John Dukewich
Dr. Jayme Peacock
3 April 2020
Audience: Anybody comfortable with computers can follow this guide, but there are added notes/explanations to explain the technical aspect of some steps if the reader wants a better understanding of what is being done

How to Deploy a WordPress Website with Amazon Web Services (AWS)

Overview

If you want to have your own personal website, then the website's files must live on a computer or server somewhere where anybody on the Internet can access them. Many companies will host common sites like Wordpress for you, but if you want to have the most control over your Wordpress site, Amazon Web Services (AWS) provides services to allow you to deploy your own website that can be reached over the Internet. This process is divided into three parts: setting up the server on AWS, installing the proper software and files onto the server, and configuring your Wordpress settings. Some of these steps may have technical aspects, but the instructions are presented so that anybody should be able to follow them even if they are unfamiliar with some of the technical details. At the end of this guide, you will have a running Wordpress website that can be accessed by visiting an IP address in your Internet browser of choice.

Required Materials

- An AWS account (can be following AWS directions at https://portal.aws.amazon.com/billing/signup)
- A computer with Internet access

⚠CAUTION: AWS Services Cost Money

This guide utilizes the Free Tier of AWS, but the Free Tier lasts for a year after account creation. If you were to leave these services running, you may be charged to the billing information of your account. Be sure to stop and terminate any AWS resources created in this guide if you no longer want them to run.

Part 1 (of 3): Setting up the AWS Server

1. Login to the AWS management console by going to https://console.aws.amazon.com and entering your account user email address and password.

2. Click "Services" in the navigation bar at the top of the screen, then click "EC2" (see Figure 1 for where these links are located). The AWS EC2 service will be used to create a server where the WordPress files will be installed and served from.

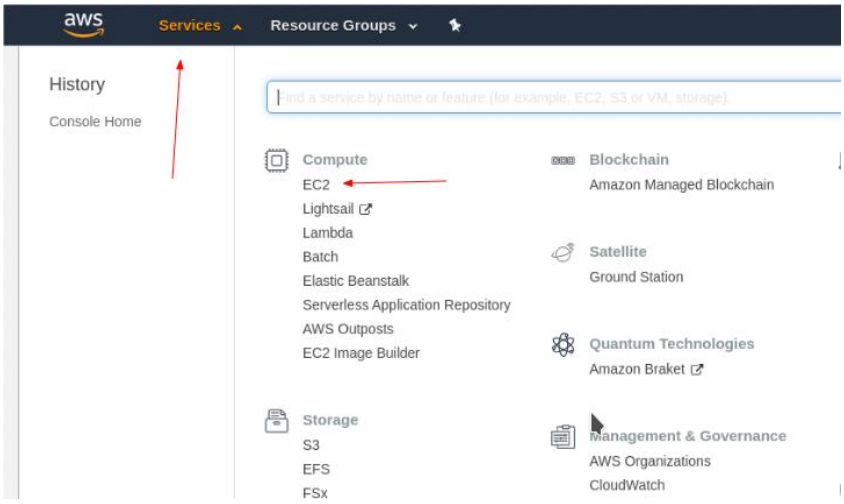

Figure 1: AWS Console – the EC2 service can be found by navigating to the Services tab in the navigation bar

3. Click “Launch instance” button near the middle of your screen as seen in Figure 2.

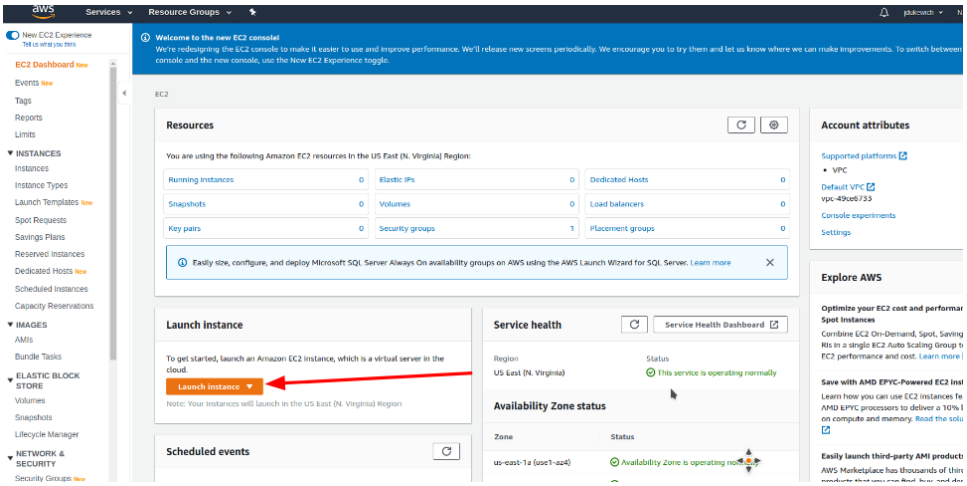

Figure 2: EC2 Homepage – a new EC2 server can be launched directly from this page

4. Click the “Select” button for the “Ubuntu Server 18.04” image (see Figure 3). Any image and operating system will work here, but this guide will use commands that work with Ubuntu 18.04. Commands may vary for other operating systems.

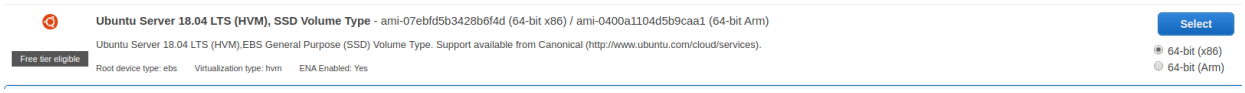

Figure 3: Ubuntu Server Image – this guide uses the Ubuntu 18.04 operating system

5. Ensure “t2.micro” is selected as the instance type as in Figure 4, and click “Review and Launch” at the bottom of your screen. AWS EC2 servers can come in various hardware specifications, with more hardware being more expensive. The “t2.micro” instance type falls under the AWS Free Tier, meaning it won’t cost money to run for the first year after creating your AWS account.

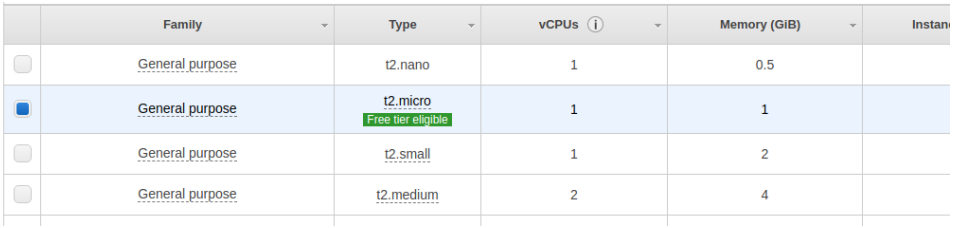

Figure 4: EC2 Instance Type – select the t2.micro instance type so that your account will not be billed under the Free Tier

6. You will be taken to a screen to review details about your EC2 server instance. Under the "Security Groups" section, click "Edit security groups" (see Figure 6). Security groups are firewall rules that can control what IP addresses can access your instance over which protocol and port number.

Figure 5: EC2 Server Details – the security groups on the server can be reviewed and edited here

7. Two rules must be added so that your WordPress site can be accessed over the Internet. Click "Add Rule", choose "HTTP" for "Type." This rule will allow visitors to access your site via the commonly used HTTP protocol. Once again, click "Add Rule", and choose "HTTPS" for "Type" this time. This will allow visitors to your site over HTTPS. On a more detailed note, these two rules allow for access to your EC2 server over TCP port 80 and 443, which are how Internet browsers connect to websites. Figure 6 shows how the final security groups should look.

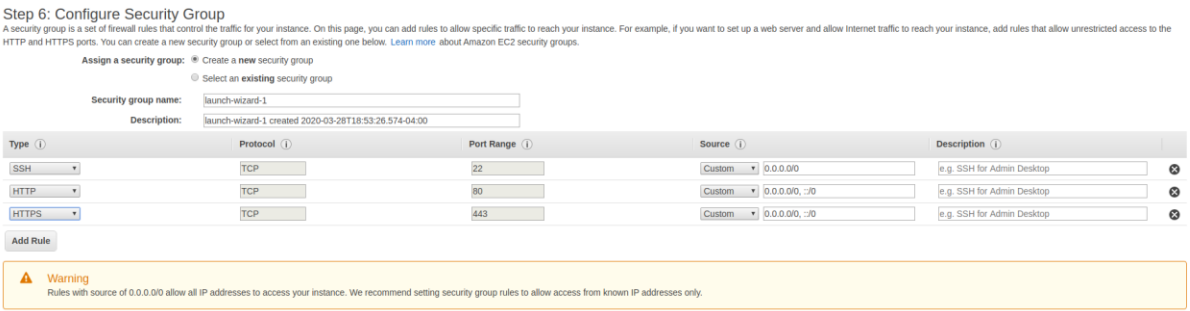

Figure 6: Security Group Rules – two rules should be added for HTTP and HTTPS to allow for your server to be accessed over the Internet

8. Click "Review and Launch" then "Launch" in the bottom right of your screen.

9. A popup will appear like the one in Figure 7. Select "Create a new keypair" from the dropdown. Type a name for your SSH key that will be used to log into this EC2 instance. Click "Download Key Pair" and take note of the location where you save the key.

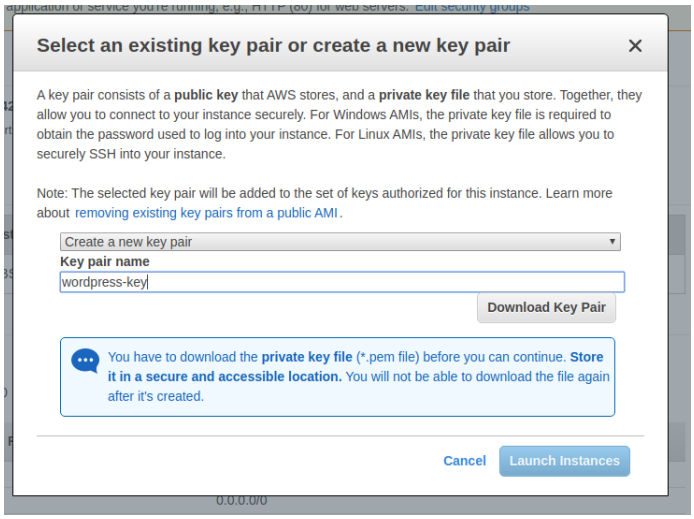

Figure 7: SSH Keys – an SSH key pair should be created and downloaded to access your server remotely

10. Click “Launch Instances” to start the EC2 server, and then click “View Instances” on the following screen.

11. Wait for instance to be setup. You will know the instance is setup and ready when the “Instance State” column is set to “running” with a green status circle like in Figure 8. Note the IP address of your instance under the “IPv4 Public IP” column (ex. 174.129.151.196).

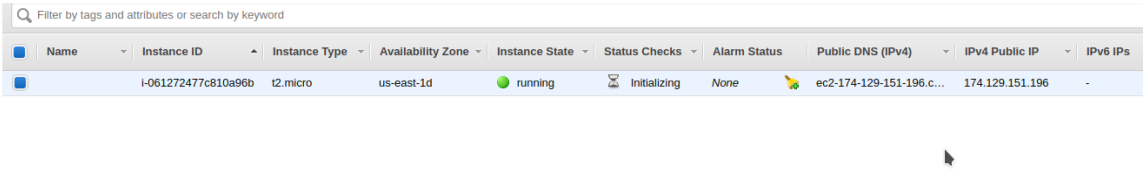

Figure 8: EC2 Instances – the EC2 page shows the status of your server as well as information about it

Part 2 (of 3): Installing Required Software and Files for WordPress

1. Open a terminal from your applications if using Mac or Linux or PowerShell if using Windows and type

   `ssh -i </path/to/your/key> ubuntu@<ip>`

   where </path/to/your/key> is the download location of the SSH key in Part 1 Step 9 (ex. /home/username/Downloads/wordpress-key.cer) and <ip> is the IP address of your EC2 server in Part 1 Step 11 (ex. 174.129.151.196). This command will connect you to your EC2 server remotely so that you can install the necessary software and files to get WordPress running.

2. Enter the command

   `sudo -i`

   to become a root user. Becoming the root user gives you permissions to install software and files on the server.

3. Enter the command

   ```
   apt update
   ```

   to update your server. This will allow you to download the most updated versions of the software needed to run WordPress.

4. Install the Apache web server by entering the command

   ```
   apt install -y apache2
   ```

   Apache is responsible for handling web requests to your server and returning the proper files.

5. Install the database software MySQL by running the command

   ```
   apt install -y mysql-server
   ```

   MySQL is a database engine that will store information regarding your WordPress site, such as users and plugins.

6. Configure MySQL by entering the command

   ```
   mysql_secure_installation --use-default
   ```

   This command will configure MySQL so that it will be ready to use for WordPress.

7. Now a database must be created for WordPress. This guide will create a database named “WordPress” but you can use any name you want. Be sure to keep track of what name you use for this. Create the database by entering the command

   ```
   mysql -u root -e "CREATE DATABASE wordpress;"
   ```

8. ⚠**CAUTION: Do not use a weak password**

   This guide uses the password of “password” for the database user, but this is not secure, so you should substitute with a more secure password.

   Now a user must be created to access the database. This guide will create a user named “wp_user” but you can use any name you want. The password in this guide is simply “password” but DO NOT use “password” as your password because this is extremely insecure and could lead to security vulnerabilities in your website. Keep track of what username and password you use in this step. Create a user for the database by entering the command

   ```
   mysql -u root -e "CREATE USER 'wp_user'@'localhost' IDENTIFIED BY 'password';"
   ```

9. Using the database name from Part 2 Step 7 and username from Part 2 Step 8 ("WordPress" and "wp_user" respectively in this guide), grant the user privileges on the database by entering the command

```
mysql –u root –e "GRANT ALL PRIVILEGES ON wordpress.* TO 'wp_user'@'localhost';"
```

this command will allow the user to perform the necessary database operations, such as creating and editing tables.

10. Install PHP and its dependencies by entering the command

```
apt install –y php libapache2-mod-php php-mysql
```

PHP is the programming language that WordPress is written with and will be needed for the site to work.

11. Download the files to run WordPress from the WordPress website by entering the command

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

12. Extract the website files from the compressed file downloaded in the last step with the command

```
tar –xvf latest.tar.gz
```

13. Now you should move the WordPress website files to a folder where the Apache web server will serve files from. To do this, run the command

```
mv wordpress/* /var/www/html
```

14. Remove the default file that Apache serves in order for your WordPress site to display properly by running the command

```
rm /var/www/html/index.html
```

15. For Apache to serve WordPress files, the owner of the files must be changed. You can allow Apache to serve the WordPress files by running the command

```
chown –R www-data: /var/www/html
```

This command changes the WordPress files to be owned by the server user "www-data" which is the user that the Apache process runs as.

16. Restart the Apache process so that it can begin to serve the correct WordPress files by running the command

```
service apache2 restart
```

PART 3 (of 3): Setting Up the WordPress Site

1. Go to http://<ip-address>/ in a browser, where <ip-address> is the IPv4 address from Part 1 Step 11 (ex. 174.129.151.196).

2. You should be brought to a webpage that looks like Figure 9. Select “English (United States)” (other languages can be chosen if desired, but this guide will use US English) and click “Continue”, then click “Let’s Go” on the next page.

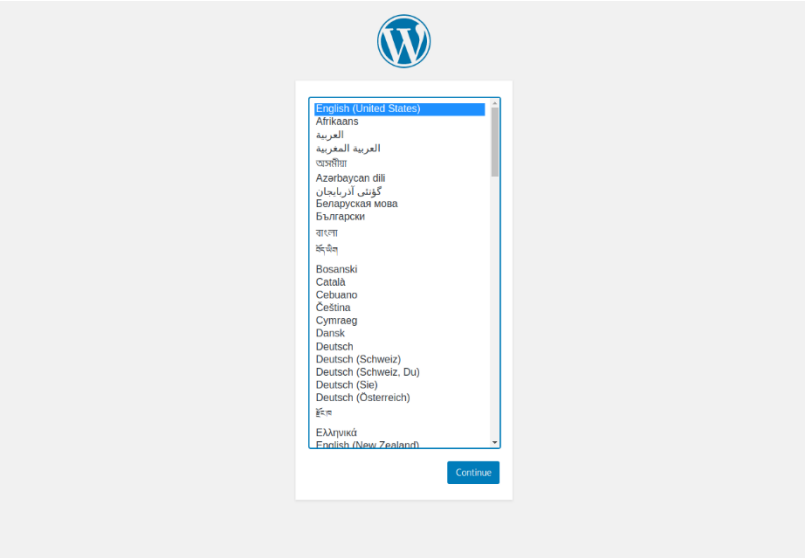

Figure 9: WordPress Installation – these pages will take you through the installation of WordPress

3. You are brought to a page like Figure 10. Provide WordPress with the database credentials created in Part 2. Enter the database name from Part 2 Step 7, username from Part 2 Step 8, and password from Part 2 Step 8. Leave the “Database Host” as “localhost” and leave “Table Prefix” as “wp_”, then click “Submit.”

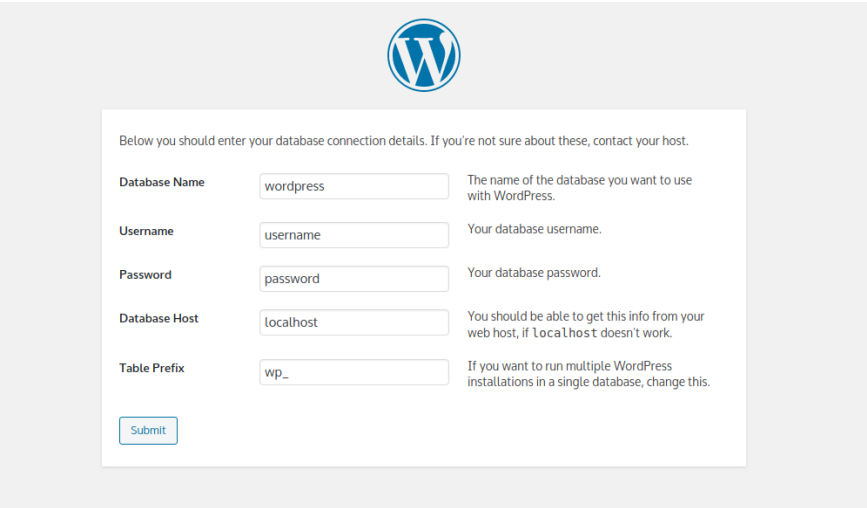

Figure 10: WordPress Database Setup – credentials for database access are entered here so WordPress can utilize MySQL that was installed in Part 2

4. Click “Run the Installation” and WordPress will configure the connection to the database.

5. You should be brought to a page that looks like Figure 11. This page will ask you for information about your new WordPress site. “Site Title” is the name of your website that will appear in the tabs at the top of your web browser. Also fill out a default username and password that you will use to login to the administration panel of your site. Once this information is filled in, click “Install WordPress.”

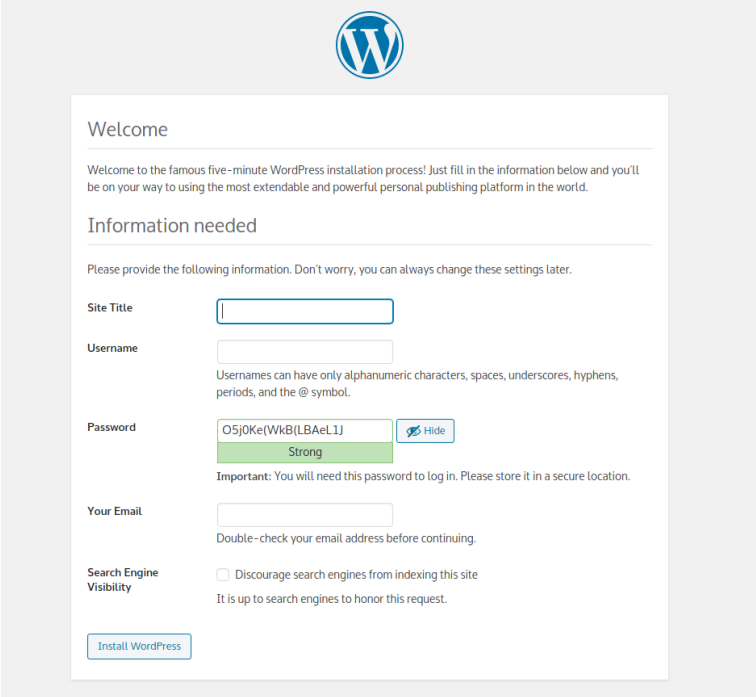

Figure 11: WordPress Site Information – information about your site and users can be entered here

6. The site is now setup! Go to http://<ip-address> to see your webpage where <ip-address> is the IP address from Part 1 Step 11 (ex. 174.129.151.196). You should see a webpage similar to Figure 12.

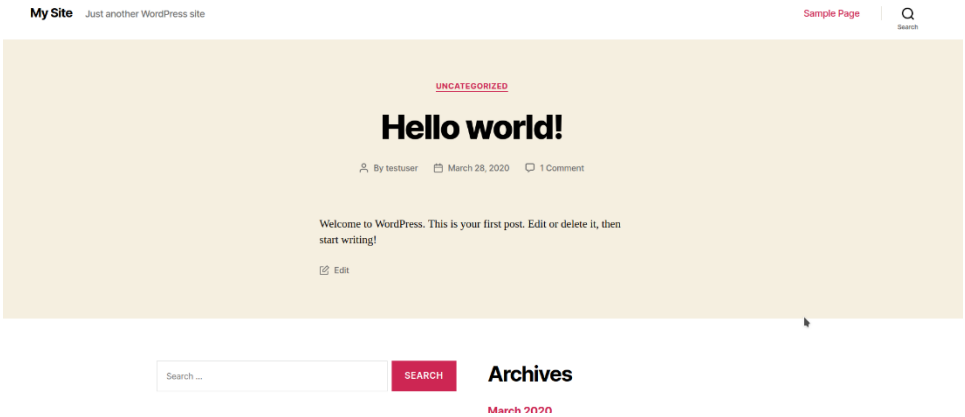

Figure 12: WordPress Homepage – welcome to your personal WordPress website!

7. Pages can be created or edited by going to http://<ip-address>/wp-admin (logging in with the new user we made in Part 3 Step 5). This guide doesn’t cover how to use

WordPress, but https://wordpress.com/learn/ is a resource for learning how to use WordPress.

Future Steps

- As mentioned in Part 3 Step 7, go to https://wordpress.com/learn/ to learn more about building websites on WordPress.
- Purchase a domain name and point that domain name at your server's IP. This means that you could then access your site by a URL such as http://example.com instead of directly by its IP address. Domains can be purchased via registrars such as GoDaddy (https://www.godaddy.com/) or CloudFlare (https://www.cloudflare.com/)
- Secure your WordPress site behind HTTPS. This is a complex topic that requires a decent amount of technical experience working with servers. The website https://certbot.eff.org/ is a good guide to get a website setup with HTTPS.

References

All figures were provided by me.