# Setting up a WordPress site in AWS

## WordPress Site in AWS with EC2, RDS and S3

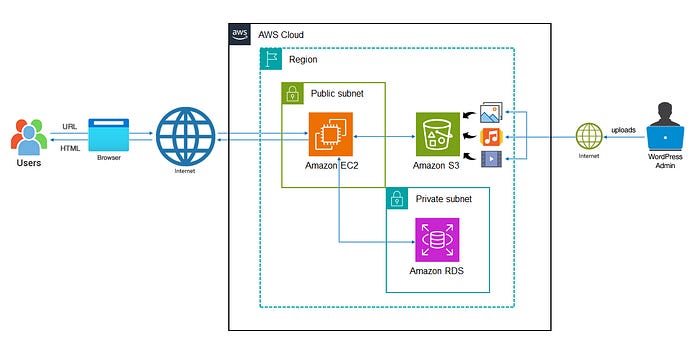
[[](https://medium.com/@rogernem?source=post_page-----84e0b34fd724--------------------------------)](https://medium.com/@rogernem?source=post_page-----84e0b34fd724--------------------------------)

[Roger Nem](https://medium.com/@rogernem?source=post_page-----84e0b34fd724--------------------------------)

12 min read

·

Jan 9



— Architecture Diagram of a WordPress site in AWS

# Table of Contents

1. [Introduction](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#9129)
2. [Quick Definitions](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#ad79)  
   - [WordPress](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#d716)  
   - [Content Management System (CMS)](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#b2d5)  
   - [Amazon Elastic Compute Cloud (Amazon EC2)](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#d36d)  
   - [Amazon Relational Database Service (Amazon RDS)](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#0861)  
   - [Amazon Simple Storage Service (Amazon S3)](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#c29c)
3. [Setting up a WordPress Site in AWS](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#d7c3)  
   - [Step 1: Create an RDS Instance](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#0b96)  
   - [Step 2: Create an EC2 Instance](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#7b10)  
   - [Step 3: Install WordPress on your EC2](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#08e2)  
   - [Step 4: Run the WordPress Install Script](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#0541)  
   - [Step 5: Upload WordPress Media to S3](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#9e8b)
4. [Conclusion](https://medium.com/@rogernem/hosting-a-wordpress-site-in-aws-84e0b34fd724#5c3e)

# Introduction

In this article you will learn how to setup a WordPress site in AWS using [Amazon Elastic Compute Cloud](https://aws.amazon.com/ec2/) (Amazon EC2), [Amazon Relational Database Service](https://aws.amazon.com/rds/) (Amazon RDS) and [Amazon Simple Storage Service](https://aws.amazon.com/s3/) (Amazon S3).

[Amazon Web Services (AWS)](https://aws.amazon.com/free/) stands out as a premier cloud computing platform, providing a diverse array of services and resources encompassing computing, storage, networking, security, and beyond.

AWS extends a free tier catering to beginners, granting complimentary access to specific services for a limited duration.

# Quick Definitions

## WordPress

At its essence, WordPress stands as the **preeminent, uncomplicated solution for the creation of personal websites or blogs**. Remarkably, it commands a commanding presence, fueling over 43.1% of the total websites existing on the Internet as of December 2023.

Indeed, WordPress serves as the cornerstone for more than one in four websites, solidifying its status as the most universally embraced [Content Management System (CMS)](https://medium.com/@rogernem/creating-a-wordpress-site-in-aws-fa14ba298cf2#e84d) on a global scale.

Notably versatile, WordPress extends its capabilities to support diverse applications, ranging from e-commerce platforms to interactive message boards and beyond.

On a slightly more technical level, WordPress emerges as a user-friendly, open-source CMS governed by the GPLv2 license, empowering individuals to freely utilize or modify the WordPress software.

## Content Management System (CMS)

In simple terms, a Content Management System (CMS) is a software application that allows individuals or teams to create, manage, and organize digital content on a website without requiring advanced technical skills.

It simplifies the process of publishing, editing, and updating content, making it easier for users to maintain a website without delving into the complexities of coding or web development. With a CMS, users can add and modify text, images, and other elements, facilitating the efficient management of online content.

## Amazon Elastic Compute Cloud (Amazon EC2)

Amazon EC2 is a web service provided by Amazon Web Services (AWS) that allows users to rent virtual computers, known as instances, in the cloud.

These virtual machines can be configured and customized to run various applications and tasks, providing scalable computing capacity on-demand.

Users can choose the type of instance, the operating system, and other specifications to meet their specific needs, and they pay for the computing resources they use without needing to invest in physical hardware.

EC2 is widely used for hosting websites, running applications, and handling various computational workloads in a flexible and cost-effective manner.

## Amazon Relational Database Service (Amazon RDS)

Amazon RDS is a cloud-based service provided by Amazon Web Services (AWS) that simplifies the setup, operation, and scaling of relational databases, allowing users to easily deploy and manage databases like MySQL, PostgreSQL, Oracle, SQL Server, and others without dealing with the complexities of hardware provisioning, database installation, or maintenance tasks.

With Amazon RDS, users can focus on their applications rather than the underlying database infrastructure. The service handles tasks such as backups, software patching, and database scaling automatically.

It provides a scalable and efficient solution for storing and retrieving data in a structured manner, making it particularly useful for applications that rely on relational databases for data management.

Users pay for the resources they consume, and Amazon RDS offers a managed environment for relational databases in the cloud.

## Amazon Simple Storage Service (Amazon S3)

Amazon S3 is a cloud-based storage service provided by Amazon Web Services (AWS). In technical terms, it’s a scalable and secure platform that allows users to store and retrieve data over the internet.

Users can organize their data into containers called “buckets” and access them through a web interface or application programming interface (API).

Amazon S3 is designed for high durability and availability, making it a reliable choice for various purposes like data backup, hosting static websites, and supporting scalable applications.

Users pay for the storage space they use, making it a flexible and cost-efficient storage solution in the cloud.

You can read more at <https://medium.com/@rogernem/amazon-simple-storage-service-eb0808cf30d5>

**[Amazon Simple Storage Service (Amazon S3)](https://medium.com/@rogernem/amazon-simple-storage-service-eb0808cf30d5?source=post_page-----84e0b34fd724--------------------------------" \t "_blank)**

**[AWS Object Storage Service](https://medium.com/@rogernem/amazon-simple-storage-service-eb0808cf30d5?source=post_page-----84e0b34fd724--------------------------------" \t "_blank)**

[medium.com](https://medium.com/@rogernem/amazon-simple-storage-service-eb0808cf30d5?source=post_page-----84e0b34fd724--------------------------------" \t "_blank)

# Setting up a WordPress Site in AWS

Our WordPress application will be installed and hosted in an EC2 instance and we will use RDS to store our WordPress data (e.g. settings, posts, pages, comments, etc.) and S3 to store and serve our WordPress media files (e.g. images, videos, audio, etc.).

To follow along this guide, please make sure to setup your **AWS account first** (You can sign up [here](https://aws.amazon.com/) and follow this [tutorial](https://aws.amazon.com/premiumsupport/knowledge-center/create-and-activate-aws-account/) to set it up).

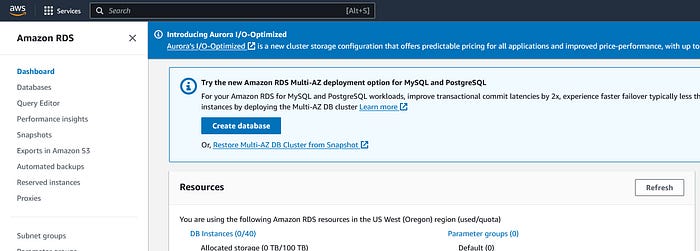
## Step 1: Create an RDS Instance

WordPress depends on a MySQL database for data storage, and therefore we will create an RDS instance specifically for the MySQL database as our first step.

1. Sign in to the AWS Management Console and open the Amazon RDS console at <https://console.aws.amazon.com/rds/>.
2. Select a **Region**. (e.g. ***US East (N. Virginia) us-east-1***)

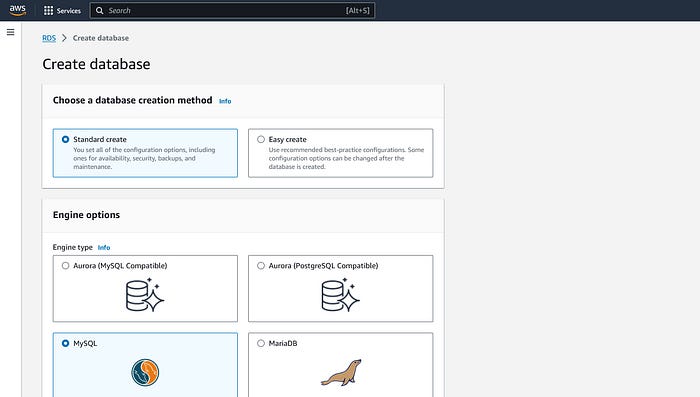


3. Click on “**Create Database”.**



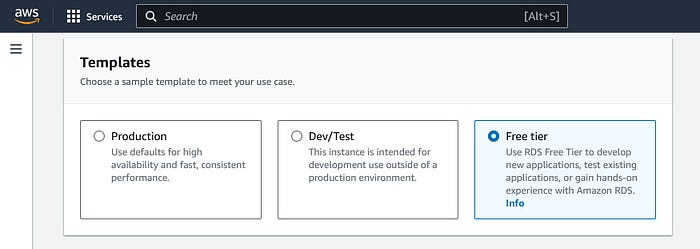
— Amazon RDS Console

4. On the next screen, under “**Choose a database creation method**” select “***Standard Create*”** and under “**Engine options**” select “***MySQL***”.



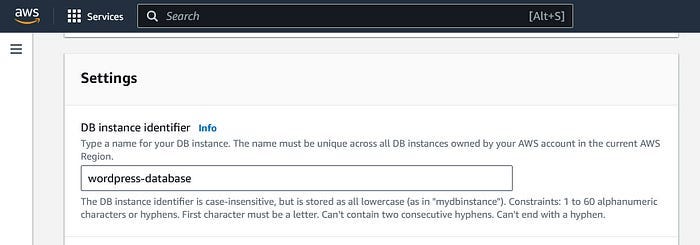
— Amazon RDS | Create Database

5. Scroll down until “**Templates**” and select “***Free tier***”.



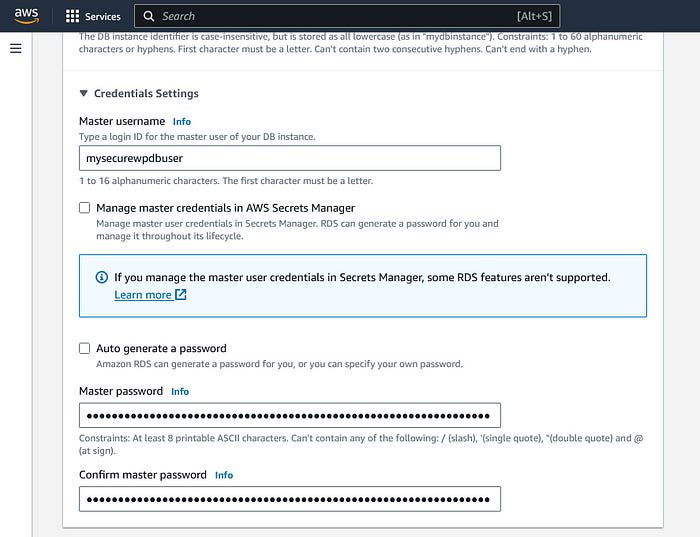
— Amazon RDS | Create Database | Templates

6. Scroll down until “**Settings**” and choose your database instance name (e.g. wordpress-database).



— Amazon RDS | Create Database | Settings

7. Under “**Credentials Settings**” enter a strong username and a password for your database.



— Amazon RDS | Create Database | DB Credentials

- The Master username **must be 16 characters or less**.  
- Remember these credentials as you will need them to connect to your database.

8. Under “**Connectivity**”, select the desired VPC and subnet group or leave the **defaults**.

9. Under “**VPC security group (firewall)**”, choose one or more VPC security groups to allow access to your database. You can leave the **default** if you wish to.

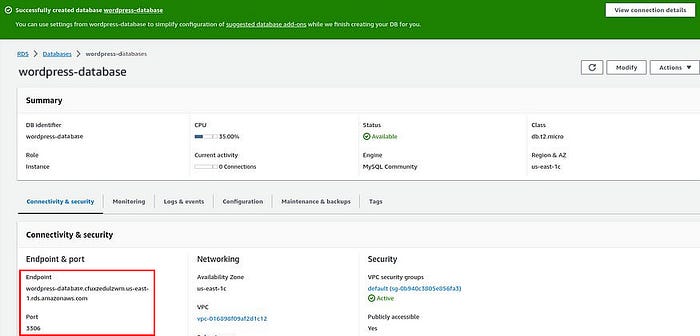
Make sure that the security group rules allow the appropriate incoming traffic.

⚠️Before proceeding, make sure you agree with the “**Estimated Monthly costs**”.

10. Leave all the other settings as **default** and click on “**Create Database”** at the bottom of the page.

It may take a few minutes for your database instance to be created and **when ready, write down its endpoint and port** (shown below).

**Endpoint**: wordpress-database.cfuxzedulzwrn.us-east-1.rds.amazonaws.com   
**Port:** 3306

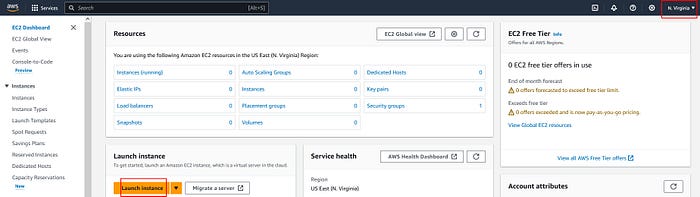


— Amazon RDS | Database Created

## Step 2: Create an EC2 Instance

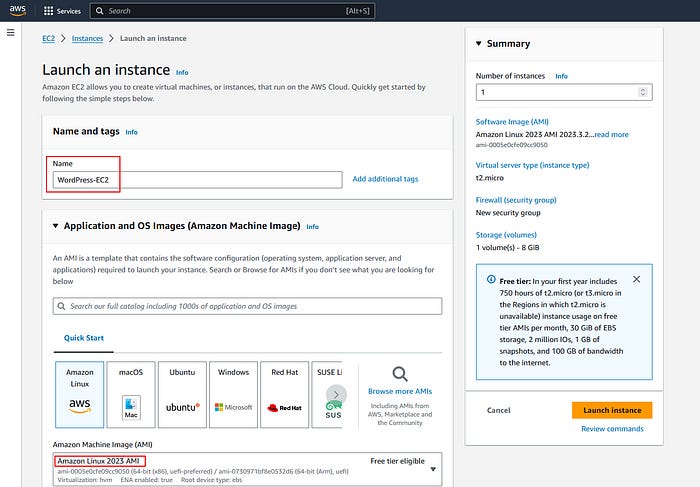
1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. Select the **Region**.
3. Click on “**Launch instance**”.

It must be the same region as you have created your RDS Database instance earlier — **US East (N. Virginia) us-east-1**.



— Amazon EC2 | Dashboard

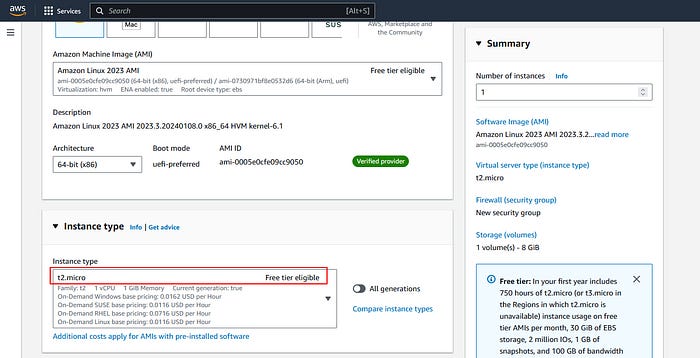
4. Under “**Name and tags**”, give a name to your instance (e.g. WordPress-EC2) to help identify it later and under “**Amazon Machine Image (AMI)**” make sure “***Amazon Linux 2023 AMI***” is selected (Free tier eligible).



— Amazon EC2 | Launch an Instance

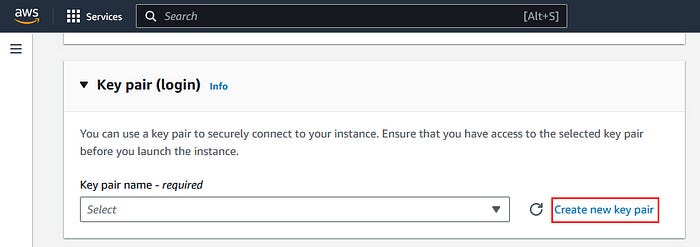
6. Under “**Instance Type**”, select “**t2.micro”** which is the smallest and cheapest instance type eligible for the free tier.

You may change the instance type later in case you need more resources.



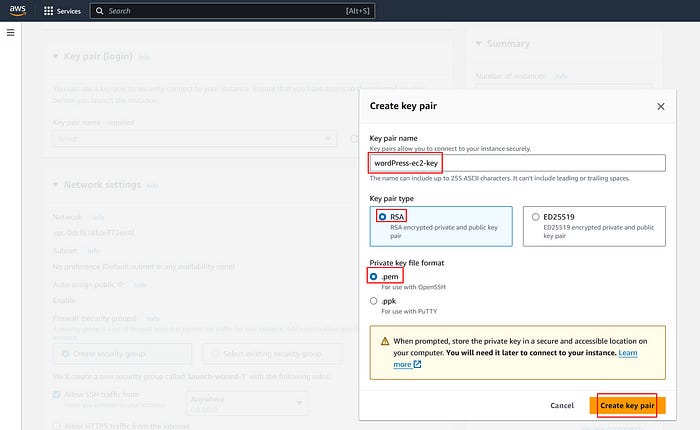
— Amazon EC2 | Instance Type

7. Under “**Key pair (login)**” click on “**Create new key pair**”.



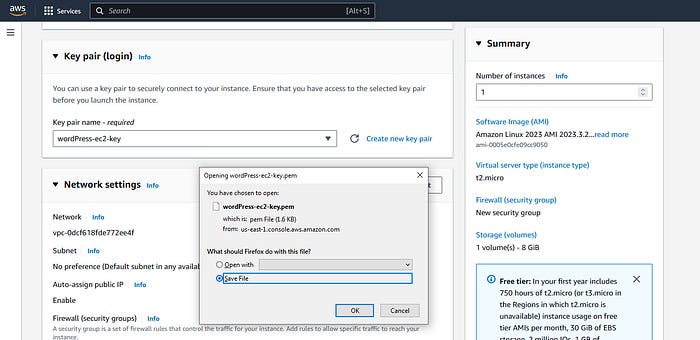
— Amazon EC2 | Key Pair (Login)

8. Enter a key pair name (e.g. wordPress-ec2-key), select “**RSA**” for the key pair type and “**.pem**” for private key format and click on “**Create key pair**”.



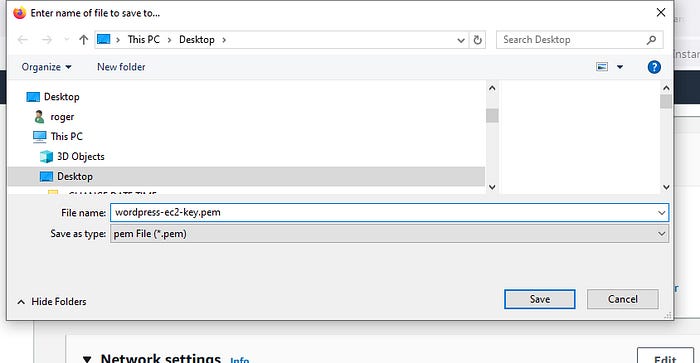
— Amazon EC2 | Key Pair Details

9. Click on “**Save File**” and “**OK**”.



— Amazon EC2 | Key Pair File

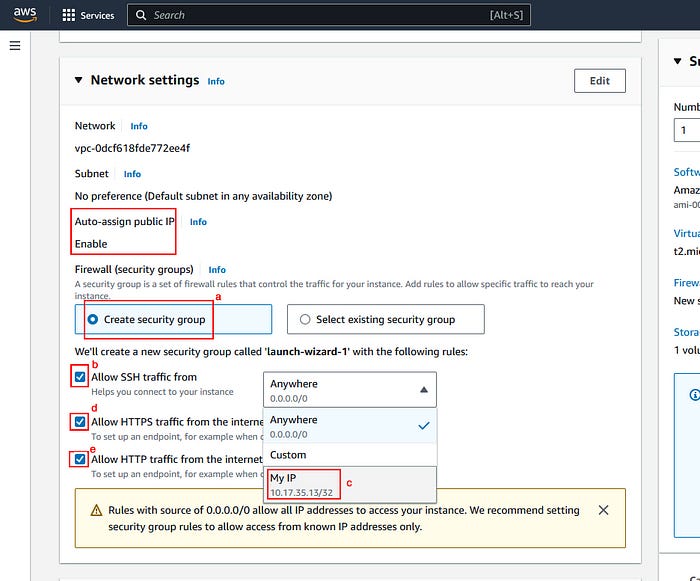
10. Select a location on your computer to save your file (e.g. Desktop).



⛔ **\*\* Do not lose this file as you will not be able to access your instance without it. And for security do not share it with anyone. \*\***

11. Under “**Network settings**”, as shown below:

* Make sure “**Auto-assign public IP**” is “**Enable**”.
* Select “**Create security group**” (a).
* Check “**Allow SSH traffic from**” (b) and change it from “Anywhere” to “**My IP**” (c).
* Select “**Allow HTTPS traffic from the internet**” (d) and “**Allow HTTP traffic from the internet**” (e).



- The **SSH rule** allows you to connect to your EC2 instance remotely via a secure shell (SSH) protocol and the **HTTP and HTTPS** (encrypted protocol) **rules** allow anyone to browse your blog.

12. Click on “**Launch Instance”**

The launch of your EC2 instance may require a few minutes. You can monitor its status on the EC2 console. Once the instance is operational, you can access its details, including the public IP address, private IP address, security group, and key pair.

**Take note of your instance’s public IP address**, as it will be necessary for future connections to your instance and accessing your WordPress site.

The **public IP** **address** is the one visible to the internet and may undergo changes if you stop and restart your EC2 instance. Alternatively, you have the option to assign an **elastic IP address** to your instance, providing a static IP address that remains constant.

❗ Starting on February 1, 2024, AWS will charge for all public IPv4 addresses, including public IPv4 addresses associated with running instances and Elastic IP addresses. For more information, see the **Public IPv4 Address** tab on the [Amazon VPC pricing page](http://aws.amazon.com/vpc/pricing/).

## Step 3: Install WordPress on your EC2

The next step is to install and configure WordPress on your [EC2 instance created earlier](https://medium.com/@rogernem/creating-a-wordpress-site-in-aws-fa14ba298cf2##987a).

1. Connect to your EC2 instance using SSH. You can use the following command to connect to your instance from a terminal window, where wordPress-ec2-key.pem is the name of your key pair file and ec2-user@public-ip-address is the user name and the public IP address of your EC2 instance.

ssh -i wordPress-ec2-key.pem ec2-user@public-ip-address

* You may see a warning message about the authenticity of the host. Type yes to continue connecting. You may also be prompted to enter the passphrase for your key pair file in case you have set one.

The authenticity of host 'ec2-10-20-30-11.compute-1.amazonaws.com (10-20-30-11)' can't be established.  
ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.  
Are you sure you want to continue connecting (yes/no/[fingerprint])?

More information can be found at <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/connect.html>

Once connected, you should see a welcome message and a command prompt:

[ec2-user@ip-10-20-30-11 ~]$

2. Update the packages on your EC2 instance by running the following command:

sudo yum update -y

3. Reboot your EC2 instance when done.

4. After rebooting your EC2 instance, install the Apache web server, the PHP software, and the MySQL client by running the following command:

sudo yum install -y httpd php mysql

5. Start the Apache web server and enable it to start automatically on boot by running the following commands:

sudo systemctl start httpd  
sudo systemctl enable httpd

6. Download the latest WordPress installation package with the **wget** command. The following command should always download the latest release.

wget https://wordpress.org/latest.tar.gz

7. Unzip and unarchive the installation package. The installation folder is unzipped to a folder called wordpress.

tar -xzf latest.tar.gz

8. Move the WordPress files to the Apache web server document root directory by running the following command:

sudo mv wordpress/\* /var/www/html/

9. Change the ownership and permissions of the WordPress files and directories by running the following commands:

sudo chown -R apache:apache /var/www/html/  
sudo find /var/www/html/ -type d -exec chmod 755 {} \;  
sudo find /var/www/html/ -type f -exec chmod 644 {} \;

10. Create a WordPress configuration file by copying the sample file by running the following command:

cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php

11. Edit the WordPress configuration file by using a text editor of your choice, such as nano or vi:

sudo nano /var/www/html/wp-config.php

* In the WordPress configuration file, find the following lines and replace the values with the credentials of your [RDS database instance that you created earlier](https://medium.com/@rogernem/creating-a-wordpress-site-in-aws-fa14ba298cf2#32f8). Remember to use the endpoint and the port of your database instance as the host name, and the username and the password that you specified when creating the RDS database instance.

/\*\* 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' );  
/\*\* MySQL hostname \*/  
define( 'DB\_HOST', 'localhost' );

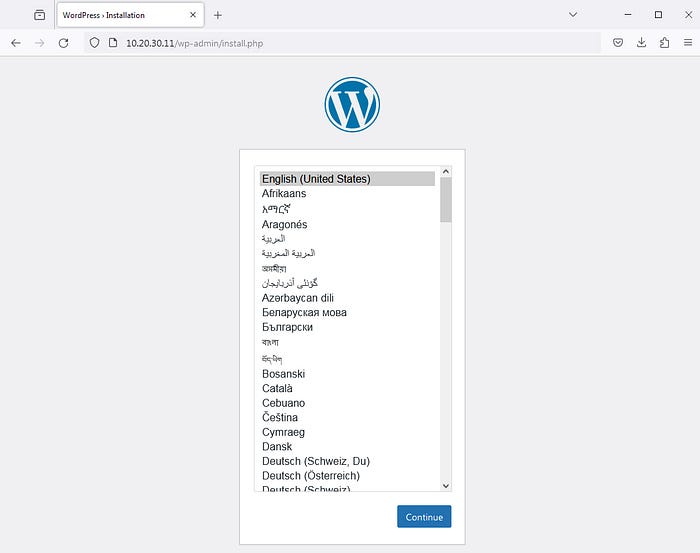
In my example, my database name is wordpress-database, my username is mysecurewpdbuser, my password is MyverYsecreTpassWord!, and my endpoint and port are wordpress-database.cfuxzedulzwrn.us-east-1.rds.amazonaws.com:3306:

/\*\* The name of the database for WordPress \*/  
define( 'DB\_NAME', 'wordpress-database' );  
/\*\* MySQL database username \*/  
define( 'DB\_USER', 'mysecurewpdbuser' );  
/\*\* MySQL database password \*/  
define( 'DB\_PASSWORD', 'MyverYsecreTpassWord!' );  
/\*\* MySQL hostname \*/  
define( 'DB\_HOST', 'wordpress-database.cfuxzedulzwrn.us-east-1.rds.amazonaws.com' );

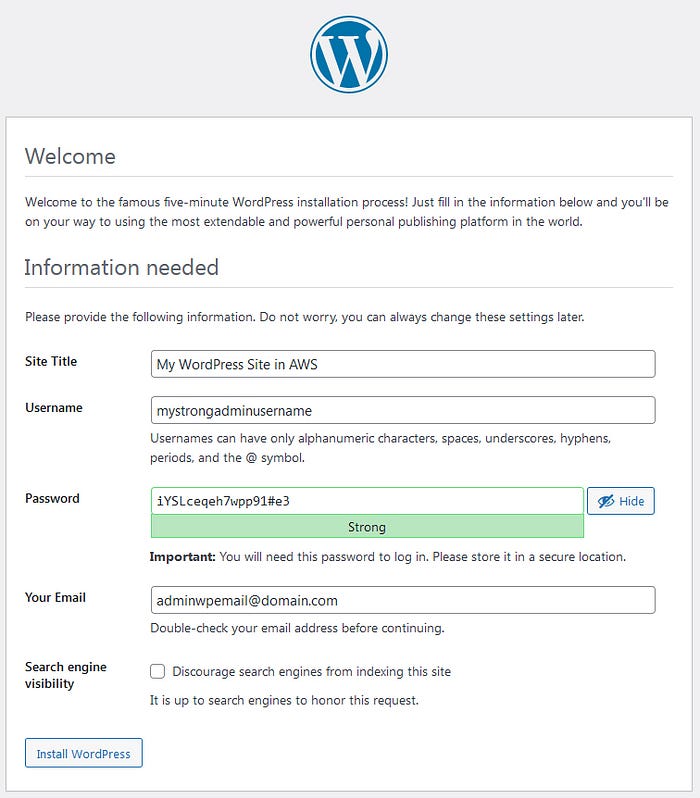
* Save the WordPress configuration file and exit (Ctrl+x).

## Step 4: Run the WordPress Install Script

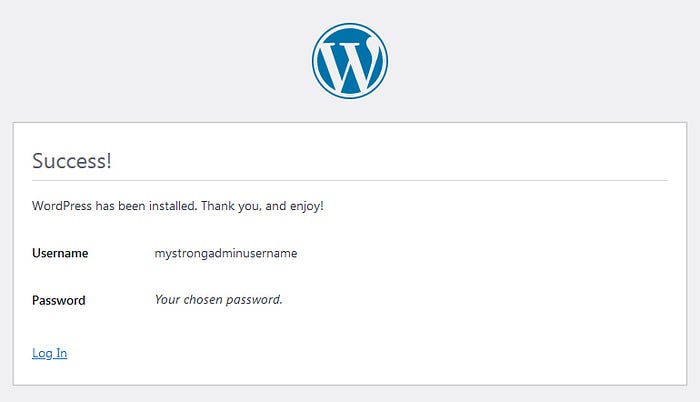
1. Browse <http://YOUR-EC2-PUBLIC-IP/wp-admin/install.php> and click on “**Continue**”.



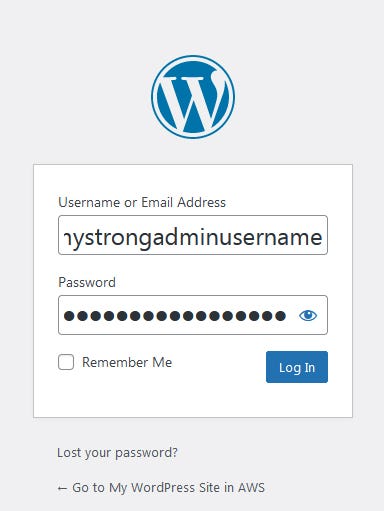
2. Enter your WordPress **Site Title**, the **Username** and **Password** to administer your WordPress site, the administrator **Email** address, and click on “Install WordPress” when finished.



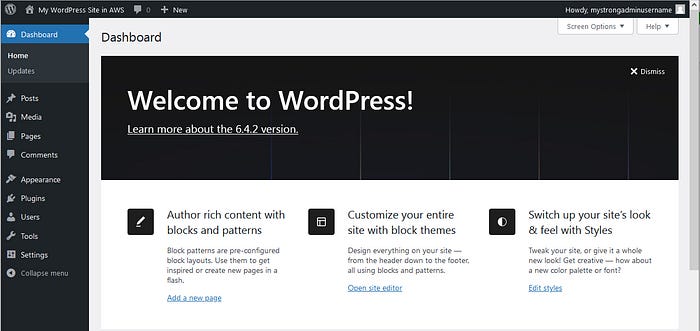
If everything goes well, you should see a success message like this:



3. Click on “**Log In**” and on the next screen, inform the username and password you chose earlier during the installation.



4. Click on “**Log In**” one more time and you should see the WordPress dashboard as follows.

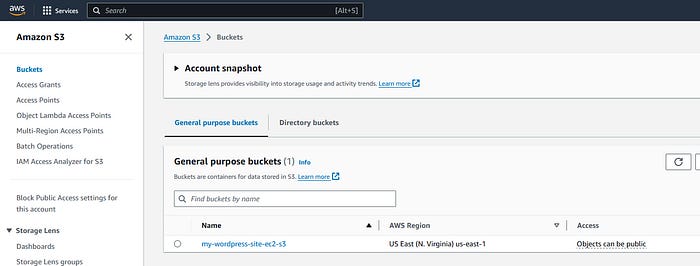


— WordPress Dashboard

Congratulations! You have successfully installed and configured WordPress on an EC2 instance.

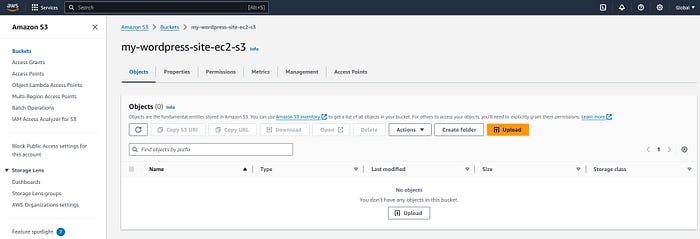
## Step 5: Upload WordPress Media to S3

You can refer to [my article](https://medium.com/@rogernem/hosting-my-static-website-in-an-aws-s3-bucket-d5e1d94417f4#360b) to create an S3 bucket that will be used to store and serve our WordPress media files (e.g. my-wordpress-site-ec2-s3). This S3 bucket can also be used to back up your WordPress database, for example.



— Amazon S3 Bucket

* Upload the desired media files to your bucket by either dragging and dropping them from your computer to the S3 console, or by clicking on “**Upload”** and choosing the desired media files you want to upload.

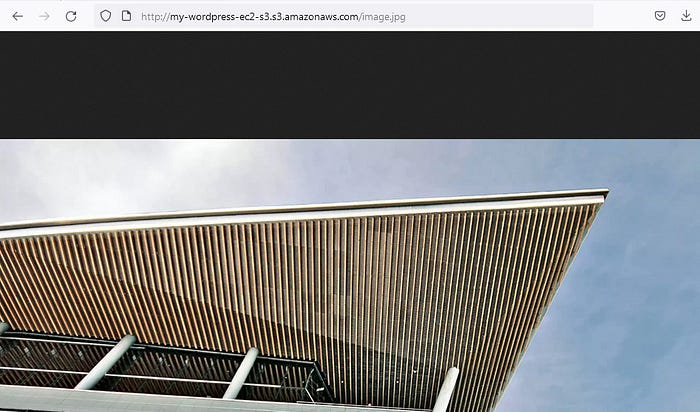


* To access your files from the web, you need to **use the URL of your bucket and your object**.

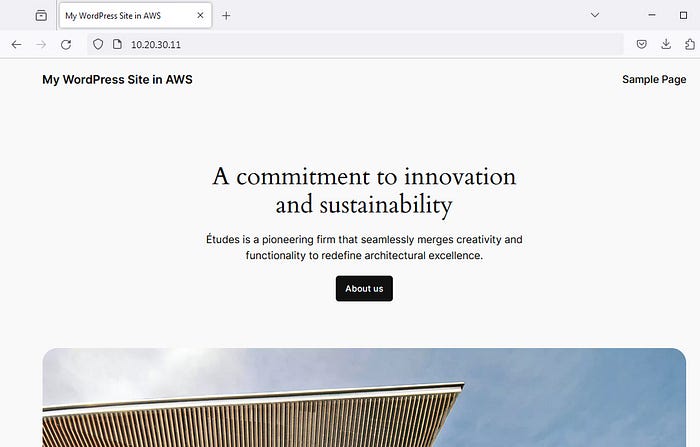
The URL of your bucket is http://bucket-name.s3.amazonaws.com/, where bucket-name is the name of your bucket and the URL of your object is http://bucket-name.s3.amazonaws.com/object-key, where object-key is the name of your object.

Because my bucket name is my-wordpress-site-ec2-s3 and my object name is image.jpg, the URL of my object is <http://my-wordpress-ec2-s3.s3.amazonaws.com/image.jpg.>

To test our media files we can browse their corresponding URLs and we should be able to see them displayed.



You can now customize your WordPress site by adding themes, plugins, posts, pages, and more. The end-result should be something like the following:



— My WordPress Site in AWS

# Conclusion:

By following the steps outlined in this article, you’ve successfully constructed a cost-effective WordPress site within an EC2 instance, utilizing Amazon RDS for data storage and Amazon S3 for media storage.

**If you enjoyed this article and found it helpful, please don’t forget to leave a heart** ❤**, comment** 💬**, clap** 👏🏻**, and share** ➦ **it to show your support.**

**Also, don’t forget to** [**connect**](https://www.linkedin.com/in/rogertn/) **and** [**follow me**](https://medium.com/@rogernem) **for more articles. Thank you!**

References:  
- <https://wordpress.com/>  
- <https://en.wikipedia.org/wiki/WordPress>  
- <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.MySQL.html>  
- <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html>  
- <https://docs.aws.amazon.com/efs/latest/ug/gs-step-one-create-ec2-resources.html>  
- <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/hosting-wordpress.html>

[WordPress](https://medium.com/tag/wordpress?source=post_page-----84e0b34fd724---------------wordpress-----------------)

[Aws In Plain English](https://medium.com/tag/aws-in-plain-english?source=post_page-----84e0b34fd724---------------aws_in_plain_english-----------------)

[Ec2](https://medium.com/tag/ec2?source=post_page-----84e0b34fd724---------------ec2-----------------)

[S3](https://medium.com/tag/s3?source=post_page-----84e0b34fd724---------------s3-----------------)

[Rds](https://medium.com/tag/rds?source=post_page-----84e0b34fd724---------------rds-----------------)