Install WordPress on Ubuntu 22.04 with Nginx, MariaDB, PHP8.1 (LEMP)

Last Updated: September 13th, 2022 Xiao Guoan (Admin) 14 Comments Ubuntu

This tutorial is going to show you how to install **WordPress** on Ubuntu 22.04 with Nginx, MariaDB, and PHP8.1 (LEMP Stack). WordPress is the most popular CMS (Content Management System) in the world. It is estimated that more than a third of websites today are powered by WordPress. PHP8.1 is made into the Ubuntu 22.04 repository and the latest version of WordPress runs perfectly with it.

Prerequisites

1. To follow this tutorial, you need an Ubuntu 22.04 OS running on a remote server. If you are looking for a virtual private server (VPS), I recommend Kamatera VPS, which features:

- 30 days free trial.
- Starts at $4/month (1GB RAM)
- High-performance KVM-based VPS
- 9 data centers around the world, including the United States, Canada, UK, Germany, The Netherlands, Hong Kong, and Isreal.

Follow the tutorial linked below to create your Linux VPS server at Kamatera.

- How to Create a Linux VPS Server on Kamatera

2. You also need a domain name, so visitors can type a domain name in the web browser address bar to access your website. I registered my domain name at NameCheap because the price is low and they give whois privacy protection free for life.

3. This tutorial assumes that you have already set up a LEMP stack on Ubuntu 22.04. If not, please check out the following tutorial.

- How to Install LEMP Stack (Nginx, MariaDB, PHP8.1) on Ubuntu 22.04

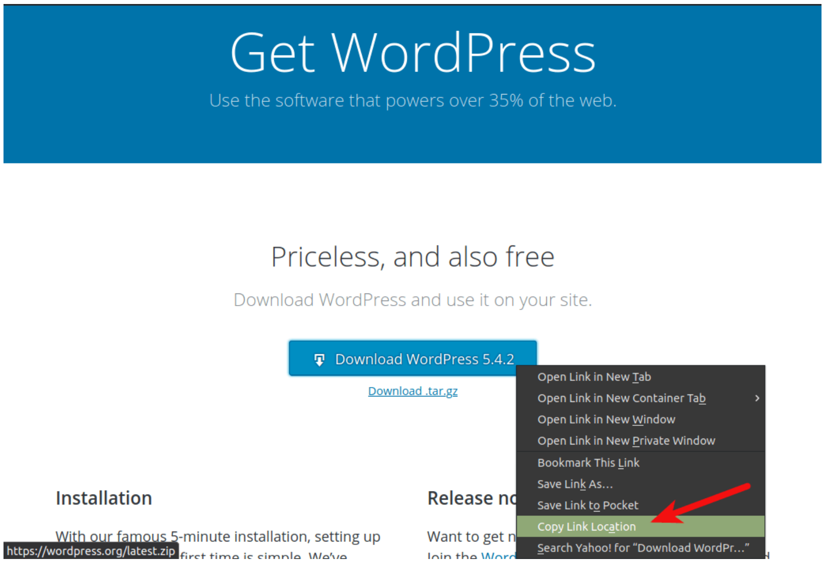
After finishing LEMP installation, come back here and read on.

Step 1: Download WordPress

SSH into your Ubuntu 22.04 server and update existing software.

```
sudo apt update && sudo apt upgrade -y
```

Next, go to wordpress.org download page and download the zip archive. You can acquire the direct download link by right-clicking the download button and select `copy link location`.



Get WordPress

Use the software that powers over 35% of the web.

Priceless, and also free

Download WordPress and use it on your site.

Installation

With our famous 5-minute installation, setting up

https://wordpress.org/latest.zip first time is simple. We've

Release no

Want to get n

Join the Word

Then at the command line prompt, type in **wget** followed by the direct download link to download WordPress to your Ubuntu 22.04 server.

```
wget https://wordpress.org/latest.zip
```

Next, extract the zip archive using the command below.

```
sudo apt install unzip

sudo mkdir -p /usr/share/nginx

sudo unzip latest.zip -d /usr/share/nginx/
```

The archive will be extracted to `/usr/share/nginx/` directory. A new directory named `wordpress` will be created (/usr/share/nginx/wordpress). Now we can rename it like below, so it's easy for us to identify each directory. Replace `example.com` with your real domain name.

```
sudo mv /usr/share/nginx/wordpress /usr/share/nginx/example.com
```

Step 2: Create a Database and User for WordPress Site

Log into MariaDB console as root with the following command.

```
sudo mariadb -u root
```

or

```
sudo mysql -u root
```

Once you are logged in, create a database for WordPress using the following command. I named it `wordpress`, but you can use whatever name you like such as your site name. (Don't leave out the semicolon.)

```
create database wordpress;
```

Then enter the command below to create a database user for WordPress. Replace `wpuser` and `your-password` with your preferred username and password.

```
create user wpuser@localhost identified by 'your-password';
```

Grants all privileges of WordPress database to the user.

```
grant all privileges on wordpress.* to wpuser@localhost;
```

Flush the privileges table for the changes to take effect and then get out of MariaDB console.

```
flush privileges;

exit;
```

Step 3: Configure WordPress

Go to your WordPress directory.

```
cd /usr/share/nginx/example.com/
```

Copy the sample configuration file and rename it to `wp-config.php`.

```
sudo cp wp-config-sample.php wp-config.php
```

Now edit the new config file with a command-line text editor like Nano.

```
sudo nano wp-config.php
```

Find the following lines and replace the red texts with the database name, username and password you created in the previous step.

```
/** The name of the database for WordPress */
define('DB_NAME', 'database_name_here');

/** MySQL database username */
define('DB_USER', 'username_here');

/** MySQL database password */
define('DB_PASSWORD', 'password_here');
```

Then scroll down to find the following line.

```
$table_prefix = 'wp_';
```

By default, every WordPress database table name begins with `wp_` as the prefix. It's highly recommended to change it to something else to improve security. Use random characters like below.

```
$table_prefix = '9OzB3g_';
```

Save and close the file. To save the file in Nano text editor, press `Ctrl+O`, then press `Enter` to confirm. Next, press `Ctrl+X` to exit.

Because this file contains a password, we run the following command to make sure only the file owner can read this file.

```
sudo chmod 640 wp-config.php
```

We also need to set the Nginx user (`www-data`) as the owner of the WordPress site directory by using the following command.

```
sudo chown www-data:www-data /usr/share/nginx/example.com/ -R
```

Step 4: Create an Nginx Server Block for WordPress

We will create the server block file in `/etc/nginx/conf.d/` directory. The file name must end with `.conf`.

```
sudo nano /etc/nginx/conf.d/example.com.conf
```

Put the following texts into the file. Replace the red texts with your own domain name. Don’t forget to create A records for your domain name in your DNS manager.

```
server {
  listen 80;
  listen [::]:80;
  server_name www.example.com example.
com;
  root /usr/share/nginx/example.com/;
  index index.php index.html index.htm
index.nginx-debian.html;

  error_log /var/log/nginx/wordpress.e
rror;
  access_log /var/log/nginx/wordpress.
access;

  location / {
    try_files $uri $uri/ /index.php;
  }

   location ~ ^/wp-json/ {
     rewrite ^/wp-json/(.*?)$ /?rest_r
oute=/$1 last;
   }

  location ~* /wp-sitemap.*\.xml {
    try_files $uri $uri/ /index.php$is
_args$args;
  }

  error_page 404 /404.html;
  error_page 500 502 503 504 /50x.htm
l;

  client_max_body_size 20M;

  location = /50x.html {
    root /usr/share/nginx/html;
  }

  location ~ \.php$ {
    fastcgi_pass unix:/run/php/php8.1-fpm.sock;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    include fastcgi_params;
    include snippets/fastcgi-php.conf;
  }

  #enable gzip compression
  gzip on;
  gzip_vary on;
  gzip_min_length 1000;
  gzip_comp_level 5;
  gzip_types application/json text/css application/x-javascript application/javascript image/svg+xml;
  gzip_proxied any;

  # A long browser cache lifetime can speed up repeat visits to your page
  location ~* \.(jpg|jpeg|gif|png|webp|svg|woff|woff2|ttf|css|js|ico|xml)$ {
       access_log        off;
       log_not_found     off;
       expires           360d;
  }

  # disable access to hidden files
  location ~ /\.ht {
      access_log off;
      log_not_found off;
      deny all;
    }
}
```

Save and close the file. Then test Nginx configurations.

```
sudo nginx -t
```

If the test is successful, reload Nginx.

```
sudo systemctl reload nginx
```

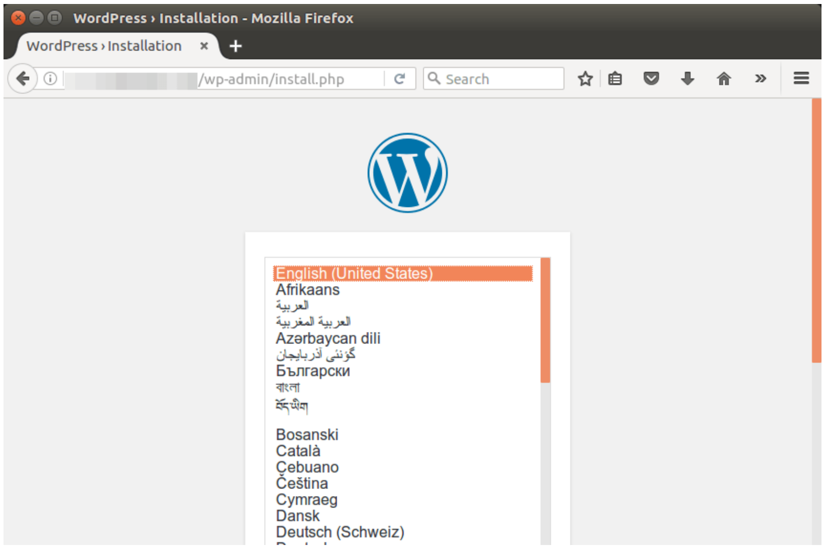
Enter your domain name in the browser address bar.

```
example.com
```

or

```
example.com/wp-admin/install.php
```

You shall see the WordPress installation wizard. Select a language.



If the installation wizard isn't displayed, you probably need to install some PHP extensions.

```
sudo apt install php-imagick php8.1-fpm php8.1-mbstring php8.1-bcmath php8.1-xml php8.1-mysql php8.1-common php8.1-gd php8.1-cli php8.1-curl php8.1-zip
```

Then reload PHP-FPM and Nginx. The wizard should now be displayed.

```
sudo systemctl reload php8.1-fpm nginx
```

Before entering your sensitive information in the setup wizard, it’s recommended to enable HTTPS to prevent traffic hijacking.

Step 5: Enabling HTTPS

To encrypt the HTTP traffic, we can enable HTTPS by installing a free TLS certificate issued from Let’s Encrypt. Run the following command to install Let’s Encrypt client (certbot) on Ubuntu 22.04 server.

```
sudo apt install certbot python3-certbot-nginx
```

And run this command to obtain and install TLS certificate.

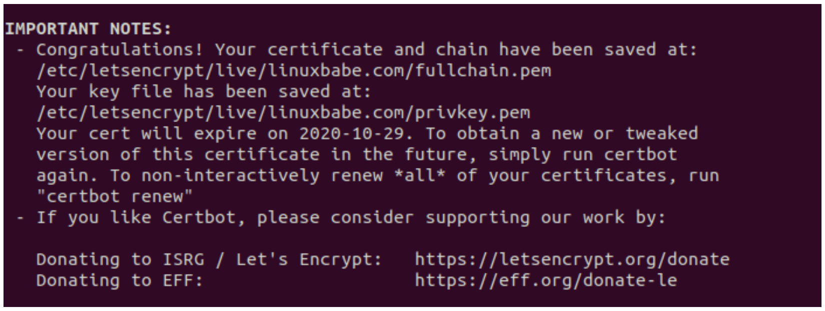
```
sudo certbot --nginx --agree-tos --redirect --hsts --staple-ocsp --email you@example.com -d yourdomain.com,www.yourdomain.com
```

Wher

- `--nginx`: Use the Nginx plugin.

- `--agree-tos`: Agree to terms of service.
- `--redirect`: Force HTTPS by 301 redirect.
- `--hsts`: Add the Strict-Transport-Security header to every HTTP response. Forcing the browser to always use TLS for the domain. Defends against SSL/TLS Stripping.
- `--staple-ocsp`: Enables OCSP Stapling. A valid OCSP response is stapled to the certificate that the server offers during TLS.
- `--email`: Email used for registration and recovery contact.
- `-d` flag is followed by a list of domain names, separated by comma. You can add up to 100 domain names.

The certificate should now be obtained and automatically installed.



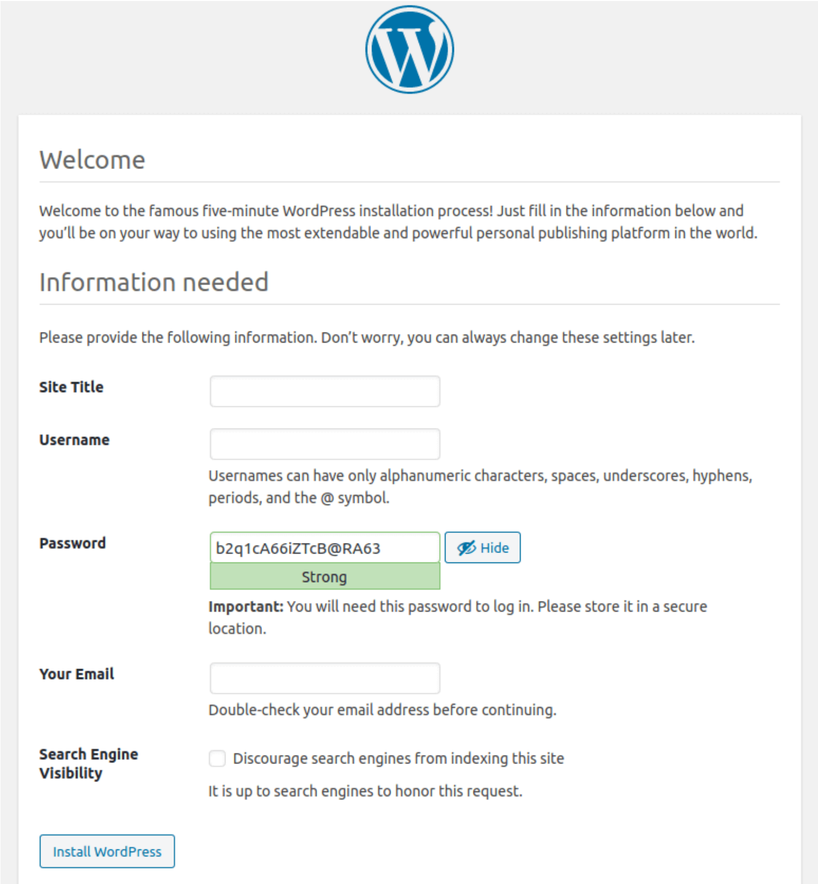
```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/linuxbabe.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/linuxbabe.com/privkey.pem
   Your cert will expire on 2020-10-29. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

Now if you reload the WordPress setup wizard, you can see that HTTP is automatically redirected to HTTPS connection.

Step 6: Finish the Installation with the Setup Wizard

Create an admin account and click the **Install WordPress** button.



Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title

Username

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password b2q1cA66iZTcB@RA63 Hide

Strong

Important: You will need this password to log in. Please store it in a secure location.

Your Email

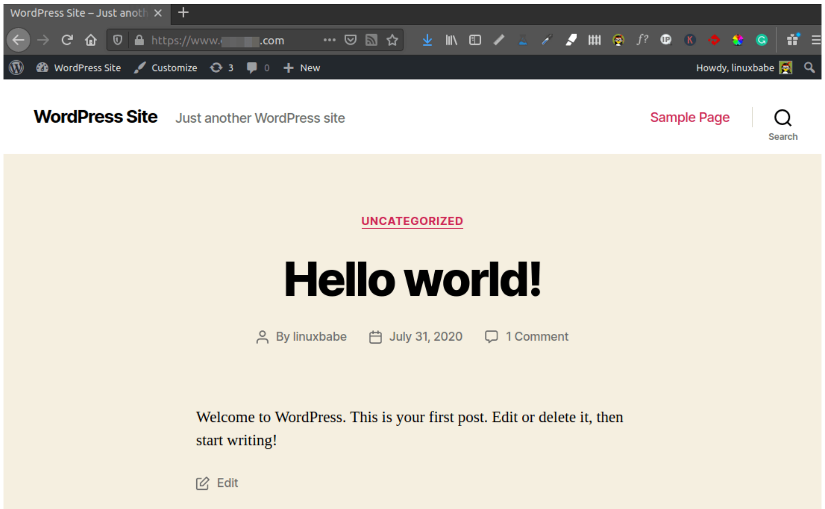
Double-check your email address before continuing.

Search Engine Visibility □ Discourage search engines from indexing this site

It is up to search engines to honor this request.

Install WordPress

And now your new WordPress site is installed.



WordPress Site Just another WordPress site

Sample Page Search

UNCATEGORIZED

Hello world!

By linuxbabe July 31, 2020 1 Comment

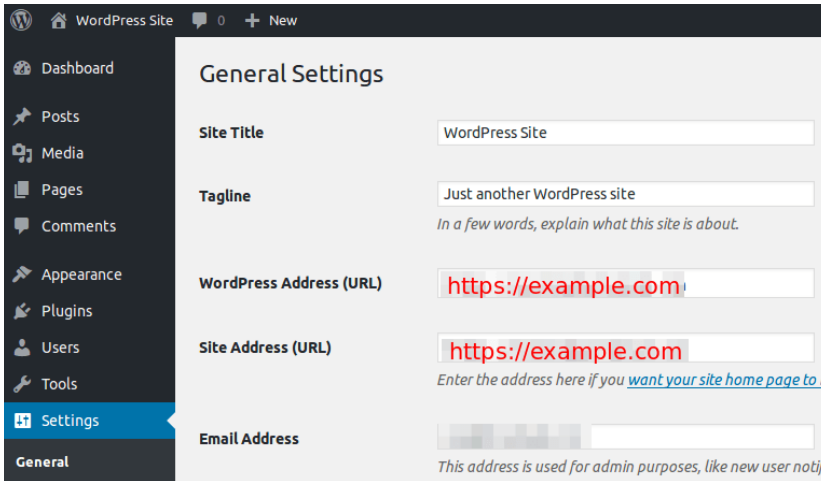
Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

Edit

How to Redirect www to non-www (Or Vice Versa)

We have already enabled redirecting HTTP to HTTPS, what's left to do is redirect www to non-www, or vice versa. It's very easy. Simply go to **WordPress Dashboard > Settings >**

General and set your preferred version (www or non-www) in **WordPress Address** and **Site Address**. Be sure to include the `https://` prefix.



General Settings

Site Title: WordPress Site

Tagline: Just another WordPress site

In a few words, explain what this site is about.

WordPress Address (URL): https://example.com

Site Address (URL): https://example.com

Enter the address here if you want your site home page to

Email Address: [redacted]

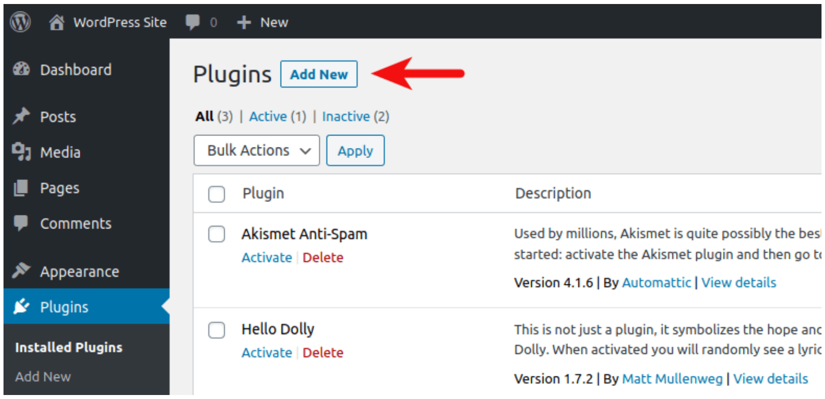
This address is used for admin purposes, like new user noti

How to Send Emails in WordPress

Your WordPress site needs to send emails like account registration emails, password-resetting emails, comment notification emails, etc. Instead of using expensive third-party solutions like Gsuite to create professional email addresses for your website, you can follow this iRedMail tutorial to set up your own mail server with your own domain name, so you can have unlimited mailboxes and send unlimited emails without breaking the bank.

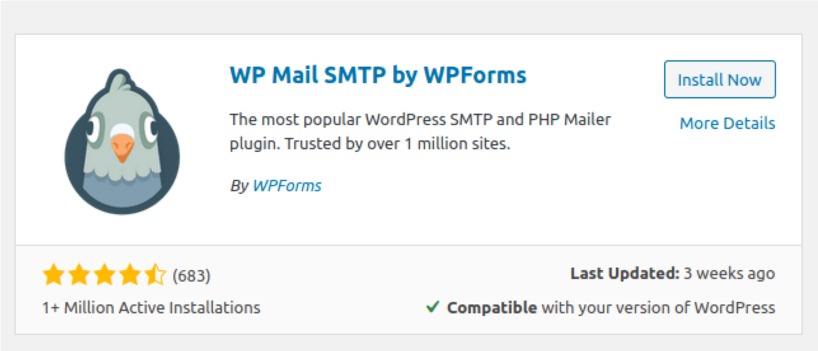
Note that it’s a good practice to install the mail server and WordPress on **two different virtual private servers** because you don’t want the mail server to slow down your WordPress site speed, and the mail server will leak the IP address of your WordPress site if they are on the same virtual private server, which means hackers can bypass any CDN (Content Delivery Network) you are using and launch DDoS attack directly at your origin server.

Once your mail server is up and running, you can install an SMTP plugin in WordPress, so it can connect to your mail server and send emails. Go to your **WordPress dashboard -> Plugins**, click **Add New** to install a new plugin.



□ Plugin	Description
□ Akismet Anti-Spam Activate | Delete	Used by millions, Akismet is quite possibly the bes started: activate the Akismet plugin and then go t Version 4.1.6 | By Automattic | View details
□ Hello Dolly Activate | Delete	This is not just a plugin, it symbolizes the hope anc Dolly. When activated you will randomly see a lyric Version 1.7.2 | By Matt Mullenweg | View details

Then type in **WP Mail SMTP** in the search box. Install and activate the **WP Mail SMTP by WPForms** plugin.



WP Mail SMTP by WPForms

The most popular WordPress SMTP and PHP Mailer plugin. Trusted by over 1 million sites.

By WPForms

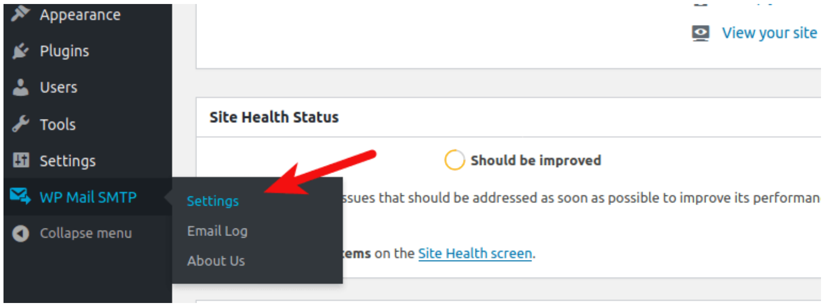
★★★★⯪ (683)

1+ Million Active Installations

Last Updated: 3 weeks ago

✓ **Compatible** with your version of WordPress

Reload the WordPress dashboard web page, you will see **WP Mail SMTP** on the left menu bar. Click on it and select **Settings**.



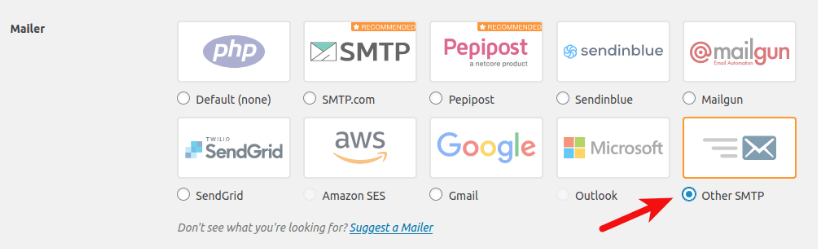
Site Health Status

Should be improved

ssues that should be addressed as soon as possible to improve its performan

ems on the Site Health screen.

Then scroll down to the Mailer section. By default, the PHP mailer is selected. We need to change it to **Other SMTP**.



Mailer

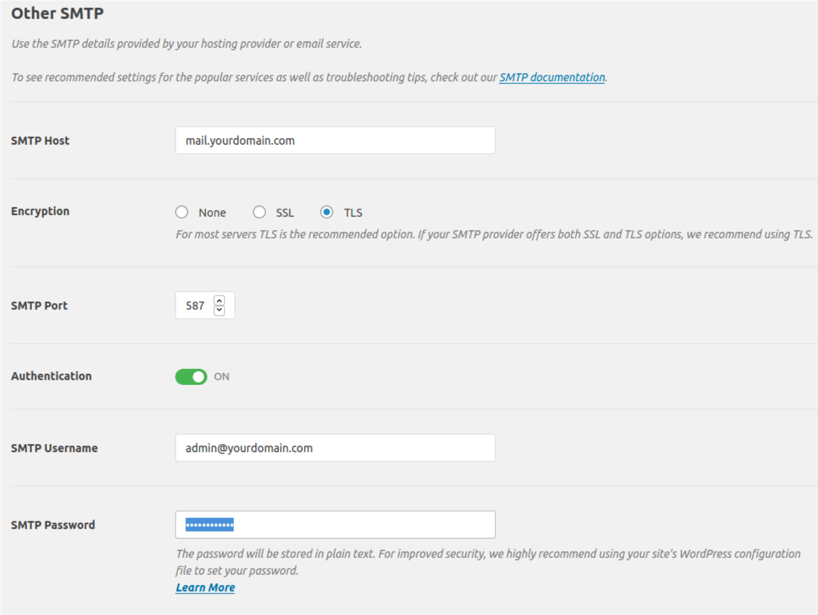
○ Default (none) ○ SMTP.com ○ Pepipost ○ Sendinblue ○ Mailgun

○ SendGrid ○ Amazon SES ○ Gmail ○ Outlook ● Other SMTP

Don't see what you're looking for? Suggest a Mailer

Scroll down and you will need to enter the SMTP settings.

- Enter the hostname of your mail server.
- Select TLS as Encryption.
- Use port 587.
- Enable Authentication.
- Enter an email address of your domain and the password.



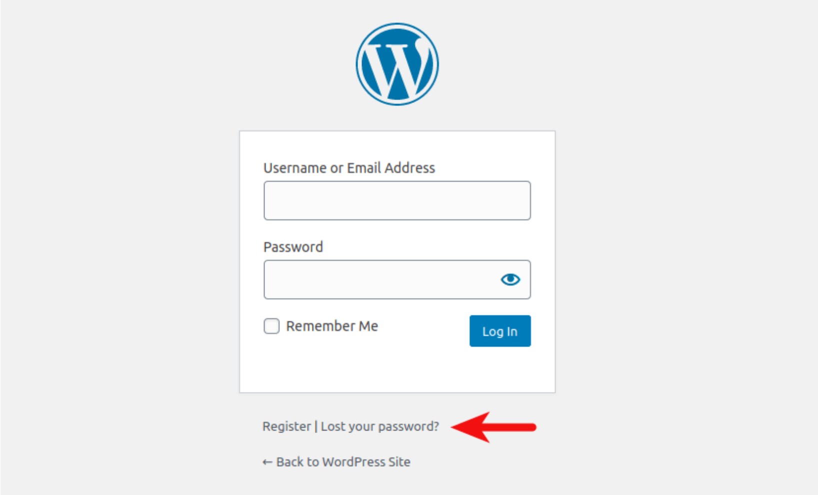
Other SMTP

Use the SMTP details provided by your hosting provider or email service.

To see recommended settings for the popular services as well as troubleshooting tips, check out our SMTP documentation.

SMTP Host	mail.yourdomain.com
Encryption	○ None ○ SSL ● TLS *For most servers TLS is the recommended option. If your SMTP provider offers both SSL and TLS options, we recommend using TLS.*
SMTP Port	587
Authentication	ON
SMTP Username	admin@yourdomain.com
SMTP Password	••••••••••••• *The password will be stored in plain text. For improved security, we highly recommend using your site's WordPress configuration file to set your password.* *Learn More*

After saving the settings, you can test email sending by logging out the WordPress dashboard, and click **lost your password** link to send a password-resetting email.



Increase Upload File Size Limit

By default, files such as images, PDF files uploaded to the WordPress media library can not be larger than 2MB. To increase the upload size limit, edit the PHP configuration file.

```
sudo nano /etc/php/8.1/fpm/php.ini
```

Find the following line (line 850). In the Nano text editor, you can press `Ctrl+/`, then enter `850`, to jump to line 850.

```
upload_max_filesize = 2M
```

Change the value like below:

```
upload_max_filesize = 20M
```

Then find the following line (line 694).

```
post_max_size = 8M
```

Change the maximum size of POST data that PHP will accept.

```
post_max_size = 20M
```

Save and close the file. Alternatively, you can run the following two commands to change the value without manually opening the file.

```
sudo sed -i 's/upload_max_filesize = 2M/upload_max_filesize = 20M/g' /etc/php/7.4/fpm/php.ini

sudo sed -i 's/post_max_size = 8M/post_max_size = 20M/g' /etc/php/7.4/fpm/php.ini
```

Then restart PHP-FPM.

```
sudo systemctl restart php8.1-fpm
```

Nginx also sets a limit of upload file size. The default maximum upload file size limit set by Nginx is 1MB. To allow uploading large files to your WordPress site, edit the Nginx configuration file.

```
sudo nano /etc/nginx/conf.d/example.com.conf
```

We have already set the maximum file size in this file, as indicated by

```
client_max_body_size 2M;
```

You can change it if you prefer, like 20M.

```
client_max_body_size 20M;
```

Save and close the file. Then reload Nginx for the changes to take effect.

```
sudo systemctl reload nginx
```

Next Steps

I hope this tutorial helped you install WordPress on Ubuntu 22.04 with Nginx, MariaDB and PHP8.1 (LEMP stack). As always, if you found this post useful, then subscribe to our free newsletter to get more tips and tricks.

Backup is important in case of hacking, data center disasters, etc. You should have a backup strategy for your WordPress site.

- Back Up and Restore MariaDB Databases From the Command line
- Use Duplicati to Back Up Files on Debian, Ubuntu, Linux Mint

Linux Server Performance Tuning and Monitoring

- Easily Boost Ubuntu Network Performance by Enabling TCP BBR
- What is HTTP/2 and How to Enable it on Nginx
- Linux Server Performance Monitoring with Netdata (2022)

WordPress Command Line Utility

- You may also want to use the WP-CLI command-line tool to manage your WordPress site.
- You can use WPScan to scan the vulnerabilities of your WordPress site.

Take care 🙂

Rate this tutorial

[Total: 5 Average: 4.6]

You may also like:

Install Next…

Install Word…

Install Next…

Install Proje…

Install Word…

Install Next…

How to Instal…

How to Instal…