

# Mautic

Novo Aplicativo de e-mail marketing

- Repositórios Mautic
  - Mautic docker oficial

# Repositórios Mautic

Repositórios Mautic

Repositórios Mautic

# Mautic docker oficial

Link: <https://github.com/mautic/docker-mautic>  
[mautic/tree/mautic4](https://github.com/mautic/docker-mautic/tree/mautic4)

Link2: [https://github.com/mautic/docker-](https://github.com/mautic/docker-mautic/tree/mautic4)

git clone <https://github.com/mautic/docker-mautic.git>

## Docker Mautic Image



## License

Mautic is distributed under the GPL v3 license. Full details of the license can be found in the [Mautic GitHub repository](#).

## How to use Docker-Mautic

Docker-Mautic provides a basic environment for Mautic to run correctly.

Docker images are always simple and functional by design, if you need to add specific features to your Mautic Setup just create a new image based on this official image. You can access and customize Docker Mautic from [Official Docker Hub image](#).

*This repository refers to Mautic 3 Series. If you would like information about Mautic 2, see [README](#) at 'mautic2.x' folder.*

# Pulling Mautic images from Docker Hub

If you want to pull the latest **stable** image from Mautic 3 Series on DockerHub:

```
docker pull mautic/mautic:v4
```

***Note that during the 4.0.x period, the 'mautic/mautic:latest' tag still refers to Mautic 2 for backward compatibility. If you intend to use Mautic 2, use the 'mautic/mautic:v2' tag instead of 'mautic/mautic:latest'.***

If you want to pull the latest **stable** image from Mautic 4 Series on DockerHub:

```
docker pull mautic/mautic:v4
```

If you want to pull the latest **stable** image based on Apache2 from Mautic 4 Series on DockerHub:

```
docker pull mautic/mautic:v4-apache
```

If you want to pull the latest **stable** image based on FPM from Mautic 4 Series on DockerHub:

```
docker pull mautic/mautic:v4-fpm
```

If you want to pull the latest **stable** image from Mautic 3 Series on DockerHub:

```
docker pull mautic/mautic:v3
```

If you want to pull the latest **stable** image based on Apache2 from Mautic 3 Series on DockerHub:

```
docker pull mautic/mautic:v3-apache
```

If you want to pull the latest **stable** image based on FPM from Mautic 3 Series on DockerHub:

```
docker pull mautic/mautic:v3-fpm
```

# Running Basic Container

Setting up a Network to connect Mautic and MySQL:

```
$ docker network create mauticnet
```

Setting up MySQL Server 5.7+ (Percona, MariaDB or MySQL):

```
$ docker volume create mysql_data

$ docker run --name database -d \
  --restart=always \
  -p 3306:3306 \
  -e MYSQL_ROOT_PASSWORD=mypassword \
  -v mysql_data:/var/lib/mysql \
  --net=mauticnet \
  percona/percona-server:5.7 \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_general_ci
```

Setting Up Mautic:

```
$ docker volume create mautic_data

$ docker run --name mautic -d \
  --restart=always \
  -e MAUTIC_DB_HOST=database \
  -e MAUTIC_DB_USER=root \
  -e MAUTIC_DB_PASSWORD=mypassword \
  -e MAUTIC_DB_NAME=mautic \
  -e MAUTIC_RUN_CRON_JOBS=true \
  -p 8080:80 \
  --net=mauticnet \
  -v mautic_data:/var/www/html \
  mautic/mautic:v3
```

This will run a basic Mautic on <http://localhost:8080>.

# Building your own containers

You can build your own containers easily using the docker build command in the root of this directory:

```
docker build . -f apache/Dockerfile -t mautic/mautic:v4-apache
docker build . -f fpm/Dockerfile -t mautic/mautic:v4-fpm
```

## Customizing Mautic Container

The following environment variables are also honored for configuring your Mautic instance:

### Database Options

- `-e MAUTIC_DB_HOST=...` (defaults to the IP and port of the linked `mysql` container)
- `-e MAUTIC_DB_USER=...` (defaults to "root")
- `-e MAUTIC_DB_PASSWORD=...` (defaults to the value of the `MYSQL_ROOT_PASSWORD` environment variable from the linked `mysql` container)
- `-e MAUTIC_DB_NAME=...` (defaults to "mautic")
- `-e MAUTIC_DB_TABLE_PREFIX=...` (defaults to empty) Add prefix do Mautic Tables. Very useful when migrate existing databases from another server to docker.

If you'd like to use an external database instead of a linked `mysql` container, specify the hostname and port with `MAUTIC_DB_HOST` along with the password in `MAUTIC_DB_PASSWORD` and the username in `MAUTIC_DB_USER` (if it is something other than `root`).

If the `MAUTIC_DB_NAME` specified does not already exist on the given MySQL server, it will be created automatically upon startup of the `mautic` container, provided that the `MAUTIC_DB_USER` specified has the necessary permissions to create it.

## Mautic Options

- `-e MAUTIC_RUN_CRON_JOBS=...` (defaults to true - enabled) If set to true runs mautic cron jobs using included cron daemon
- `-e MAUTIC_RUN_MIGRATIONS=...` (defaults to false - disabled) If set to true runs database migrations automatically on startup.
- `-e MAUTIC_TRUSTED_PROXIES=...` (defaults to empty) If Mautic sits behind a reverse proxy, you can set a json array of CIDR network addresses here, and mautic will set those

addresses as trusted proxies. You can use ["0.0.0.0/0"] or See [documentation](#)

- -e MAUTIC\_CRON\_HUBSPOT=... (defaults to empty) Enables mautic crons for Hubspot CRM integration
- -e MAUTIC\_CRON\_SALESFORCE=... (defaults to empty) Enables mautic crons for Salesforce integration
- -e MAUTIC\_CRON\_PIPEDRIVE=... (defaults to empty) Enables mautic crons for Pipedrive CRM integration
- -e MAUTIC\_CRON\_ZOHO=... (defaults to empty) Enables mautic crons for Zoho CRM integration
- -e MAUTIC\_CRON\_SUGARCRM=... (defaults to empty) Enables mautic crons for SugarCRM integration
- -e MAUTIC\_CRON\_DYNAMICS=... (defaults to empty) Enables mautic crons for Dynamics CRM integration

## PHP options

- -e PHP\_INI\_DATE\_TIMEZONE=... (defaults to UTC) Set PHP timezone
- -e PHP\_MEMORY\_LIMIT=... (defaults to 256M) Set PHP memory limit
- -e PHP\_MAX\_UPLOAD=... (defaults to 20M) Set PHP upload max file size
- -e PHP\_MAX\_EXECUTION\_TIME=... (defaults to 300) Set PHP max execution time

## PHP options

- -e PHP\_INI\_DATE\_TIMEZONE=... (defaults to UTC) Set PHP timezone
- -e PHP\_MEMORY\_LIMIT=... (defaults to 256M) Set PHP memory limit
- -e PHP\_MAX\_UPLOAD=... (defaults to 20M) Set PHP upload max file size
- -e PHP\_MAX\_EXECUTION\_TIME=... (defaults to 300) Set PHP max execution time

## Persistent Data Volumes

On first run Mautic is unpacked at /var/www/html. You need to attach a volume on this path to persist data.

## Mautic Versioning

Mautic Docker has two ENV that you can specify an version do start your new container:

- -e MAUTIC\_VERSION (Defaults to "3.0.0")
- -e MAUTIC\_SHA1 (Defaults to "ed4287367b8484aa146a1fa904b261ab30d9c6e7")

# Automated installation

If you wish your mautic instance to automatically run the installer when the container is first started, provide the following environment variables:

- `-e MAUTIC_URL` The URL at which your mautic instance will be accessed
- `-e MAUTIC_ADMIN_EMAIL` The email address of your initial admin user
- `-e MAUTIC_ADMIN_PASSWORD` The password of your initial admin user

You can also optionally provide other installation variables:

- `-e MAUTIC_INSTALL_FORCE` If the URL used above is HTTP instead of HTTPS, automated installation will fail. Set to true to allow this.
- `-e MAUTIC_ADMIN_USERNAME` The username of your initial admin user
- `-e MAUTIC_ADMIN_FIRSTNAME` The first name of your initial admin user
- `-e MAUTIC_ADMIN_LASTNAME` The last name of your initial admin user

## Mautic configuration

Additional configuration variables can be added to mautic's local config by prepending them with `MAUTIC_CONFIG_`. This will be re-applied every time the container is re-started, so can be used for external config management of mautic instances.

- `-e MAUTIC_CONFIG_DEBUG=1` Set debug=1 in `local.php`

## Accessing the Instance

Access your new Mautic on `http://localhost:8080` or `http://host-ip:8080` in a browser.

## Add SSL to your Mautic

If you change the *Site Address* of *Mautic General Settings tab* to HTTPS (behind a reverse proxy), you can use `0.0.0.0/0` as Trusted Proxies to avoid a redirect loop error. See [documentation](#)

... via `docker-compose`



Example `docker-compose.yml` for `mautic`:

```
version: "2"

services:
  database:
    image: powertic/percona-docker
    container_name: database
    environment:
      MYSQL_ROOT_PASSWORD: mypassword
    ports:
      - "3306:3306"
    volumes:
      - database:/var/lib/mysql
    restart: always
    networks:
      - mauticnet
    command: --character-set-server=utf8mb4 --collation-server=utf8mb4_general_ci --sql-mode=""

  mautic:
    container_name: mautic
    image: mautic/mautic:v4-apache
    volumes:
      - mautic_data:/var/www/html
    environment:
      - MAUTIC_DB_HOST=database
      - MAUTIC_DB_USER=root
      - MAUTIC_DB_PASSWORD=mypassword
      - MAUTIC_DB_NAME=mautic4
    restart: always
    networks:
      - mauticnet
    ports:
      - "8880:80"
```

Run `docker-compose up`, wait for it to initialize completely, and visit `http://localhost:8080` or `http://host-ip:8080`.

“ This compose file was tested on compose file version 3.0+ (docker engine 1.13.0+), see the relation of compose file and docker engine [here](#).

# Supported Docker versions

This image is officially supported on Docker version 1.7.1.

Support for older versions (down to 1.0) is provided on a best-effort basis.

# User Feedback

## Issues

If you have any problems with or questions about this image, please contact us through a [GitHub issue](#).

You can also reach the Mautic community through its [online forums](#) or the [Mautic Slack channel](#).

## Contributing

You are invited to contribute new features, fixes, or updates, large or small; we are always thrilled to receive pull requests, and do our best to process them as fast as we can.

Before you start to code, we recommend discussing your plans through a [GitHub issue](#), especially for more ambitious contributions. This gives other contributors a chance to point you in the right direction, give you feedback on your design, and help you find out if someone else is working on the same thing.