How to Install Mautic 5 on Ubuntu 22.04 with SSL and Node.js 20

How to Install Mautic 5 on Ubuntu 22.04 with SSL and Node.js 20

This guide shows you how to install **Mautic 5** on a fresh **Ubuntu 22.04** DigitalOcean droplet with **4 GB of RAM**. We'll configure Apache, MariaDB, PHP 8.1, Node.js 20, and Certbot (for Let's Encrypt SSL) — ensuring the certificate is installed before running the Mautic web-based setup.

---

1. Create a 4GB Ubuntu 22.04 Droplet

1. Log into DigitalOcean and create a Droplet.
2. Choose Ubuntu 22.04 (LTS) as the OS image.
3. Select the Basic plan with 4 GB of RAM (or higher).
4. Under "Authentication," add your SSH keys or choose a root password.
5. (Optional) Add your domain if you have one, or skip for now.
6. Click **Create Droplet**.

Once created, copy the public IP address of your new droplet. You'll use this to SSH in and to access Mautic in a browser.

2. Initial System Update

```
apt-get update -yapt-get upgrade -yreboot
```

This updates and upgrades server packages to their latest versions. A reboot is typically a good idea afterward on a new installation.

3. Install Apache

```
apt-get install -y apache2systemctl enable apache2systemctl start apa
che2systemctl status apache2
```

Visit http://YOUR_DROPLET_IP in a browser to confirm Apache's default page loads.

4. Install MariaDB and Secure It

```
apt-get install -y mariadb-server mariadb-clientsystemctl enable mari
adbsystemctl start mariadbmysql_secure_installation
```

Follow the prompts to set a root password, remove the test database, etc.

5. Create the Mautic Database and User

Next, create a dedicated database and user for Mautic:

```
mysql -e "CREATE DATABASE mautic;"mysql -e "CREATE USER 'mautic_user'
@'localhost' IDENTIFIED BY 'YourStrongPassword';"mysql -e "GRANT ALL
PRIVILEGES ON mautic.* TO 'mautic_user'@'localhost';"mysql -e "FLUSH
PRIVILEGES;"
```

You can use any names you like, but in this example we create a database mautic and a user mautic_user with a secure password.

6. Install PHP 8.1 and Extensions

```
apt-get install -y php8.1 php8.1-cli php8.1-curl php8.1-mbstring php8
.1-mysql  php8.1-xml php8.1-zip php8.1-intl php8.1-gd php8.1-imap php
8.1-bcmath  libapache2-mod-php8.1 unzip
```

php8.1-imap is important if Mautic will handle inbound email. If not needed, you can omit it.

7. Install Node.js 20 (Required by Mautic 5)

```
apt-get remove --purge -y nodejs libnode-devcurl -fsSL https://deb.no
desource.com/setup_20.x | bash -apt-get install -y nodejsnode -v # Sh
ould show v20.xnpm -v # Should show 10.x
```

Mautic 5's build process and certain plugins require Node.js 20 and npm 10.

8. Install and Configure Composer

```
curl -sS https://getcomposer.org/installer -o composer-setup.phpphp c
omposer-setup.php --install-dir=/usr/local/bin --filename=composerrm
composer-setup.phpsudo -u www-data composer -V
```

We'll continue to run Composer as www-data to avoid permissions issues.

9. Enable Apache Rewrite and AllowOverride

```
a2enmod rewritea2enmod headerssystemctl restart apache2
```

10. Create the Mautic Directory and Set Ownership

```
mkdir -p /var/www/mauticchown -R www-data:www-data /var/www
```

Ensuring www-data can write here simplifies installation via Composer later.

11. Configure Apache Virtual Host

Create (or edit) /etc/apache2/sites-available/mautic.conf:

```
<VirtualHost *:80> ServerName your-domain.com ServerAlias www.your-do
main.com DocumentRoot /var/www/mautic/docroot <Directory /var/www/mau
tic/docroot> AllowOverride All Require all granted </Directory> Error
Log ${APACHE_LOG_DIR}/mautic_error.log CustomLog ${APACHE_LOG_DIR}/ma
utic_access.log combined</VirtualHost>
```

```
a2ensite mautic.confa2dissite 000-default.conf # optionalsystemctl re
load apache2
```

12. Point Your Domain to the Droplet (DNS)

1. In your domain registrar's DNS settings, create an **A record** for example.com pointing to your droplet's public IP.
2. (Optional) Create a **CNAME** for www.example.com pointing to example.com.
3. Wait for DNS to propagate (can take minutes to hours).

13. Install Certbot and Obtain SSL Certificate

Let's Encrypt will secure your domain before you run Mautic's web-based setup.

```
sudo apt-get updatesudo apt-get install -y certbot python3-certbot-
```

```
apachecertbot --version # Optional: verifysudo certbot --apache
```

Follow the prompts to select your domain, enable HTTPS, and optionally force redirect from HTTP to HTTPS.

14. Install Mautic 5 via Composer (as www-data)

```
sudo -u www-data composer create-project mautic/recommended-project:^5 /var/www/mautic --no-interaction
```

This downloads the Mautic 5.x codebase into /var/www/mautic, owned by www-data.

15. Adjust PHP Settings

```
sed -i 's/^memory_limit =.*/memory_limit = 512M/' /etc/php/8.1/apache2/php.inised -i 's/^upload_max_filesize =.*/upload_max_filesize = 64M/' /etc/php/8.1/apache2/php.inised -i 's/^post_max_size =.*/post_max_size = 64M/' /etc/php/8.1/apache2/php.inised -i 's/^max_execution_time =.*/max_execution_time = 300/' /etc/php/8.1/apache2/php.inisystemctl restart apache2
```

16. Complete Mautic Setup in the Browser

Now that SSL is configured, visit **https://example.com** to run Mautic's web-based setup:

- Mautic Setup Wizard appears.
- Provide your database info:
    - Database Host: localhost
    - Database Name: mautic

        - Database User: mautic_user
        - Database Password: *the password you set*
    - Set up an admin account with email, username, and password.
    - Click **Next** to finalize.
    - Log in and start configuring your Mautic instance over HTTPS!

Tip: If you see a 404 error after install, confirm your DocumentRoot is /var/www/mautic/docroot (or public, depending on your Mautic structure) and that .htaccess is allowed in your Apache config.

You now have a fully functional **Mautic 5** instance with SSL in place before opening the Mautic installer in a browser. From here, you can configure cron jobs, integrate email providers, and use Mautic's powerful marketing automation features!

Online URL: https://kb.mautic.org/article/how-to-install-mautic-5-on-ubuntu-22-04-with-ssl-and-node-js-20.html