwarding from a domain - you really should set your URLS rights.

You need to add to the .env file the line: APP_URL=HTTPS://xxxxxx to avoid mixed content. But from that point, you only can display the website via the external URL.

Update Database Links to relative paths

```
SET SQL_SAFE_UPDATES = 0;
UPDATE  bookstackapp.images SET URL= REPLACE(REPLACE(url, 'http://wiki.yourdomain.com', ''),'https://wiki.yourdomain.com','')
UPDATE bookstackapp.settings set value = REPLACE(value, 'https://wiki.yourdomain.com', '') where value like '%wiki%'
UPDATE bookstackapp.pages set html = replace(html, 'http://wiki.yourdomain.com', '') where HTML LIKE '%wiki%';
UPDATE bookstackapp.pages set html = replace(html, 'https://wiki.yourdomain.com', '') where HTML LIKE '%wiki%';
```