DEPLOYING A WORDPRESS ON AWS

Prepared in the partial fulfillment of the Summer Internship Program on AWS

AT

Under the guidance of

Mrs. Sumana Bethala, APSSDC

Mr. Rama Krishna, APSSDC

Submitted by

Bhavana Chukka AP21110010825

Poojitha Pulagam AP21110010815

Charan yarra AP21110010816

Bhargav Sri Krishna AP21110010881

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to all those who have contributed to the successful completion of my summer internship project at **Andhra Pradesh Skill Development Corporation (APSSDC)**. This opportunity has been an enriching and transformative experience for me, and I am truly thankful for the support, guidance, and encouragement I have received along the way.

First and foremost, I extend my sincere regards to Mrs Sumana Bethala and Mr Rama Krishna, my supervisors and mentors, for providing me with valuable insights, constant guidance, and unwavering support throughout the duration of the internship. Their expertise and encouragement have been instrumental in shaping the direction of this project.

I would like to thank the entire team at **Andhra Pradesh Skill Development Corporation (APSSDC)** for fostering a collaborative and innovative environment. The camaraderie, knowledge sharing, and feedback I received from my colleagues significantly contributed to the development and success of this project.

In conclusion, I am honoured to have been a part of this internship program, and I look forward to leveraging the skills and knowledge gained to contribute positively to future endeavours.

Thank you.

TABLE OF CONTENTS

ABSTRACT

This abstract provides an overview of the document "Deploying WordPress on AWS: The full document delves into the step-by-step process of setting up a WordPress website on the Amazon Web Services (AWS) platform. It covers everything from initial prerequisites and account setup to advanced topics such as high availability, security, and cost optimization.

The guide begins by introducing AWS and its benefits, emphasizing the advantages of using AWS services for hosting WordPress. It highlights the importance of AWS account setup and understanding core AWS concepts like Regions, Availability Zones, and Virtual Private Cloud (VPC).

Next, the document explores selecting the appropriate AWS services for deploying WordPress, including Elastic Compute Cloud (EC2) for web hosting, Elastic Block Store (EBS) for data storage, Relational Database Service (RDS) for database management, Simple Storage Service (S3) for media storage, and CloudFront for content delivery.

The deployment process starts with creating a VPC and configuring subnets and security groups. It then guides users through launching an EC2 instance, installing Apache, PHP, MySQL, and deploying WordPress. Emphasis is given to securing the EC2 instance and database, creating regular backups, and setting up IAM roles and permissions.

To ensure high availability and scalability, the guide explains how to use Elastic Load Balancer and Auto Scaling for managing traffic surges and distributing workloads.

Integrating Amazon S3 and CloudFront for media storage and content delivery is also covered in detail. Monitoring and logging with AWS CloudWatch and CloudTrail assist in performance optimization and auditing.

The document concludes with cost optimization strategies, including EC2 Reserved Instances and best practices for WordPress on AWS.

Overall, this comprehensive guide equips readers with the knowledge and practical instructions needed to confidently deploy a fully functional and secure WordPress website on the AWS cloud infrastructure. Whether for personal blogs or enterprise websites, this guide serves as a valuable resource for a seamless WordPress deployment on AWS.

INTRODUCTION

Amazon Web Services (AWS), the cloud computing giant, has emerged as a leading provider of scalable and reliable cloud solutions. By leveraging AWS's powerful suite of services, users can ensure a seamless and efficient deployment of their WordPress websites, delivering exceptional performance and user experience.

The Deploying WordPress on AWS is a comprehensive and practical documentation aimed at simplifying the process of setting up a WordPress website on the AWS platform. This guide is tailored to benefit a diverse audience, including aspiring bloggers, website administrators, developers, and businesses seeking a secure and scalable hosting environment for their WordPress projects.

Throughout this guide, we will navigate the journey of deploying WordPress on AWS step-by-step, starting from the initial setup of an AWS account and understanding key AWS concepts such as Regions, Availability Zones, and Virtual Private Cloud (VPC). We will delve into selecting the most suitable AWS services for our WordPress deployment, ensuring that each component fits together harmoniously to create a seamless website infrastructure.

From launching an Elastic Compute Cloud (EC2) instance to configuring the Relational Database Service (RDS) for robust data management, we will leave no stone unturned in ensuring a reliable and high-performance WordPress environment. Additionally, we will explore integrating Amazon S3 and CloudFront to handle media storage and content delivery efficiently.

Security and data protection are paramount, and this guide dedicates a substantial section to securing our EC2 instance and database, setting up IAM roles and permissions, and establishing regular backups to safeguard our website's integrity.

Moreover, we understand that website traffic can be unpredictable, and maintaining high availability is crucial. We will uncover how to utilize Elastic Load Balancer and Auto Scaling to efficiently handle fluctuations in traffic and maintain an uninterrupted user experience.

Deploying WordPress on AWS aims to empower users with the knowledge and practical know-how to confidently set up and manage a WordPress website on AWS. This guide will serve as a valuable resource for achieving an efficient, scalable, and secure WordPress hosting solution on AWS. Let us embark on this journey together to unlock the full potential of WordPress on the cloud.

PURPOSE OF DEPLOYING A WORDPRESS ON AWS

Deploying WordPress on AWS offers several compelling benefits:

1. Scalability: Easily scale resources to handle fluctuating website traffic, ensuring optimal performance.

2. Reliability: AWS's fault-tolerant infrastructure guarantees high availability and reduces downtime risk.

3. Performance: Global network and CDN services improve page load times, benefiting SEO and user experience.

4. Security: Robust security features, encryption, and compliance certifications protect your data.

5. Cost-Effectiveness: Pay-as-you-go pricing reduces upfront costs, making it ideal for startups and small businesses.

6. Easy Management: User-friendly tools simplify administrative tasks and monitoring.

7. Global Reach: Multiple data centres worldwide allow you to host closer to your audience, minimizing latency.

8. Flexibility: Choose from various instance types and integrate AWS services for tailored configurations.

9. Auto Scaling: Automatically adjust resources based on demand, ensuring optimal performance and cost efficiency.

10. Backups and Disaster Recovery: Reliable backup options and disaster recovery solutions provide data protection.

In summary, AWS empowers WordPress users with a scalable, secure, and cost-efficient hosting environment, freeing them to focus on content creation and growth.

PREREQUISITES

Before starting the deployment of WordPress on AWS, several prerequisites must be in place to ensure a smooth and successful setup. Here is a list of essential prerequisites:

AWS Account: You need a valid AWS account to access and use AWS services. If you don't have one, you can sign up for a free AWS account or a paid account, depending on your requirements.

SSL Certificate: Obtain an SSL certificate to enable HTTPS for your website. This is crucial for securing the data transmitted between your users and the server.

Knowledge of AWS Services: Familiarize yourself with the key AWS services required for hosting WordPress, such as EC2 (Elastic Compute Cloud), RDS (Relational Database Service), S3 (Simple Storage Service), Route 53 (DNS), etc.

Basic Web Server and Database Knowledge: A basic understanding of web servers (e.g., Apache, Nginx) and databases (e.g., MySQL, MariaDB) will be helpful during the setup and configuration process.

VPC and Subnet Setup: If you plan to use a Virtual Private Cloud (VPC) to isolate your resources, create a VPC with appropriate subnets for your deployment.

Security Groups and IAM Roles: Understand how to create security groups to control inbound and outbound traffic and set up IAM roles with necessary permissions for AWS services.

WordPress Installation Package: Download the latest WordPress installation package from the official website (https://wordpress.org) or use tools like `wp-cli` for command-line installation.

Database Information: Decide on the database credentials, including username, password, and database name, for setting up the WordPress database.

Content: If you have existing content from a previous WordPress site or other sources, ensure you have a backup or a plan to migrate the content to the new instance.

DNS Management Access: Access to your domain registrar or DNS management console to update the DNS records to point to your AWS instance.

Having these prerequisites in place before starting the deployment will help streamline the process and minimize potential roadblocks during the setup of WordPress on AWS.

REGULAR BACKUPS FOR BOTH WORDPRESS FILES AND DATABASE

WordPress Database Backup:

a. Create an RDS Snapshot: If you are using Amazon RDS to host your WordPress database, take regular snapshots using the AWS RDS console. This can be scheduled within the RDS service itself.

Log in to your AWS console, navigate to the RDS service, select your database instance, and go to the "Snapshots" tab.

Click on "Create snapshot" and provide a meaningful name for the snapshot.

You can set a retention policy to automatically delete older snapshots to manage storage costs.

Install and configure the backup plugin of your choice from the WordPress dashboard.

Set the plugin to perform regular database backups, and choose to store the backups in a designated folder on your server or an external storage service like Dropbox or Google Drive.

Backup Validation:

Periodically check the backups to ensure they are running correctly and the data is being stored in the designated locations.

Restore Testing:

It's crucial to test the restoration process from backups to verify that the backups are functioning correctly. Ensure you can restore both the WordPress files and the database to a test environment and verify that the site works as expected.

Remember to secure your backups by implementing access controls and encryption where necessary. Regularly monitor the backup process to ensure it continues to run smoothly. Having a reliable backup strategy gives you peace of mind knowing your valuable WordPress data is safe and recoverable in case of emergencies.

STEPS FOR DEPLOYING A WORDPRESS ON AWS

Deploying WordPress on AWS involves several steps, starting from creating an EC2 instance and setting up the necessary components. Below are the main steps for deploying WordPress on AWS from EC2 instance creation:

Step 1: Sign in to AWS Console

- Log in to your AWS Management Console using your AWS account credentials.

Step 2: Launch an EC2 Instance

- Navigate to the EC2 Dashboard and click on the "Launch Instance" button.

- Choose an Amazon Machine Image (AMI) for your instance. You can select a pre-configured WordPress AMI or a basic Linux AMI.

- Select an instance type based on your website's requirements and estimated traffic.

Step 3: Configure Instance Details

- Choose the number of instances to launch

- Configure the network settings, including selecting a VPC and subnet.

- Assign a public IP to the instance for internet access.

Step 4: Add Storage

- Select the desired storage volume (EBS) for your instance

Step 5: Configure Security Group

- Create or select an existing security group to control inbound and outbound traffic to your instance.

- Allow HTTP (port 80) and HTTPS (port 443) traffic to enable web access.

Step 6: Review and Launch

- Review the configuration settings for your instance and make any necessary changes.

- Launch the instance and select an existing key pair or create a new one to connect securely via SSH.

Step 7: Connect to the Instance

- Once the instance is running, use SSH to connect to the instance using the key pair you selected.

Step 8: Install LAMP Stack (Linux, Apache, MySQL, PHP)

- Update the system packages**: `sudo apt update`** (for Ubuntu-based systems).

This command updates the package list on the Linux system. It ensures that the package manager retrieves the latest information about available software packages and their versions

- Install Apache web server: **`sudo apt install apache2`.**

This command installs the Apache web server on the Linux system. Apache is a widely used web server that serves web content to visitors' browsers.

- Install MySQL server: **`sudo apt install mysql-server -y`.**

This command installs the MySQL server on the Linux system. MySQL is a relational database management system used to store data for WordPress.

-Install MySQL client: **sudo apt install mysql-client -y`.**

-To enter into mysql prompt: **mysql -u root -p**

the MySQL client will prompt you to enter the root user's password. After you enter the correct password, the MySQL client will connect to the MySQL server as the root user, and you can then execute SQL commands or manage the databases and users as needed.

-To create a Database: **create database wordpress;**

command is used in MySQL to list all the databases available on the MySQL server. When you run this command, MySQL will display a list of database names that are currently present on the server.

-To create the username and the password for the database wordpress

mysql>GRANT ALL ON wordpress.* TO 'apssdc' @ '%'IDENTIFIED BY '123456';

This command is used to grant specific privileges to a MySQL user for a specific database**.**

-To exit from the mysql : **\q**

When you run this command, it will terminate the current session and return you to the regular command prompt or shell, effectively ending your interaction with the MySQL server.

PHP INSTALLATION

-apt-get install php -y

This command is used to install the PHP programming language on a Linux system using the Advanced Package Tool (APT).

-apt-get install php-mysql -y

The PHP MySQL extension is required if you want to connect and interact with MySQL databases from PHP scripts. It enables PHP to communicate with MySQL servers, allowing you to perform database operations such as querying data, inserting records, updating data, and more.

-sudo apt install php-curl php-gd php-mbstring php-xmlrpc php-soap php-intl php-zip

php-curl: This extension adds support for cURL (Client URL Library) functions, allowing PHP to make HTTP requests, access remote resources, and handle URLs.

php-gd: This extension enables PHP to work with the GD Graphics Library, which provides functions for image processing and manipulation.

php-mbstring: This extension provides multibyte string functions, essential for handling multibyte character encodings like UTF-8.

php-xmlrpc: This extension allows PHP to act as an XML-RPC server and client, enabling communication with other systems using XML-RPC.

php-soap: This extension provides support for SOAP (Simple Object Access Protocol) web services, allowing PHP to interact with SOAP-based services.

php-intl: This extension adds internationalization and localization capabilities to PHP, supporting features like language translation, formatting dates, and handling Unicode text.

php-zip: This extension provides support for handling ZIP archives, allowing PHP to create and extract ZIP files.

WORDPRESS INSTALLATION

Step 9: Download and Configure WordPress

- Download the latest version of WordPress**: curl -O https://wordpress.org/latest.tar.gz`.**

By using the above link we will download the compressed WordPress archive to your local machine. This archive contains all the files and directories necessary to set up a WordPress website.

- Extract the downloaded file: **`tar -xvzf latest.tar.gz`.**

This command will decompress and extract the files from the "latest.tar.gz" archive into the current working directory. The extracted files will typically include the entire WordPress installation, including the core files, plugins, themes, and other necessary resources.

- Move WordPress files to Apache's web directory: **cp -r wordpress/ /var/www/html/`.**

The cp command will copy all the files and subdirectories inside the "wordpress" directory to the "/var/www/html/" directory. It effectively duplicates the WordPress installation from the "wordpress" directory to the web server's document root, which is typically "/var/www/html/" on many Linux web servers.

- Configure the WordPress database by creating a MySQL database and user.

-cd /var/www/html/

The shell or terminal will change the current directory to /var/www/html/. This means that any subsequent commands or operations you perform will take place in this directory.

-chmod 777 wordpress/

This command is used to change the permissions of the "wordpress" directory and its contents on a Linux or Unix-based system.

Step 10: Set Permissions and Configure Apache

-chown -R www-data:www-data wordpress

This command is used to change the ownership of the "wordpress" directory and all its contents to the user and group "www-data" on a Linux or Unix-based system**.**

-cd wordpress

The command is used to change the current working directory to the "wordpress" directory, assuming that it exists in the current directory or is a subdirectory of the current directory

-nano wp-config-sample.php

The command nano wp-config-sample.php is used to open the "wp-config-sample.php" file in the nano text editor on a Linux or Unix-based system.

In the nano editor give the database name,username and the password as required

-ip/wordpress

The URL "ip/wordpress" is typically used to access the WordPress installation on a web server using the server's IP address (or domain name) and the "wordpress" subdirectory.

- Create an Apache configuration file for your website and enable it: **`sudo nano /etc/apache2/sites-available/your_domain.conf`.**

The command is used to open a configuration file for an Apache Virtual Host on a Linux system using the nano text editor with administrative (sudo) privileges.

-nano 000-default.conf

The command is used to open the configuration file for the default Apache Virtual Host on a Linux system using the nano text editor.

-Document editor:/var/www/html/wordpress

This command is used for the directory path where the WordPress application has been installed. When you access your web server through a web browser, the server will serve the WordPress files from this directory to display your website.

-cd/etc/mysql/

This command is used for the shell or terminal will change the current directory to "/etc/mysql/". This means that any subsequent commands or operations you perform will take place within the "/etc/mysql/" directory.

-cd mysql.conf.d

This is the command where the shell or terminal will change the current directory to "mysql.conf.d." This means that any subsequent commands or operations you perform will take place within the "mysql.conf.d" directory.

-nano mysqld.cnf

The command nano mysqld.cnf is used to open the "mysqld.cnf" configuration file for the MySQL server in the nano text editor on a Linux or Unix-based system.

Step 11: Enable Required Apache Modules

- Enable Apache modules: `sudo a2enmod rewrite` and `sudo a2enmod headers`.

- Restart Apache: `sudo service apache2 restart`.

Step 12: Complete WordPress Installation

- Access your domain in a web browser and follow the WordPress setup wizard.

- Provide the database credentials and other required information.

TROUBLESHOOTING

Deploying and managing a WordPress site on AWS can encounter various challenges. Here are some common issues that may arise and troubleshooting steps to address them:

Instance Connectivity Issues:

- Problem: Unable to connect to the EC2 instance via SSH.

- Troubleshooting:

  - Verify that the instance has a public IP or is in a public subnet.

  - Check the security group settings to ensure SSH access is allowed.

  - Verify that the SSH key used for connecting has the correct permissions (chmod 400).

  - If using a custom VPC, ensure that network ACLs and route tables allow inbound and outbound traffic.

Website Not Loading:

- Problem: The WordPress website doesn't load or shows an error page.

- Troubleshooting:

  - Check the web server (e.g., Apache, Nginx) logs for errors or configuration issues.

  - Verify that the domain name is correctly mapped to the EC2 instance's public IP or Elastic Load Balancer.

  - Ensure the web server is running, and there are no issues with the PHP or WordPress installation.

Database Connection Errors:

- Problem: WordPress displays errors related to the database connection.

- Troubleshooting:

  - Verify that the database server (e.g., RDS) is running and accessible.

  - Double-check the database configuration settings in the WordPress wp-config.php file.

  - Ensure the database credentials (username and password) are correct.

  - Check if the database security group allows inbound traffic from the web server.

High Resource Utilization:

- Problem: The WordPress site experiences slow performance or becomes unresponsive.

- Troubleshooting:

  - Monitor the EC2 instance's CPU, memory, and disk usage to identify resource bottlenecks.

  - Check for inefficient plugins or themes that may be causing high resource consumption.

  - Consider scaling the infrastructure or optimizing the server setup to handle increased traffic.

Backup and Restore Issues:

- Problem: Unable to restore from backups or backup process failing.

- Troubleshooting:

  - Verify that the backup files are accessible and located in the correct location.

  - Check the backup script or plugin configuration for any errors or misconfigurations.

  - Ensure that you have sufficient storage space for backups and manage retention policies to avoid storage overflows.

Security and Permission Errors:

- Problem: WordPress or AWS services display permission-related errors.

- Troubleshooting:

  - Review the IAM roles and permissions associated with your AWS resources.

  - Ensure that appropriate security groups and network ACL settings are configured.

  - Check file and directory permissions on the EC2 instance to prevent unauthorized access.

Email Deliverability Issues:

- Problem: WordPress emails, such as password reset or contact form submissions, are not being delivered.

- Troubleshooting:

  - Ensure that the EC2 instance is not on an email blacklist.

  - Consider using third-party email services like Amazon SES for better email deliverability.

  - Check WordPress email settings and plugins to confirm they are configured correctly.

CONCLUSION

In conclusion, this comprehensive Word document serves as a definitive guide for deploying WordPress on AWS (Amazon Web Services). By following the step-by-step instructions and best practices provided in this document, users can confidently set up a scalable, reliable, and secure WordPress website on the AWS cloud.

The document starts by outlining the purpose and benefits of deploying WordPress on AWS, highlighting the advantages of AWS services such as scalability, cost efficiency, reliability, and security. It emphasizes the importance of understanding the prerequisites, including the need for an AWS account, domain name, SSL certificate, and basic knowledge of AWS services and web servers.

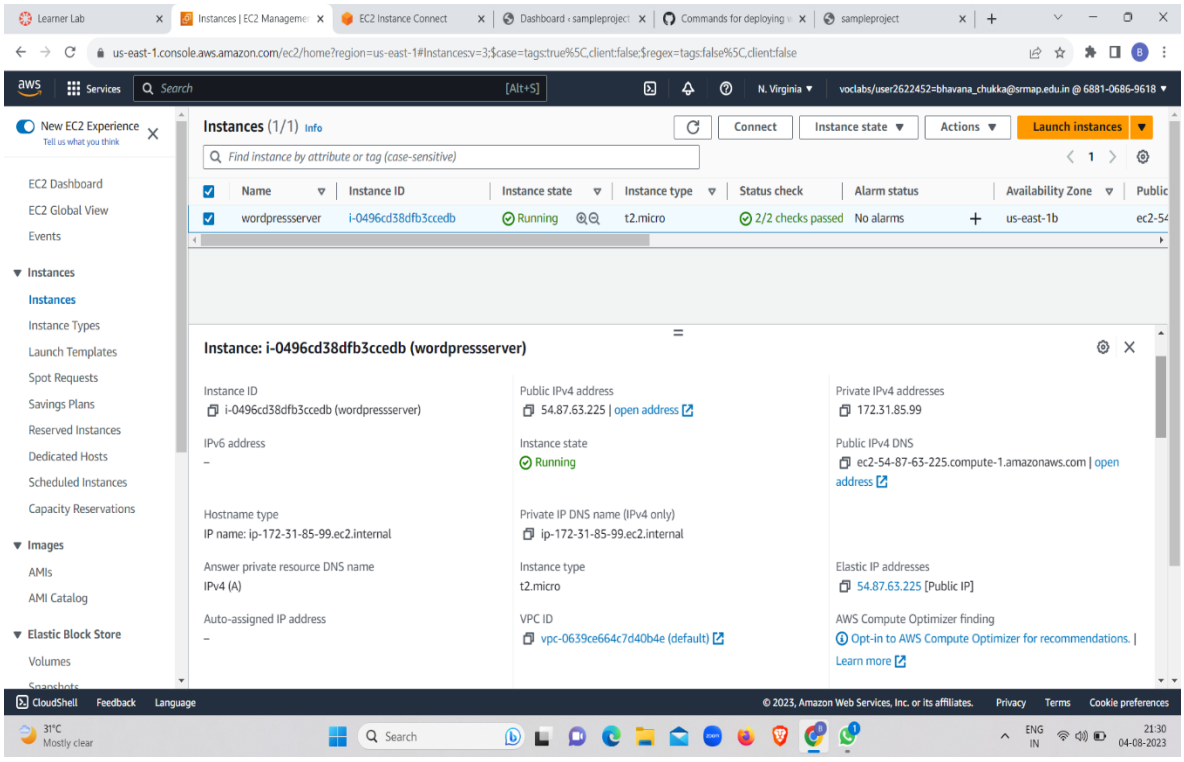
The deployment process is thoroughly covered, guiding users through creating the necessary AWS resources, setting up an EC2 instance, configuring an RDS database, installing WordPress, securing the environment, and configuring DNS for the website. Additionally, it explains how to monitor the website's performance and scale the infrastructure when needed, ensuring optimal performance during traffic spikes.

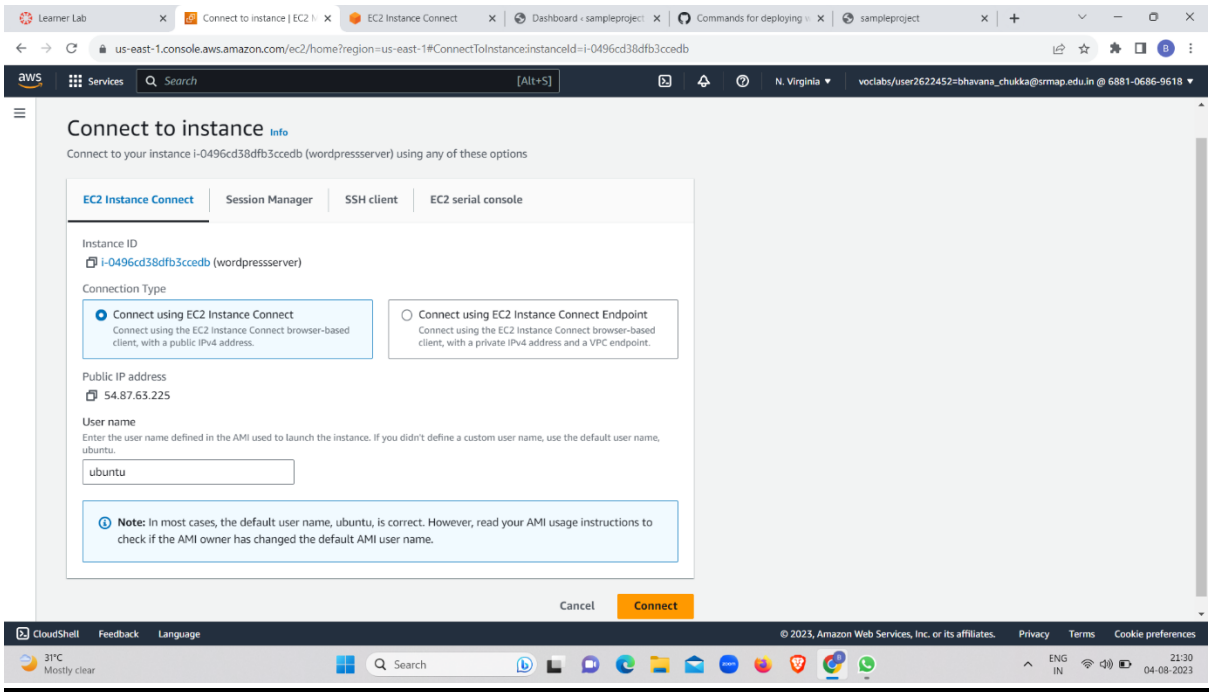
Crucially, the document underscores the significance of implementing a robust backup and disaster recovery strategy. By setting up regular backups for both WordPress files and the database, users can ensure data safety and quick recovery in case of any unforeseen incidents.

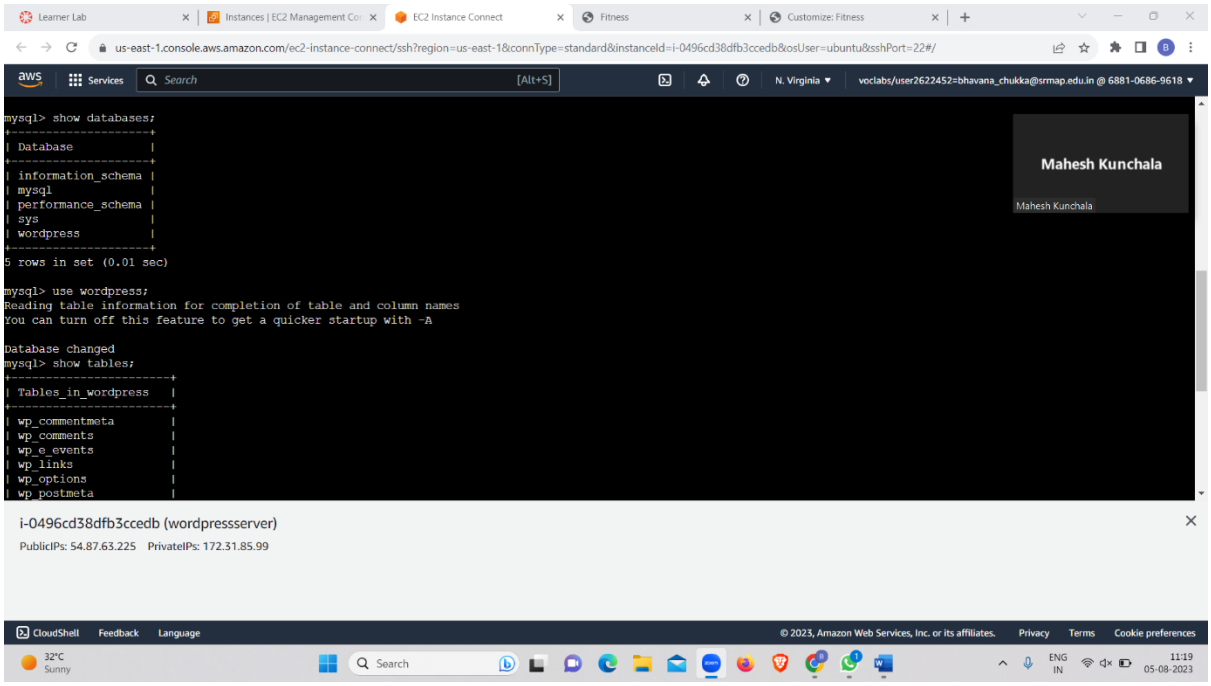
Moreover, a dedicated troubleshooting section has been included to assist users in identifying and resolving common issues that may arise during the deployment or management of the WordPress site on AWS. With step-by-step troubleshooting guidance, users can efficiently resolve connectivity, performance, security, and other potential problems.

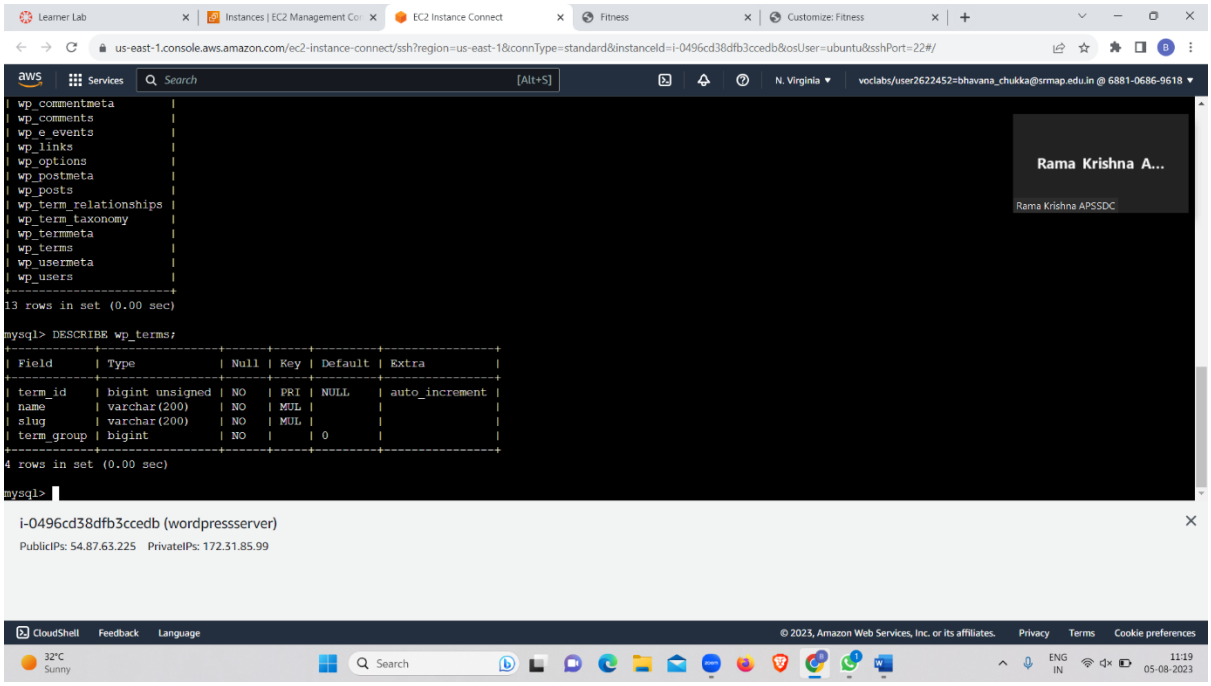
In conclusion, this Word document serves as a valuable resource for anyone seeking to deploy WordPress on AWS. It provides a clear roadmap for successfully creating and maintaining a high-performance WordPress website while leveraging the powerful features and benefits offered by AWS. By adhering to the documented steps and recommendations, users can confidently navigate the deployment process, optimize their AWS resources, and deliver a reliable and engaging WordPress site to their audience.

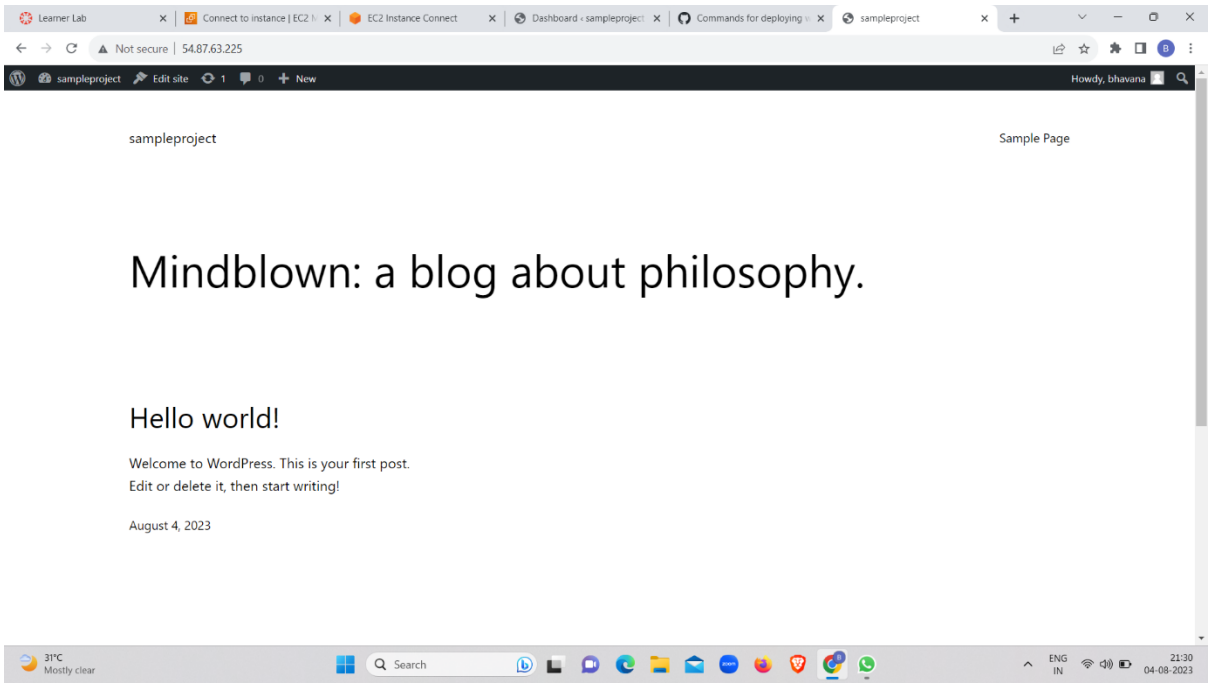
RESULTS:

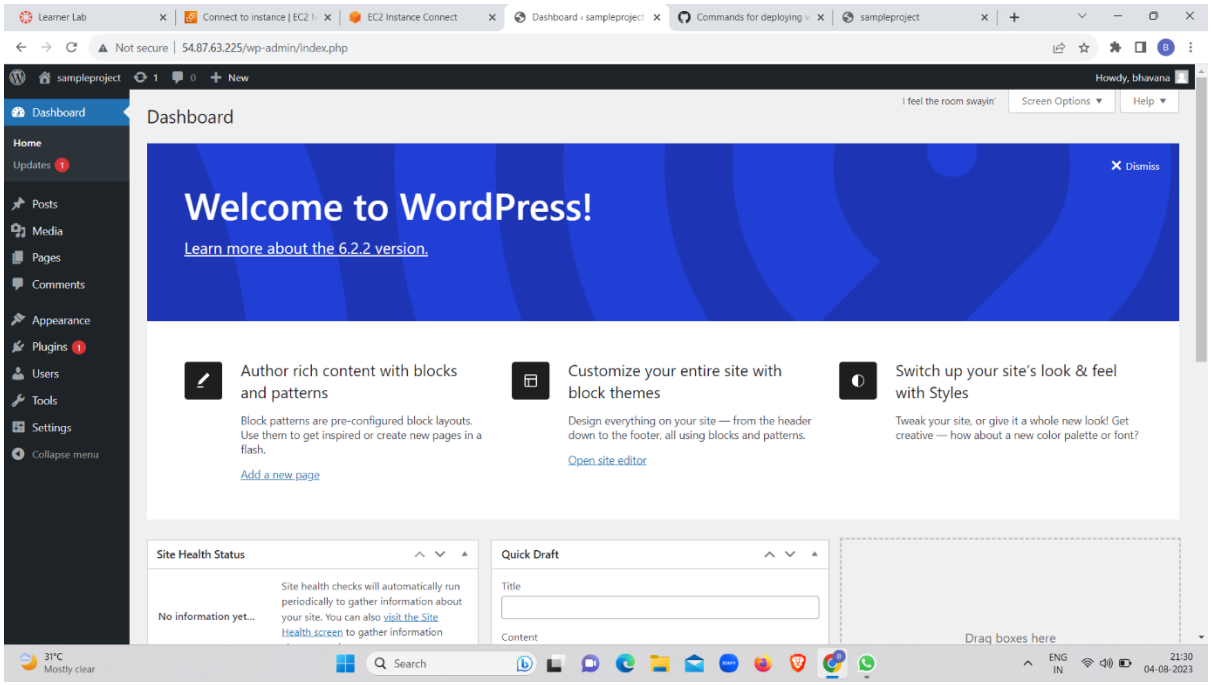

Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
wordpressserver
i-0496cd38dfb3ccedb
Running
t2.micro
2/2 checks passed
No alarms
us-east-1b
Instance: i-0496cd38dfb3ccedb (wordpressserver)
Instance ID
i-0496cd38dfb3ccedb (wordpressserver)
Public IPv4 address
54.87.63.225 | open address
Private IPv4 addresses
172.31.85.99
IPv6 address
Instance state
Running
Public IPv4 DNS
ec2-54-87-63-225.compute-1.amazonaws.com | open address
Hostname type
IP name: ip-172-31-85-99.ec2.internal
Private IP DNS name (IPv4 only)
ip-172-31-85-99.ec2.internal
Answer private resource DNS name
IPv4 (A)
Instance type
t2.micro
Elastic IP addresses
54.87.63.225 [Public IP]
Auto-assigned IP address
VPC ID
vpc-0639ce664c7d40b4e (default)
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more


Connect to instance Info
Connect to your instance i-0496cd38dfb3ccedb (wordpressserver) using any of these options
EC2 Instance Connect
Session Manager
SSH client
EC2 serial console
Instance ID
i-0496cd38dfb3ccedb (wordpressserver)
Connection Type
Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.
Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.
Public IP address
54.87.63.225
User name
Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.
ubuntu
Note: In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.
Cancel
Connect


Learner Lab
Instances | EC2 Management Co
EC2 Instance Connect
Fitness
Customize: Fitness
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-0496cd38dfb3ccedb&osUser=ubuntu&sshPort=22#/
aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user2622452=bhavana_chukka@srmap.edu.in @ 6881-0686-9618
mysql> show databases;
+--------------------+
| Database |
+--------------------+
| information_schema |
| mysql |
| performance_schema |
| sys |
| wordpress |
+--------------------+
5 rows in set (0.01 sec)
mysql> use wordpress;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> show tables;
+----------------------+
| Tables_in_wordpress |
+----------------------+
| wp_commentmeta |
| wp_comments |
| wp_e_events |
| wp_links |
| wp_options |
| wp_postmeta |
Mahesh Kunchala
Mahesh Kunchala
i-0496cd38dfb3ccedb (wordpressserver)
PublicIPs: 54.87.63.225 PrivateIPs: 172.31.85.99
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
32°C
Sunny
Search
ENG
IN
11:19
05-08-2023


Learner Lab
Instances | EC2 Management Co
EC2 Instance Connect
Fitness
Customize: Fitness
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-0496cd38dfb3ccedb&osUser=ubuntu&sshPort=22#/
aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user2622452=bhavana_chukka@srmap.edu.in @ 6881-0686-9618
| wp_commentmeta |
| wp_comments |
| wp_e_events |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----------------------+
13 rows in set (0.00 sec)
mysql> DESCRIBE wp_terms;
+------------+-----------------+------+-----+---------+----------------+
| Field | Type | Null | Key | Default | Extra |
+------------+-----------------+------+-----+---------+----------------+
term_id	bigint unsigned	NO	PRI	NULL	auto_increment
name	varchar(200)	NO	MUL		
slug	varchar(200)	NO	MUL		
term_group	bigint	NO		0	
+------------+-----------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
mysql>
Rama Krishna A...
Rama Krishna APSSDC
i-0496cd38dfb3ccedb (wordpressserver)
PublicIPs: 54.87.63.225 PrivateIPs: 172.31.85.99
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
32°C
Sunny
Search
ENG
IN
11:19
05-08-2023


Learner Lab
Connect to instance | EC2
EC2 Instance Connect
Dashboard ‹ sampleproject
Commands for deploying
sampleproject
Not secure | 54.87.63.225
sampleproject
Edit site
New
Howdy, bhavana
sampleproject
Sample Page
Mindblown: a blog about philosophy.
Hello world!
Welcome to WordPress. This is your first post.
Edit or delete it, then start writing!
August 4, 2023
31°C
Mostly clear
Search
ENG
IN
21:30
04-08-2023


Learner Lab
Connect to instance | EC2
EC2 Instance Connect
Dashboard ‹ sampleproject
Commands for deploying
sampleproject
Not secure | 54.87.63.225/wp-admin/index.php
sampleproject
New
Howdy, bhavana
Dashboard
Home
Updates
Posts
Media
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
Dashboard
I feel the room swayin'
Screen Options
Help
Dismiss
Welcome to WordPress!
Learn more about the 6.2.2 version.
Author rich content with blocks and patterns
Block patterns are pre-configured block layouts. Use them to get inspired or create new pages in a flash.
Add a new page
Customize your entire site with block themes
Design everything on your site — from the header down to the footer, all using blocks and patterns.
Open site editor
Switch up your site's look & feel with Styles
Tweak your site, or give it a whole new look! Get creative — how about a new color palette or font?
Site Health Status
No information yet...
Site health checks will automatically run periodically to gather information about your site. You can also visit the Site Health screen to gather information
Quick Draft
Title
Content
Drag boxes here
31°C
Mostly clear
Search
ENG
IN
21:30
04-08-2023


Learner Lab
Instances | EC2 Management Co
EC2 Instance Connect
Fitness
Customize: Fitness
Not secure | 54.87.63.225/wp-admin/customize.php?return=%2Fwp-admin%2Fadmin.php%3Fpage%3Doceanwp
Published
Active theme
OceanWP
Change
Site Identity
Menus
Widgets
Homepage Settings
Typography
General Options
Blog
Header
Top Bar
Footer Widgets
Footer Bottom
Sidebar
Custom CSS/JS
Hide Controls
+111 22 33 444
Monday - Sunday 10:00 - 22:00
COACH
HOME
YOUR OBJECTIVE
ACHIEVE YOUR GOAL
CHOOSE YOUR PACK
PRICING
Rama Krishna A...
Rama Krishna APSSDC
Auctor sem massa mattis sem at interdum magna augue eget diam vestibulum ante ipsum primis in faucibus orci luctus
PROFESSIONAL
SPORTS COACH
SPORTS COACH AT HOME
MAKE AN APPOINTMENT
32°C
Sunny
Search
ENG
IN
11:19
05-08-2023