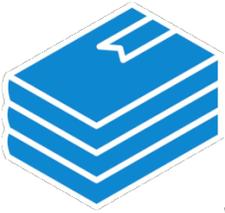# Backups, SMTP Mail & Webhooks

- [Create & Restore Backups Guide](#)
- [Setup Email Configuration for Bookstack](#)
- [Webhooks](#)

Create & Restore Backups Guide

While BookStack does not currently have a built-in way to backup and restore content, it can usually be done via the command line with relative ease. The below commands are based on using Ubuntu. If you are using a different operating system you may have to alter these commands to suit.

Backup

There are two types of content you need to backup: Files and database records.

Database

The easiest way to backup the database is via `mysqldump` :

```
# Syntax
## Only specify the `-p` option if the user provided has a password
mysqldump -u {mysql_user} -p {database_name} > {output_file_name}



# Example
mysqldump -u benny bookstack > bookstack.backup.sql
```

If you are using MySQL on Ubuntu, and are using the `root` MySQL user, you will likely have to run the command above with `sudo` :

```
sudo mysqldump -u root bookstack > bookstack.backup.sql
```

The resulting file (`bookstack.backup.sql` in the examples above) will contain all the data from the database you specified. Copy this file to somewhere safe, ideally on a different device.

Files

Below is a list of files and folders containing data you should back up. The paths are shown relative to the root BookStack folder.

- `.env` - File, contains important configuration information.
- `public/uploads` - Folder, contains any uploaded images.
- `storage/uploads` - Folder, contains uploaded page attachments.
- `themes` - Folder, contains any configured <u>visual/logical themes</u>.

Alternatively you could backup up your whole BookStack folder but only the above contain important instance-specific data by default.

The following command will create a compressed archive of the above folders and files:

```
tar -czvf bookstack-files-backup.tar.gz .env public/uploads storage/uploads themes
```

The resulting file (`bookstack-files-backup.tar.gz`) will contain all your file data. Copy this to a safe place, ideally on a different device.

Automatic Backup Script

backupscript.sh

```
#!/bin/bash


# Directory to store backups within
# Should not end with a slash and not be stored within
# the BookStack directory
BACKUP_ROOT_DIR="$HOME"


# Directory of the BookStack install
# Should not end with a slash.
BOOKSTACK_DIR="/var/www/bookstack"


# Get database options from BookStack .env file
```

```
export $(cat "$BOOKSTACK_DIR/.env" | grep ^DB_ | xargs)

# Create an export name and location
DATE=$(date "+%Y-%m-%d_%H-%M-%S")
BACKUP_NAME="bookstack_backup_$DATE"
BACKUP_DIR="$BACKUP_ROOT_DIR/$BACKUP_NAME"
mkdir -p "$BACKUP_DIR"

# Dump database to backup dir using the values
# we got from the BookStack .env file.
mysqldump --single-transaction \
 --no-tablespaces \
 -u "$DB_USERNAME" \
 -p"$DB_PASSWORD" \
 "$DB_DATABASE" > "$BACKUP_DIR/database.sql"

# Copy BookStack files into backup dir
cp "$BOOKSTACK_DIR/.env" "$BACKUP_DIR/.env"
cp -a "$BOOKSTACK_DIR/storage/uploads" "$BACKUP_DIR/storage-uploads"
cp -a "$BOOKSTACK_DIR/public/uploads" "$BACKUP_DIR/public-uploads"

# Create backup archive
tar -zcf "$BACKUP_DIR.tar.gz" \
 -C "$BACKUP_ROOT_DIR" \
 "$BACKUP_NAME"

# Cleanup non-archive directory
rm -rf "$BACKUP_DIR"

echo "Backup complete, archive stored at:"
echo "$BACKUP_DIR.tar.gz"
```

Restore

If you are restoring from scratch follow the <u>installation</u> instructions first to get a new BookStack instance set-up but **do not run the** `php artisan migrate` **installation step when installing BookStack**. You may need to comment this command out if using an installer script.

If you are using a docker-container-based set-up, restore the database before running the BookStack container. An example of the process using a linuxserver.io-based docker-compose setup can be seen <u>in our video here</u>.

Database

To restore the database you simply need to execute the sql in the output file from the `mysqldump` you performed above. To do this copy your database SQL backup file onto the BookStack or database host machine and run the following:

```
# Syntax
mysql -u {mysql_user} -p {database_name} < {backup_file_name}
## Only specify the -p if the user provided has a password


# Example
mysql -u benny -p bookstack < bookstack.backup.sql


# If using the root user on Ubuntu you may
# have to run the above with root permissions via sudo:
sudo mysql -u root bookstack < bookstack.backup.sql
```

If you are restoring to a new version of BookStack you will have to run `php artisan migrate` after restore to perform any required updates to the database.

Files

To restore the files you simply need to copy them from the backup archive back to their original locations. If you created a compressed `bookstack-files-backup.tar.gz` archive as per the backup instructions above you can simply copy that file to your BookStack folder then run the following command:

```
tar -xvzf bookstack-files-backup.tar.gz
```

If you get errors during the above command it may be due to permissions. Change permissions so you can write to the restore locations.

After a backup of the files you should reset the permissions to ensure any write-required locations are writable by the server. The locations required for this can be found in the installation instructions.

Configuration (.env File)

During a restore, you may end up merging various configuration options between your old and new instance `.env` files, to get things working for the new environment. For example, it's common to use the old `.env` settings for most things but use database settings from the `.env` file of a newly created instance.

One thing to be aware of is that you should use the `APP_KEY` value of the old `.env` file since this is used for various features like the encryption of multi-factor authentication credentials. Changing the `APP_KEY` may cause such features to break.

URL Changes

If you are restoring into an environment where BookStack will run on a different URL, there are a couple of things you'll need to do after restoring everything:

- Within the `.env` config file update the `APP_URL` value to exactly match your new base URL.
- Run the "Update System URL" command to update your database content to use your new URL.

If you migrated web-server configuration files, you may also need to tweak those to correctly use the new URL.

Setup Email Configuration for Bookstack

BookStack sends out emails for a range of purposes such as email-address confirmation & "forgot password" flows. Both SMTP and Sendmail (Linux Sendmail) are supported email mechanisms.

SMTP

To get up and running with SMTP you will need to add, or set, the following variables in your `.env` file or `docker-compose.yaml`:

```
MAIL_DRIVER=smtp

# SMTP server host address
MAIL_HOST=smtp.provider.tld

# SMTP server port
# Using port 465 will force connections to be via TLS
MAIL_PORT=587

# Connection encryption to use
# Valid values are: tls, null
# Using 'tls' will require either TLS or STARTTLS to be used.
# When using 'null' STARTTLS will still be attempted if announced
# as supported by your SMTP server.
# Using port 465 above will force connections to be via TLS.
```

```
MAIL_ENCRYPTION=tls

# Authentication details for your SMTP service
MAIL_USERNAME=user@provider.tld
MAIL_PASSWORD=onlyifneeded

# The "from" email address for outgoing email
MAIL_FROM=noreply@yourdomain.tld

# The "from" name used for outgoing email
MAIL_FROM_NAME=BookStack
```

Connection TLS/SSL Certificate Verification

In some cases your SMTP server may be using a private/self-signed TLS/SSL certificate that would usually fail certificate verification. In these cases its common for that certificate (Or its CA) to be added to the BookStack's host trusted certificate database. If that's not possible, you can alternatively disable SSL/TLS certificate verification for mail sending by adding this setting to your `.env` file or `docker-compose.yaml` :

```
# Verify SSL/TLS certificates during SMTP sending
# WARNING: Disabling verification using a 'false' value
# can make you vulnerable to MITM attacks
MAIL_VERIFY_SSL=false
```

Sendmail

The `sendmail` drivers uses the sendmail application on the host system. By default it will call `/usr/sbin/sendmail -bs` although this is configurable.
To enable this option you can set the following in your `.env` file or `docker-compose.yaml` :

```
MAIL_DRIVER=sendmail

# The "from" email address for outgoing email
MAIL_FROM=noreply@yourdomain.tld

# The "from" name used for outgoing email
MAIL_FROM_NAME=BookStack

# The command to use for calling sendmail
```

```
MAIL_SENDMAIL_COMMAND="/usr/sbin/sendmail -bs"
```

Debugging Email

You can follow the instructions provided in the <u>debugging documentation page</u> to help gain more details about issues you may come across. Within the "Settings > Maintenance" area of BookStack you can find a "Send a Test Email" action which provides a quick & easy way to send emails after changing your configuration. This action will also attempt to capture any errors thrown and display them.

Webhooks

https://foss.video/w/xu4T7mafyLqkLU1VTgNaCV