

Backup + Migration

Not tested yet!

“ I installed BookStack via Docker (as in [the guide](#)). Could you please explain how to create backups (full content backup) so that I can migrate the website if needed, e.g., to another server/domain? [Here](#) is BookStack's official guide.

Full BookStack backup (Docker) – so that a move to *another server/domain* works ??

For a **complete, restorable backup**, you essentially need **two components**:

1. **Database** (all pages, books, users, permissions, settings ...)
2. **Files/config** (uploads, attachments, `.env`/APP_KEY, possibly themes/modules)

Since you have the setup like in your guide (`/opt/bookstack` with `docker-compose.yml`, volumes `./mariadb` and `./bookstack`), this is pleasantly simple.

? What exactly needs to be backed up?

1) Database (MariaDB)

- The content is in the DB container, and in your setup it's persisted under:
 - `./mariadb` (as volume `./mariadb:/config`)
- **Cleanest approach**: *SQL dump* (portable, tolerant across versions)

2) Files (BookStack container volume)

- Persistent data is located under:
 - `./bookstack` (as volume `./bookstack:/config`)
- In it, especially important:
 - `.env` (including **APP_KEY!**)
 - uploads/attachments
 - possibly `www/themes/custom/...` (your hacks/modules)

3) Compose/Caddy (helpful for migration)

- `docker-compose.yml`
- `Caddyfile`
- Caddy data optional:
 - `./caddy/data` & `./caddy/config` (certificates/state).
Not strictly required, but sometimes saves re-issuing/rate limits.

?? Create a backup (recommended approach)

“ Assumption: project is located in `/opt/bookstack` and the containers are named as in your YAML:
`bookstack-mariadb` and `bookstack`.

Step 1: Create a backup folder

```
cd /opt/bookstack
mkdir -p backups
```

Step 2: Dump the database as SQL ?

```
docker exec -i bookstack-mariadb \
  mysqldump -u bookstack -pbookstack \
  --single-transaction --routines --triggers \
  bookstack > backups/bookstack-db-$(date +%F).sql
```

Important: Replace `-pbookstack` with your real password (from `MYSQL_PASSWORD`).

Alternative (safer because the password won't end up in shell history):
Omit `-p...`, then `mysqldump` will prompt interactively for the password:

```
docker exec -it bookstack-mariadb mysqldump -u bookstack -p --single-transaction --routines --triggers bookstack > backups/bookstack-db-$(date +%F).sql
```

Step 3: Archive files/volumes ?

```
tar -czvf backups/bookstack-files-$(date +%F).tar.gz \  
docker-compose.yml Caddyfile \  
bookstack mariadb caddy
```

If you **don't** want to back up **Caddy**, you can do it leaner:

```
tar -czvf backups/bookstack-files-$(date +%F).tar.gz \  
docker-compose.yml Caddyfile \  
bookstack mariadb
```

Step 4: (Optional) Integrity check ?

```
ls -lh backups/  
head -n 5 backups/bookstack-db-*.sql  
tar -tzf backups/bookstack-files-*.tar.gz | head
```

? Restore (on a new server) – incl. domain/URL migration

Step 0: Prerequisites

- Docker + Docker Compose installed
 - DNS of the new domain points to the new server (for HTTPS via Caddy)
-

Step 1: Copy the backup over

Pack on the **old** server or copy directly:

```
scp -r /opt/bookstack/backups user@new-server:/opt/bookstack/
```

Step 2: Restore files (Compose/volumes) ?

On the **new** server:

```
cd /opt/bookstack  
tar -xvzf backups/bookstack-files-YYYY-MM-DD.tar.gz
```

(Adjust the date, of course.)

Step 3: Start containers (DB should already be running) ?

```
cd /opt/bookstack  
docker compose up -d mariadb
```

Wait a few seconds until MariaDB is ready.

Step 4: Import the database ?

```
cat backups/bookstack-db-YYYY-MM-DD.sql | \  
docker exec -i bookstack-mariadb mysql -u bookstack -pbookstack bookstack
```

Again: adjust the password or use interactive mode.

Step 5: Start BookStack

```
docker compose up -d
```

? Domain/URL changed? (very important!)

If you move to a **different URL/domain**, you must:

1) Adjust `APP_URL` in the `.env`

In your volume, the `.env` is typically located at:

- `./bookstack/www/.env` (common with linuxserver/bookstack)
- or `./bookstack/.env` (depending on image/setup)

Find it like this:

```
find /opt/bookstack/bookstack -maxdepth 3 -name ".env" -print
```

Then set `APP_URL=` `https://new-domain.tld`.

“ Pay attention: The `APP_KEY` must remain the old one (otherwise encrypted data will break, e.g., MFA secrets).

2) BookStack command: “Update System URL”

So that content/links in the DB are rewritten cleanly:

```
docker exec -it bookstack php /app/www/artisan bookstack:update-url https://old-domain.tld https://new-domain.tld
```

(If the path in the container is different, this often also works: `php artisan ...` from `/app/www/.`)

? Practical notes (so it's truly “migration-safe”)

- Always store **backups offsite** (another system/storage).
- **Database dump + file archive** is the most portable.

- If you restore to a **newer BookStack version**:
 1. Restore DB
 2. Bring containers up
 3. Then run migrations (if needed):

```
docker exec -it bookstack php /app/www/artisan migrate --force
```

If you want: I'll make you a “one-command backup script” ?

Just tell me briefly:

- Is your `.env` located at `bookstack/www/.env`?
- What are your containers actually named (`docker ps` output)?
- Do you want to back up Caddy certificates or preferably not?

Then I'll give you a small script including rotation (e.g., “keep the last 7 backups”) and optional upload to S3/Hetzner Storage Box.

Revision #10

Created 2026-04-03 18:01:17 UTC by art10m

Updated 2026-04-05 16:17:35 UTC by art10m