

ICT171

Introduction to Server Environments and Architectures

(TJD 2025)

Assignment 2- Cloud Server Project

Global IP address: **13.201.150.198**

DNS: **<https://www.umamakhan.xyz/>**

UMAMA KHAN

35405318

Table of Contents

<i>Setting Up a webserver</i>	3
<i>Downloading and Deploying WordPress</i>	6
Changing the sub path.....	10
Linking a Custom Domain Name to AWS EC2 (DNS Setup)	11
SSL/TLS Documentation.....	12
<i>Bibliography</i>	14

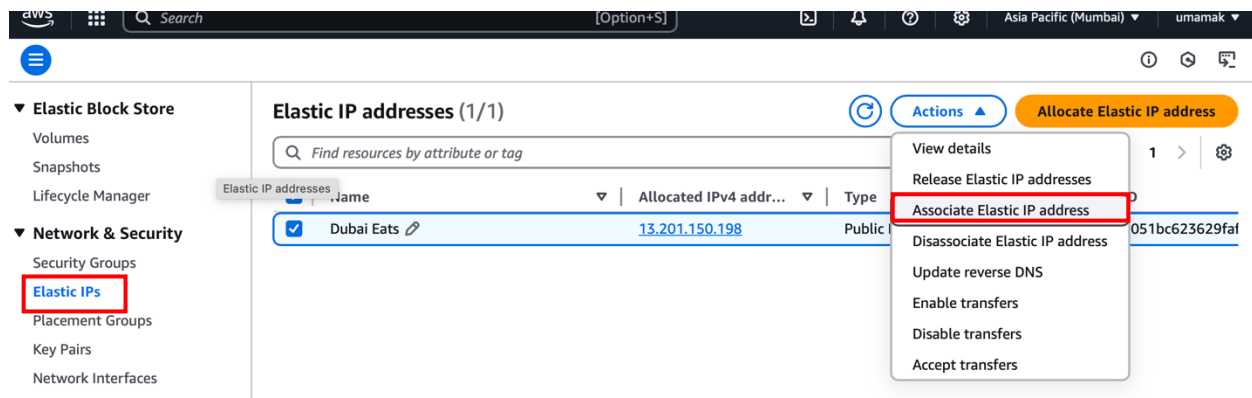
Setting Up a webserver

1. Launch EC2 Instance

An Ubuntu 22.04 LTS server was provisioned using Amazon EC2 with the following configuration:

- **AMI:** Ubuntu Server 22.04 LTS
- **Instance type:** t2.micro (Free Tier)
- **Storage:** 8 GB
- **Security Group Rules:**
 - Port 22 (SSH) – for remote terminal access
 - Port 80 (HTTP) – for web traffic
 - Port 443 (HTTPS) – for SSL traffic

An **Elastic IP** needs to be allocated and associated with the instance to ensure consistent public accessibility.



Elastic IP address: 13.201.150.198

Resource type

Choose the type of resource with which to associate the Elastic IP address.

- ☒ Instance
- ☐ Network interface

⚠ If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

Instance

Private IP address

The private IP address with which to associate the Elastic IP address.

Reassociation

Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.

- ☒ Allow this Elastic IP address to be reassociated

2. Connect to EC2 via SSH and Attach Elastic IP

Instances (1/1) Info

Last updated
less than a minute ago

Connect

Instance state

Actions

Launch instances

All states

Instance state = running

Clear filters

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	Umama's Server	i-01bf176858e489c51	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b

-Using Terminal run the following commands to access the server

Connect to instance Info

Connect to your instance i-01bf176858e489c51 (Umama's Server) using any of these options

EC2 Instance Connect | Session Manager | **SSH client** | EC2 serial console

Instance ID

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is serverkey1.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
4. Connect to your instance using its Public DNS:

Example:

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

3. Update System Packages

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

4. Install Apache Webserver

```
sudo apt install apache2 -y  
sudo systemctl start apache2  
sudo systemctl enable apache2
```

5. Install PHP and MySQL

```
sudo apt install php libapache2-mod-php php-mysql -y  
sudo apt install mysql-server -y
```

6. Start Apache

```
sudo systemctl start apache2  
sudo systemctl enable apache2
```

7. Access MySQL

```
sudo mysql -u root
```

8. Create a new database and database user for wordpress

```
CREATE DATABASE wp;  
CREATE USER 'wp_user'@'localhost' IDENTIFIED BY 'YourPassword';
```

9. Grant all privileges on the database 'wp' to the newly created use

```
GRANT ALL PRIVILEGES ON wp.* TO 'wp_user'@'localhost';  
FLUSH PRIVILEGES;  
EXIT;
```

Downloading and Deploying WordPress

10. Change directory and download WordPress.

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

11. Unzip and move the wordpress folder to apache document root

```
tar -xvf latest.tar.gz  
sudo mv wordpress /var/www/html
```

12. Setting file permissions

Gives Apache ownership of WordPress files

```
sudo chown -R www-data:www-data /var/www/html/wordpress
```

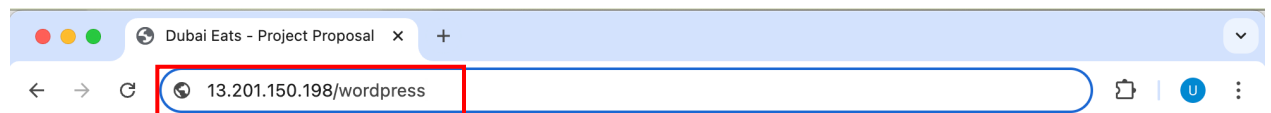
Ensures folders can be entered and executed

```
sudo find /var/www/html/wordpress -type d -exec chmod 755 {} \;
```

Secures files while keeping them readable

```
sudo find /var/www/html/wordpress -type f -exec chmod 644 {} \;
```

13. Enter in a search engine the Public IP address followed by /wordpress



Dubai Eats - Project Proposal

Student ID: 35405318


IP Address: 13.201.150.198

About Dubai Eats

Dubai Eats is a food blog where people can share recipes, review restaurants, and recommend the best places to eat in Dubai. Dubai is known for its diverse food scene, offering everything from street food to fine dining, and this blog provides a space where people can document and explore their food experiences. Whether it's a home cooked meal, a new restaurant, or a local favorite, Dubai Eats is a platform for people to share and discover food related content. The blog aims to have three main sections including- recipes of home cooked meals from different cultures and since recipes can be shared globally, the blog is not just for people in Dubai but also for anyone interested in trying new dishes. It aims to connect locals, tourists, and food enthusiasts from different backgrounds all over the world by providing this platform. Food Reviews will allow users to share their experiences at various restaurants, cafes, etc. helping others make choices about where to eat. Best Places to Eat will highlight different dining spots across the city and can be categorized by cuisine, budget, and location, making it easier to explore new options. Unlike a personal food blog, Dubai Eats is open to contributions from different people, creating a collection of shared experiences and recommendations. It provides a practical resource for those looking for meal ideas, deals, restaurant suggestions as well as insights into Dubai's food scene. Over time the platform will grow into a diverse collection of recipes reviews, and food discussions. For now, this will be a basic website running on top of Amazon EC2, allowing flexibility in hosting and managing the blog's content. Over time, the platform may expand with features such as user contributions, interactive maps, and more structured food recommendations.

License Rationale

14. A welcome page will appear on your screen. Select "Let's go!"



Welcome to WordPress. Before getting started, you will need to know the following items.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

This information is being used to create a `wp-config.php` file. **If for any reason this automatic file creation does not work, do not worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? [Read the support article on `wp-config.php`.](#)

In all likelihood, these items were supplied to you by your web host. If you do not have this information, then you will need to contact them before you can continue. If you are ready...

[Let's go!](#)

15. Enter the database connection information that we generated in Step 9. Click "Submit".

Below you should enter your database connection details. If you are not sure about these, contact your host.

Database Name
The name of the database you want to use with WordPress.

Username
Your database username.

Password [Hide](#)
Your database password.

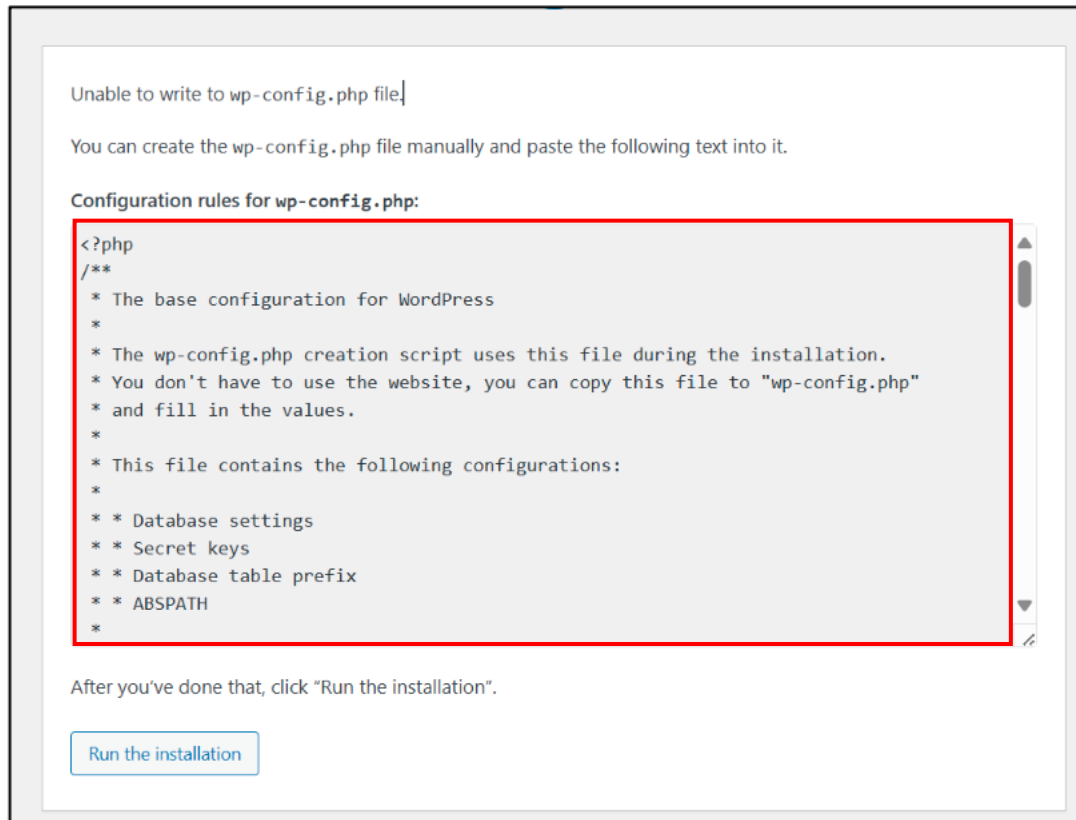
Database Host
You should be able to get this info from your web host, if `localhost` does not work.

Table Prefix
If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)

16. In some cases it will say “*Unable to write to wp-config.php file*”. To fix the issue on terminal manually enter
`cd /var/www/html/wordpress`
`sudo nano wp-config.php`

Then copy the configuration rules in the grey box and run it.



17. After the database connection succeeds fill the information needed and log in.

Below you should enter your database connection details. If you are not sure about these, contact your host.

Database Name
The name of the database you want to use with WordPress.

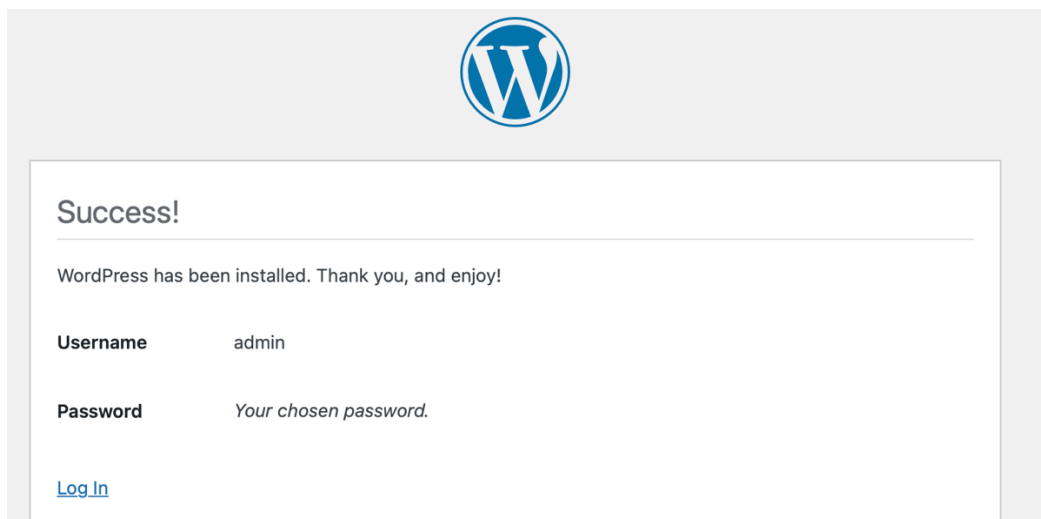
Username
Your database username.

Password [Hide](#)
Your database password.

Database Host
You should be able to get this info from your web host, if localhost does not work.

Table Prefix
If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)



Once WordPress is installed, you can log in to your site by visiting ***http://your-domain.com/wp-admin*** using the admin credentials you set up. From the dashboard, you can download and install plugins, upload themes, create pages and posts, and begin customizing your site to suit your needs.

Changing the sub path

1. Enter the following commands to move WordPress Files to the Root

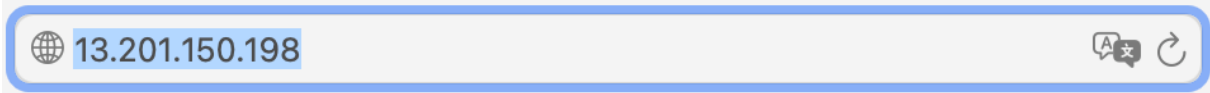
```
sudo mv /var/www/html/wordpress/* /var/www/html/
```

```
sudo mv /var/www/html/wordpress/. * /var/www/html/ 2>/dev/null
```

```
sudo rm -r /var/www/html/wordpress
```

2. Restart Apache server.

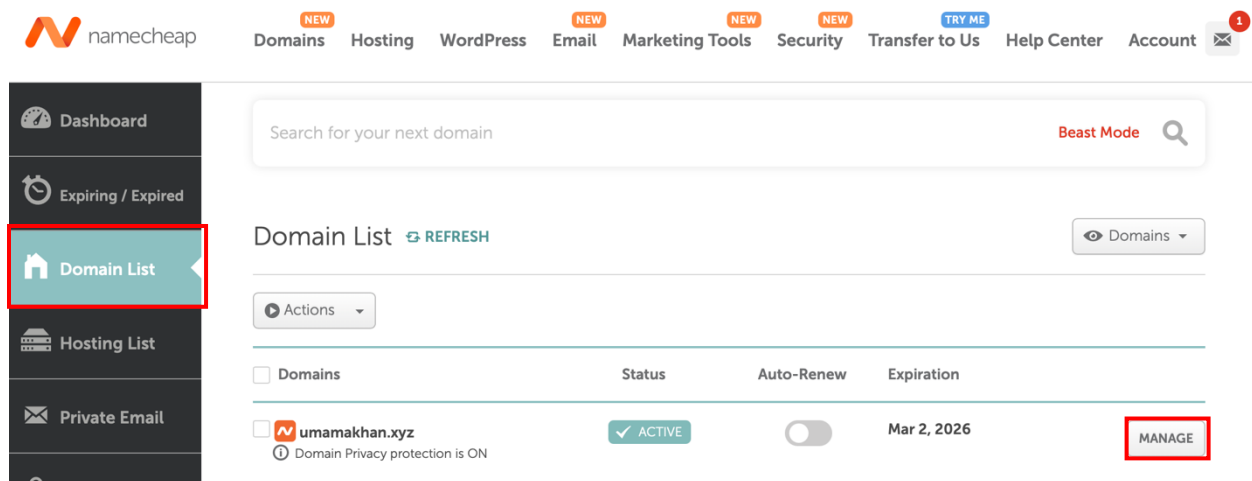
```
sudo systemctl restart apache2
```



The following sequence of commands is used to relocate WordPress from a subdirectory (/wordpress) to the root web directory (/var/www/html), so that the website loads directly at **http://13.201.150.198** instead of <http://13.201.150.198/wordpress>.

Linking a Custom Domain Name to AWS EC2 (DNS Setup)

1. Purchase and register a Domain Name from [Namecheap](#), [godaddy](#), [google domains](#) etc.
2. Access DNS Management Panel



3. In the DNS settings panel, add a **A record** to point the domain to the EC2 Elastic IP and input the values accordingly

<input type="checkbox"/> Type	Host	Value	TTL	
<input type="checkbox"/> A Record	@	13.201.150.198	Automatic	
<input type="checkbox"/> CNAME Record	www	umamakhan.xyz.	Automatic	

A CNAME record was used to ensure that the subdomain www.umamakhan.xyz resolves to the same destination as the root domain umamakhan.xyz (Wikipedia, 2024)



Now you can login using `domainName/wp-admin` and to set up your website on WordPress - <http://www.umamakhan.xyz/wp-admin>

SSL/TLS Documentation

1. Go to Certbot (<https://certbot.eff.org/>) and follow the instructions provided on certbot after selecting the following option:

My HTTP website is running Apache **on** Ubuntu 20

OR

2. Go to EC2 Dashboard and click on running instances and under security and inbound rules **Port 443** is required to enable HTTPS, allowing secure communication between the web server and browser using SSL/TLS. Without it, the SSL certificate from Let's Encrypt cannot be validated or used (Encrypt, 2024).

-	sgr-01aa8599cd3aa86f3	443	TCP	0.0.0.0/0
---	-----------------------	-----	-----	-----------

3. Allow HTTPS in UFW (Ubuntu Firewall) using the commands

```
sudo ufw allow 'Apache Full'
sudo ufw enable
sudo ufw status
```

4. Install Certbot and Apache Plugin

```
sudo apt update
sudo apt install certbot python3-certbot-apache -y
```

5. Request and Install the SSL Certificate

```
sudo certbot --apache
```

User will be prompted to: Enter your email address

```
ubuntu@ip-172-31-12-185:/var/www/html$ sudo certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): umamakhn2503@gmail.com
```

Agree to the terms

```
ubuntu@ip-172-31-12-185:~$ sudo certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
[ (Enter 'c' to cancel): umamakhn2503@gmail.com

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.5-February-24-2025.pdf. You must
agree in order to register with the ACME server. Do you agree?
-----
(Y)es/(N)o: Y

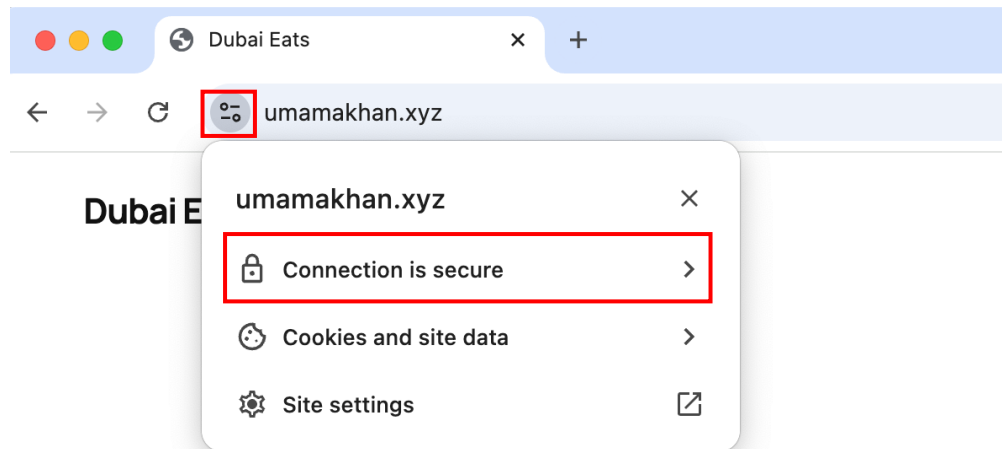
-----
Would you be willing, once your first certificate is successfully issued, to
share your email address with the Electronic Frontier Foundation, a founding
partner of the Let's Encrypt project and the non-profit organization that
develops Certbot? We'd like to send you email about our work encrypting the web,
EFF news, campaigns, and ways to support digital freedom.
-----
(Y)es/(N)o: Y
Account registered.
```

Select Domains

```
Which names would you like to activate HTTPS for?
We recommend selecting either all domains, or all domains in a VirtualHost/server block.
-----
1: umamakhn.xyz
2: www.umamakhn.xyz
-----
Select the appropriate numbers separated by commas and/or spaces, or leave input
blank to select all options shown (Enter 'c' to cancel): 1 2
Requesting a certificate for umamakhn.xyz and www.umamakhn.xyz

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/umamakhn.xyz/fullchain.pem
Key is saved at: /etc/letsencrypt/live/umamakhn.xyz/privkey.pem
This certificate expires on 2025-07-07.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for umamakhn.xyz to /etc/apache2/sites-available/000-default-le-ssl.c
```



6. Enabling Automatic SSL Renewal (Optional)

Auto-renewal ensures that the SSL certificate remains valid beyond its 90-day expiration. This step is optional, but recommended if the website will stay live and secure for long-term use (Encrypt, 2024).

```
sudo systemctl status certbot.timer  
sudo certbot renew --dry-run
```

Bibliography

Wikipedia. (2024, April 7). *CNAME record*. From Wikipedia:

https://en.wikipedia.org/wiki/CNAME_record

Encrypt, L. (2024). *Certbot instructions*. From Electronic Frontier Foundation:

<https://certbot.eff.org/instructions>