

Installing Mautic Email Marketing Tool Using Docker: A Simplified Guide

Mautic, an open-source marketing automation tool, can be installed in various ways, but this guide focuses on using Docker for its simplicity and efficiency. Docker allows for easy deployment and management of Mautic, making it an ideal choice for users looking for a straightforward installation process. While other methods exist, this guide emphasizes the Docker approach as it streamlines the setup and configuration of Mautic.

Installing Mautic with Docker

Pulling the Docker Image

To begin the installation process, the first step is to pull the latest stable Mautic image from Docker Hub. Run the following command in your terminal.

```
docker pull mautic/mautic:latest
```

Running Basic Containers

Before running Mautic, it's necessary to set up a MySQL server. Create a Docker volume for MySQL data:

```
docker volume create mysql_data
```

Next, run the MySQL container (using Percona as an example):

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

Now, create a Docker volume for Mautic data:

```
docker volume create mautic_data
```

Finally, run the Mautic container:

```
docker run --name mautic -d \  
  --restart=always \  
  -e MAUTIC_DB_HOST=127.0.0.1 \  
  -e MAUTIC_DB_NAME=mautic_db \  
  -e MAUTIC_DB_USER=mautic_db_user \  
  -e MAUTIC_DB_PASSWORD=mautic_db_password
```

```
-e MAUTIC_DB_USER=root \
-e MAUTIC_DB_PASSWORD=mypassword \
-e MAUTIC_DB_NAME=mautic \
-e MAUTIC_RUN_CRON_JOBS=true \
-e MAUTIC_TRUSTED_PROXIES=0.0.0.0/0 \
-p 8080:80 \
-v mautic_data:/var/www/html \
mautic/mautic:latest
```

This will launch a basic MySQL service within Mautic, accessible at <http://localhost:8080>.

Customizing Mautic Container

Database Options

Mautic container provides various environment variables for database configuration:

- `MAUTIC_DB_HOST`: Specifies the IP and port of the linked MySQL container (defaults to 127.0.0.1).
- `MAUTIC_DB_USER`: Sets the database user (defaults to "root").
- `MAUTIC_DB_PASSWORD`: Defines the database user's password (defaults to `MySQL_ROOT_PASSWORD` from the linked MySQL container).
- `MAUTIC_DB_NAME`: Specifies the database name (defaults to "mautic").
- `MAUTIC_DB_TABLE_PREFIX`: Adds a prefix to Mautic tables (defaults to empty).

If using an external database, provide the hostname, port, password, and username accordingly.

Mautic Options

Several environment variables control Mautic's behavior:

- `MAUTIC_RUN_CRON_JOBS`: Enables/disables Mautic cron jobs (defaults to true).
- `MAUTIC_TRUSTED_PROXIES`: Specifies trusted proxies for reverse proxy setups.
- `MAUTIC_CRON_HUBSPOT`, `MAUTIC_CRON_SALESFORCE`, and others: Enable crons for specific CRM integrations.

Enable/Disable Features

- `MAUTIC_TESTER`: Enables Mautic Github Pull Tester documentation.

PHP Options

- `PHP_INI_DATE_TIMEZONE`: Sets the PHP timezone (defaults to UTC).
- `PHP_MEMORY_LIMIT`: Configures PHP memory limit (defaults to 256M).
- `PHP_MAX_UPLOAD`: Defines the maximum file size for PHP uploads (defaults to 20M).
- `PHP_MAX_EXECUTION_TIME`: Specifies the maximum execution time for PHP scripts (defaults to 300).

Persistent Data Volumes

Mautic stores data at `/var/www/html`, requiring a volume for persistence.

Mautic Versioning

Specify the Mautic version and SHA1 hash using the following environment variables:

- `MAUTIC_VERSION` (Defaults to "2.15.0").
- `MAUTIC_SHA1` (Defaults to "b07bd42bb092cc96785d2541b33700b55f74ece7").

Accessing the Mautic Instance

After successfully setting up Mautic, access it at <http://localhost:8080> or <http://host-ip:8080> in a web browser.

Using Docker-Compose

For a more organized setup, use Docker Compose. Here's an example `docker-compose.yml` file:

```
version: '2'

services:

  mauidb:
    image: percona/percona-server:5.7
    container_name: mauidb
    volumes:
      - mysql_data:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=mysecret
    command:
      --character-set-server=utf8mb4 --collation-server=utf8mb4_general_ci
    networks:
      - mauid-net

  mauid:
    image: mauid/mauid:latest
    container_name: mauid
    links:
      - mauidb:mysql
    depends_on:
      - mauidb
```

```
ports:
  - 8080: 80

volumes:
  - mautic_data: /var/www/html

environment:
  - MAUTIC_DB_HOST=mauticdb
  - MYSQL_PORT_3306_TCP=3306
  - MAUTIC_DB_USER=root
  - MAUTIC_DB_PASSWORD=mysecret
  - MAUTIC_DB_NAME=mautic
  - MAUTIC_RUN_CRON_JOBS=true

networks:
  - mautic-net
```

```
volumes:
  mysql_data:
    driver: local
  mautic_data:
    driver: local

networks:
  mautic-net:
    driver: bridge
```

Run `docker-compose up`, wait for initialization, and access Mautic at <http://localhost:8080> or <http://host-ip:8080>.

Conclusion

By opting for Docker as the installation method for Mautic, users benefit from a simplified and efficient process. This guide covers the basics of pulling the image, running containers, customization options, and utilizing Docker Compose for a more organized deployment. Access Mautic through the provided URLs and leverage its powerful marketing automation features with ease.

Revision #1
Created 28 January 2024 18:40:58 by Admin
Updated 11 February 2024 19:37:48 by Admin