

GLA UNIVERSITY MATHURA

Mentor

Mr. Raushan Kumar Singh

Dr. Ruby Pawar

25-10-2024

1

Team



Prakhar
Shukla
(2215001261)



Ankur
Sinsinwar
(2215000273)



Tanisha Singh
(2215001832)



Anuj
Chaturvedi
(2215000302)



Sakshi Tangar
(2215001558)



Motivation

Cost Efficiency

Hosting multiple websites on a single server reduces infrastructure costs compared to using separate servers for each site.



Scalable

(Hosting on Amazon Linux offers scalable options, where you can expand the server resources as needed.)



Flexibility

You can fully customize the server for various websites, adapting to different business or user needs.



Simplified Management

Using Amazon Linux in the AWS Mumbai region ensures that latency is reduced for local users, improving performance.

Objective for Multi-Website Hosting



Overview of Multi-Website Hosting

An introduction to the concept and importance of hosting multiple websites.



Benefits of Using AWS EC2 and CloudFront

Discusses the advantages of utilizing AWS EC2 for scalable computing and Route 53 for reliable DNS management.



AWS Account Setup

Steps to create and configure an AWS account for hosting services.



Required Software and Tools

A list of essential software and tools needed for the project.



Launching an EC2 Instance

Instructions on how to launch a new EC2 instance for hosting.



Multi-Website Hosting on AWS Amazon Linux Server in Mumbai Region

Definition of Multi-Website Hosting

Multi-website hosting refers to the practice of hosting more than one website on a single server, making it a cost-effective and efficient solution for small to medium-sized websites.



AWS EC2 Overview

AWS EC2, when combined with CloudFront, provides a powerful solution for hosting multiple websites with high availability and scalability.



Pay As You Go Model

Pay only for the resources you use, making it a budget-friendly option for hosting multiple websites.



Flexibility

Gain full control over your server environment, allowing for tailored configuration and management.

AWS Services and Required Tools for Website Hosting

- **EC2 (Elastic Compute Cloud)**

Create and manage Amazon Linux instances. Choose an appropriate instance type based on your workload, such as t2.micro for light loads and m5.large for demanding applications. Set up security groups to control inbound and outbound traffic.

- **Elastic Load Balancing (ELB)**

Distribute traffic across multiple EC2 instances to ensure high availability and fault tolerance.

- **Auto Scaling**

Automatically scale your instances up or down based on demand, optimizing costs and performance.

- **Amazon RDS (Relational Database Service)**

Managed database service supporting MySQL, PostgreSQL, MariaDB, and more. Simplifies database management tasks like backups, patching, and scaling.

- **Amazon S3 (Simple Storage Service)**

Store static assets, backups, or other data that need to be accessed by your applications.

- **Amazon CloudFront**

Content Delivery Network (CDN) designed to deliver your content with low latency.

- **Amazon Route 53**

Domain Name System (DNS) web service to route end users to your applications.

Setting Up Your AWS Account

Create an AWS Account

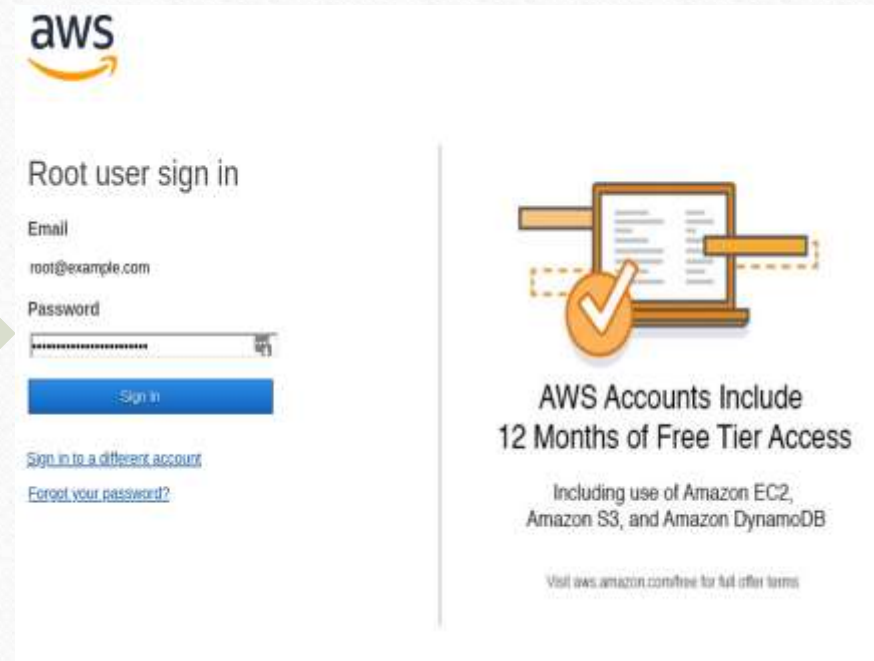
Visit the AWS website and follow the sign-up process to create your account

Valid Payment Method

Ensure you have a valid payment method ready for account setup.

Identity Verification

Complete the necessary identity verification steps during the sign-up process.



Launching an EC2 Instance on AWS

Log in to AWS Management Console

Access the AWS Management Console using your credentials.

Navigate to EC2 Dashboard

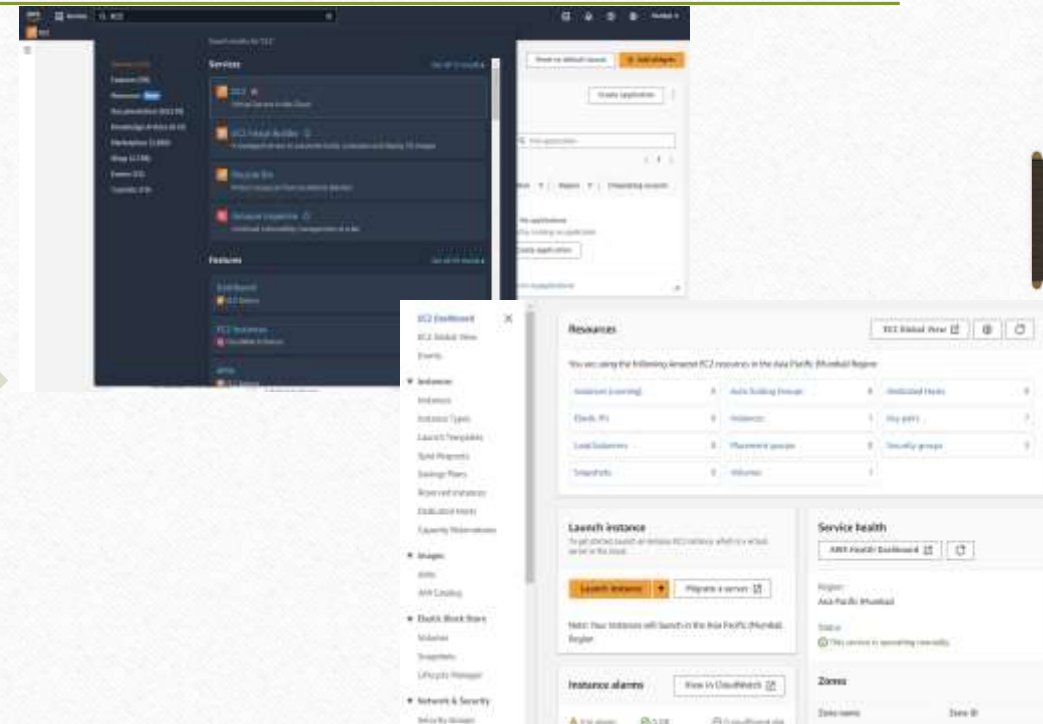
Locate and navigate to the EC2 Dashboard from the console.

Sign in as root user or IAM

Choose to sign in as a root user or with IAM to work within a Group.

Click 'Launch Instance'

Initiate the instance creation process by clicking on 'Launch Instance'.



Choosing AMI and Instance Type for EC2



Launch Instance

Click the Launch instance button to initiate the process of launching an EC2 instance.



Select Amazon Machine Image (AMI)

Choose an Amazon Machine Image (AMI) for your instance. For this guide, we will use Amazon Linux 2023.



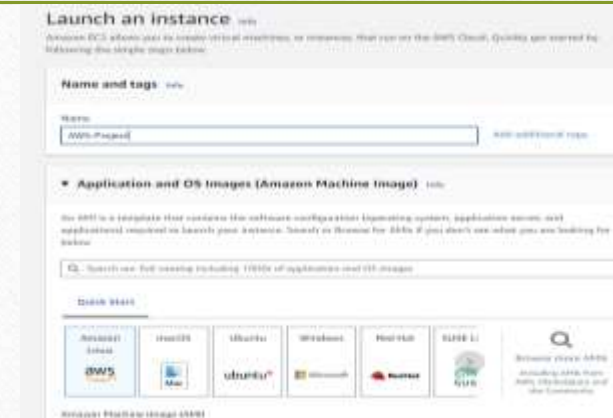
Name Your Instance

Assign the name 'AWS-Project' to your instance while selecting the Amazon Linux 2023 AMI.



Select Instance Type

Choose an instance type suitable for your needs, such as t2.micro, which qualifies for the free tier.



Instance type

Instance type

t2.micro
Family: t2 | 1 vCPU | 1 GB Memory | Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Configuring Instance Details and Key Pair



Configure Instance Details

- Set up instance details, including network settings and storage configurations.



Add Key Pair for SSH Access

- Include a key pair to enable secure SSH access to your instance.



Name Your Key Pair

- Assign a name to your key pair and select the key format (.ppm for Linux).



Review and Launch Instance

- > Check all configurations and click the Launch instance button. Wait for the instance to be ready.

Create key pair [Info](#)

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

AWS-Project-key

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type [Info](#)

☒ RSA

☐ ED25519

Private key file format

☒ .pem

For use with OpenSSH

☐ .ppk

For use with PuTTY

Tags - optional

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#)

[Create key pair](#)

Connecting to Your EC2 Instance

To connect to your EC2 instance, you typically use SSH, requiring the instance's public IP address or DNS and the PEM file (private key) created when launching the instance. Security Groups must allow inbound SSH traffic on port 22 from your IP address. Elastic IP addresses provide a consistent public IP, even after restarting the instance. You can also use EC2 Instance Connect through the AWS Management Console without managing SSH keys. Proper permissions for the PEM file are necessary. Additionally, User Data scripts allow automated configuration during instance initialization.



EC2 > Instances > i-02377ed647bad687c > Connect to instance

Connect to instance Info

Connect to your instance i-02377ed647bad687c (AWS-Project) using any of these options.

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-02377ed647bad687c (AWS-Project)

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

3.108.51.191

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Connect

Configuring a Linux Server and Installing Apache

```

AWS Services Search [Alt+S]
EC2
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Tue Jul 23 07:48:10 2024 from 13.233.177.4
[ec2-user@ip-172-31-6-20 ~]$ sudo su
[root@ip-172-31-6-20 ec2-user]# yum install httpd -y
Last metadata expiration check: 0:08:48 ago on Tue Jul 23 07:45:49 2024.
Dependencies resolved.

Package Architecture Version
Installing:
httpd x86_64 2.4.61-1.amzn202
Installing dependencies:
apr x86_64 1.7.2-2.amzn2023
apr-util x86_64 1.6.3-1.amzn2023
generic-logos-httpd noarch 18.0.0-12.amzn20
httpd-core x86_64 2.4.61-1.amzn202
httpd-filesystem noarch 2.4.61-1.amzn202
httpd-tools x86_64 2.4.61-1.amzn202
libbrotli x86_64 1.0.9-4.amzn2023
mailcap noarch 2.1.49-3.amzn202
Installing weak dependencies:
apr-util-openssl x86_64 1.6.3-1.amzn2023
mod_http2 x86_64 2.0.27-1.amzn202

```

```

[root@ip-172-31-6-20 ec2-user]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-6-20 ec2-user]#

```

First, update your package repositories by running
sudo yum update -y
on Amazon Linux This ensures all installed software is up to date.

Use
sudo yum install httpd -y
to install Apache web server

Start Apache with
Service httpd start

Web Page for Site Overview

Welcome to My Hosted Websites

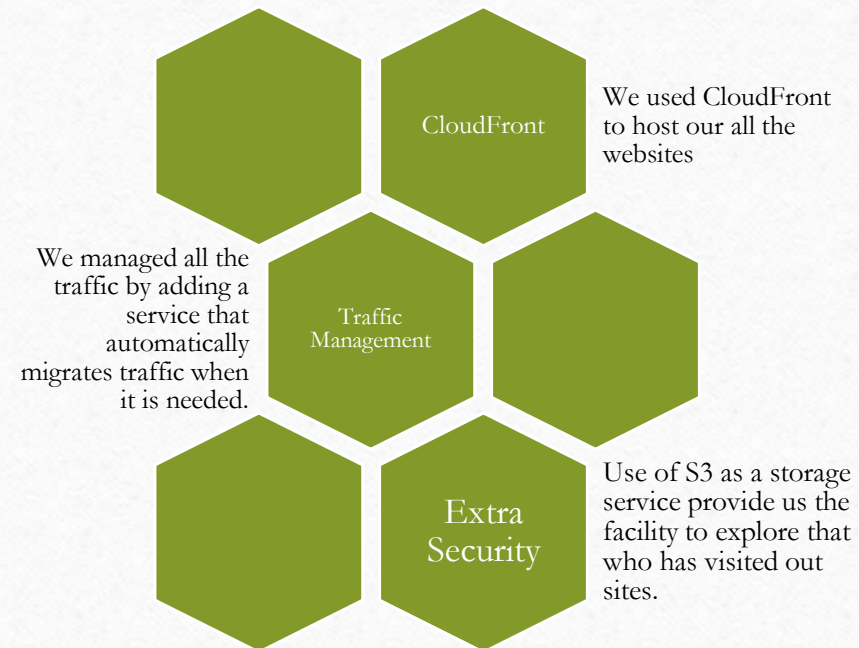
Click on the links below to access the websites:

[Website 1 : school.com](#)

[Website 2 : bakery.com](#)

[Website 3 : mosque.com](#)

[Website 4 : ITsolutions](#)





ITsolution.com



Mosque.com

All of our four sites

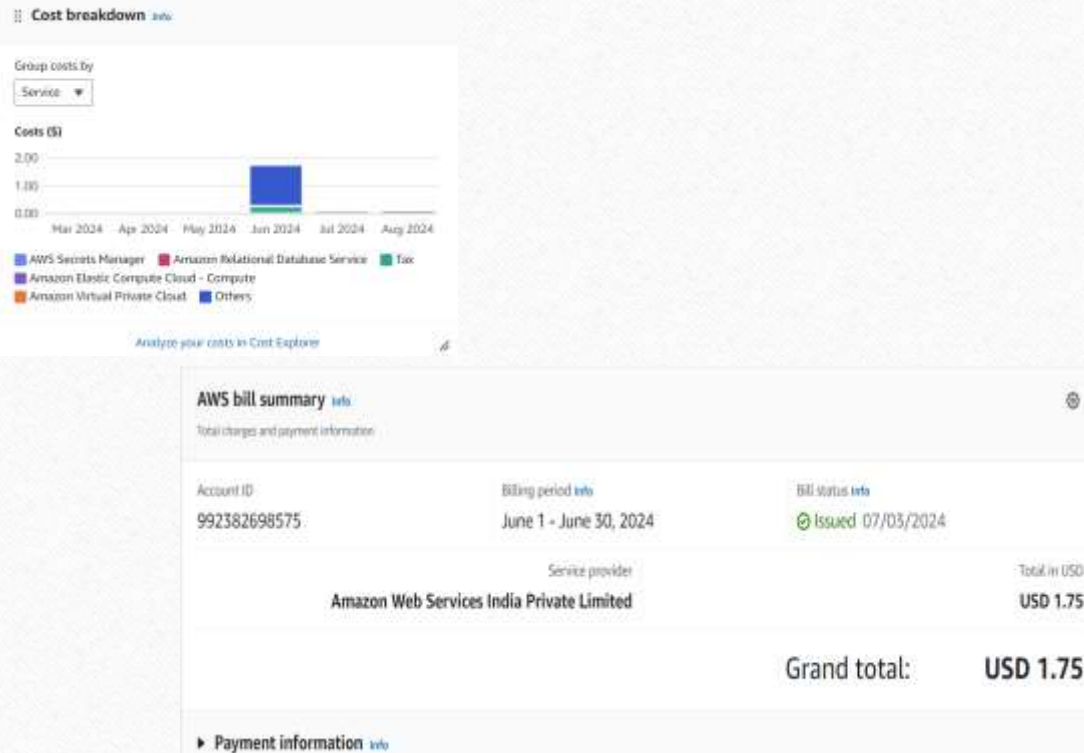
Bakery..com



Kider.com



Cost Management And Analysis



- AWS Budgets:** This allows you to set custom budgets for costs, usage, or reserved instances. You can define limits, and AWS will automatically notify you when your costs exceed those limits. For example, if you spend is nearing the budget, you can receive an email or SMS alert.
- Cost Explorer:** This tool helps analyze and visualize your usage patterns and spending over time. You can create custom reports to track your AWS usage, optimize your services, and forecast future costs. It's an essential part of cost control.

Future Scope

Scalable Infrastructure

Implementing auto-scaling for the server based on traffic, ensuring better resource management and user experience.

Performance Optimization

Integrating AWS CloudFront (CDN) and Elastic Load Balancing for improved content delivery and distribution of traffic across multiple servers if needed

Expansion to a Global Audience

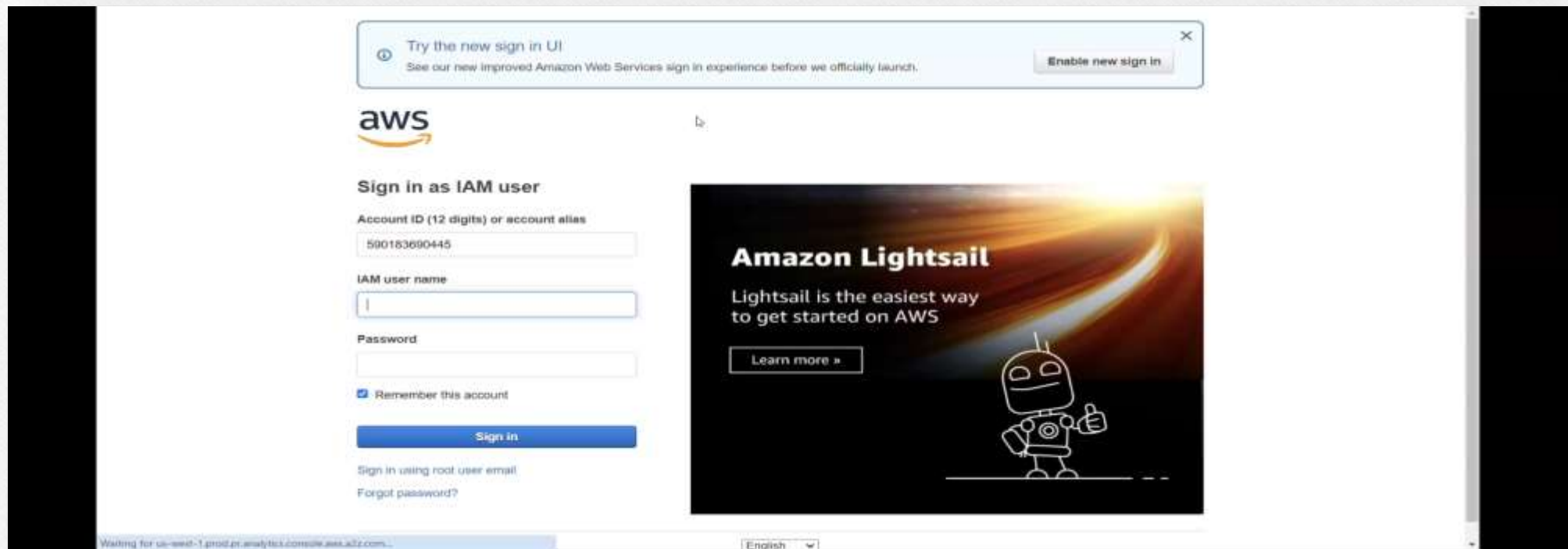
Replicating the multi-website hosting setup across multiple AWS regions for a global reach with low-latency performance

Support for New Technologies

Adopting new web technologies like Progressive Web Apps (PWA) or Web Assembly, which could require advanced server configurations



Implementation of our Project



The screenshot shows the AWS IAM sign-in interface. At the top, there is a notification banner about a new sign-in UI. Below this is the AWS logo. The main section is titled "Sign in as IAM user" and contains three input fields: "Account ID (12 digits) or account alias" with the value "590183690445", "IAM user name" (empty), and "Password" (empty). There is a "Remember this account" checkbox which is checked. A blue "Sign in" button is located below the password field. To the right of the sign-in form is a promotional banner for "Amazon Lightsail" with the text "Lightsail is the easiest way to get started on AWS" and a "Learn more" button. At the bottom of the page, there is a status bar showing "Waiting for ui-west-1.prod.pr.analytics.console.aws.amazon.com..." and a language dropdown set to "English".

Try the new sign in UI
See our new improved Amazon Web Services sign in experience before we officially launch.
Enable new sign in

aws

Sign in as IAM user

Account ID (12 digits) or account alias
590183690445

IAM user name
|

Password
|

☒ Remember this account

Sign in

Sign in using root user email
Forgot password?

Amazon Lightsail
Lightsail is the easiest way to get started on AWS
Learn more »

Waiting for ui-west-1.prod.pr.analytics.console.aws.amazon.com...
English

Thank You

