

- **Scenario:**

An online learning platform offers students access to PDF files, Word documents, and blog content. During peak times, like exam seasons, the platform faces sudden traffic surges, causing slow loading and downtime. Users experience delays in accessing study materials, impacting their learning experience. The company needs a scalable, reliable AWS-based solution for seamless content delivery, automated scaling, and real-time monitoring.

- **Problem Statement:**

Design an AWS architecture for an online learning platform that provides seamless access to PDF files, Word documents, and blog content. The platform should ensure low-latency content delivery, scalable compute resources for user interactions, and efficient management of content. Implement automated scaling and monitoring to ensure system stability during high-demand periods.

- **Objectives:**

1. To provide seamless access to PDF files, Word documents, and blog content with low-latency delivery.
2. To enable the system to automatically scale resources to handle traffic surges during peak periods.
3. To ensure high availability and minimize downtime using reliable AWS services.
4. To implement secure access controls, ensuring only authorized users can access content.
5. To monitor system performance in real-time and proactively detect and resolve issues.
6. To optimize resource utilization and reduce operational costs while maintaining performance and reliability.

- **Outcomes:**

1. **Scalable Infrastructure** – The platform can automatically adjust resources to handle varying traffic loads, ensuring smooth performance during peak periods.
2. **Improved Availability** – Minimal downtime and high availability, allowing students uninterrupted access to study materials.
3. **Low-Latency Content Delivery** – Fast response times for users globally, enhancing the learning experience.
4. **Enhanced Security** – Secure authentication and access control mechanisms protect content and user data.
5. **Optimized Cost Efficiency** – AWS services are utilized efficiently, reducing operational costs while maintaining performance.
6. **Real-Time Monitoring & Alerts** – System health and performance are continuously tracked, allowing for quick issue resolution and proactive maintenance.
7. **User Satisfaction & Engagement** – Reliable access to educational materials leads to improved student and instructor engagement.

- **Proposed AWS components:**

- a) **Amazon EC2** – Hosts the web application and backend services.
- b) **AWS Auto Scaling** – Dynamically adjusts compute resources based on traffic load.
- c) **Elastic Load Balancer (ELB)** – Distributes traffic across multiple instances to ensure high availability.
- d) **Amazon S3** – Stores and delivers PDFs, Word documents, and blog content.
- e) **Amazon CloudFront** – Caches and delivers content globally with low latency.
- f) **AWS IAM** – Manages access control and user permissions.
- g) **AWS WAF & Shield** – Protects against web attacks like SQL injection and DDoS threats.
- h) **AWS VPC** – Virtual Private Cloud

- **Solution:**

1. **Step-1: Create VPC (Virtual Private Cloud):**

Before deploying compute, storage, and database services, it's essential to set up a **Virtual Private Cloud (VPC)** and security configurations to ensure a secure and well-structured environment.

- I. Create VPC named as “myProject”

The screenshot shows the AWS VPC Dashboard. In the top right corner, there is a success message: "You successfully created vpc-0dbf44b43ce6cc995 / myProject". The main panel displays the details of the newly created VPC, including its ID (vpc-0dbf44b43ce6cc995), state (Available), and various configuration settings like Block Public Access (Off), DNS hostnames (Disabled), and Main route table (rtb-0667de94290c90e36). Below the main details, there is a "Resource map" section showing the VPC (myProject), Subnets (0), Route tables (1), and Network interfaces (0).

II. Setup Subnets (Public & Private), configure Internet Gateways, Configure Route Tables and network settings.

Created a Public Subnet and a Private Subnet.

The screenshot shows the AWS VPC dashboard with the following details:

- Subnets (2) Info:** A success message indicates 2 subnets have been created: subnet-00252643f633c4ba5 and subnet-05339bc987779f7df.
- Filter:** Subnet ID: subnet-00252643f633c4ba5, subnet-05339bc987779f7df.
- Table Headers:** Name, Subnet ID, State, VPC, Block Public..., IPv4 CIDR.
- Subnets:**
 - Public-Subnet-1:** Subnet ID: subnet-00252643f633c4ba5, Available, vpc-0dbf44b43ce6cc995 | myPr..., Off, 10.0.1.0/24
 - Private-Subnet-1:** Subnet ID: subnet-05339bc987779f7df, Available, vpc-0dbf44b43ce6cc995 | myPr..., Off, 10.0.2.0/24
- Select a subnet:** Buttons for Create, Edit, and Delete.

Attached them to the VPC myProject that we had created earlier.

The screenshot shows the AWS VPC dashboard with the following details:

- Internet gateway:** igw-027056031b1eefc5a successfully attached to vpc-0dbf44b43ce6cc995.
- Details:**
 - Internet gateway ID: igw-027056031b1eefc5a
 - State: Attached
 - VPC ID: vpc-0dbf44b43ce6cc995 | myProject
 - Owner: 550239528263
- Tags:**
 - Key: Name, Value: MyInternetGateway

aws | Search [Alt+S] United States (N. Virginia) vclabs/user3771000=rohan.mishra23@ipu.in @ 5502-3952-8263

VPC > Route tables > rtb-009e5495781679c52 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
Public-Subnet-1	subnet-00252643f633c4ba5	10.0.1.0/24	-	Main (rtb-0667de94290c90e36)	
Private-Subnet-1	subnet-05339bc9877779f7df	10.0.2.0/24	-	Main (rtb-0667de94290c90e36)	

Selected subnets

- subnet-00252643f633c4ba5 / Public-Subnet-1

Cancel Save associations

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

VPC > Route tables > rtb-009e5495781679c52

rtb-009e5495781679c52 / Public-Route-Table

You have successfully updated subnet associations for rtb-009e5495781679c52 / Public-Route-Table.

Details

Route table ID rtb-009e5495781679c52	Main No	Explicit subnet associations subnet-00252643f633c4ba5 / Public-Subnet-1	Edge associations -
VPC vpc-0dbf44b43ce6cc995 myProject	Owner ID 550239528263		

Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	igw-027056051b1eefc5a	Active	No
10.0.0.0/16	local	Active	No

Both Actions

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws | Search [Alt+S] United States (N. Virginia) vclabs/user3771000=rohan.mishra23@ipu.in @ 5502-3952-8263

VPC dashboard <

EC2 Global View Filter by VPC

Virtual private cloud

- Your VPCs
- Subnets
- Route tables**
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- NAT gateways
- Peering connections

Security

- Network ACLs
- Security groups

PrivateLink and Lattice

- Getting started Updated
- Endpoints Updated

Create VPC

Your VPCs (1/3) info

Last updated 3 minutes ago

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
-	vpc-09fd33d434805a8da	Available	Off	172.31.0.0/16	-
Work VPC	vpc-0075218acb3170c23	Available	Off	10.0.0.0/16	-
myProject	vpc-0dbf44b43ce6cc995	Available	Off	10.0.0.0/16	-

vpc-0dbf44b43ce6cc995 / myProject

Details Resource map CIDs Flow logs Tags Integrations

Details

VPC ID vpc-0dbf44b43ce6cc995	State Available	Block Public Access Off	DNS hostnames Disabled
DNS resolution Enabled	Tenancy default	DHCP option set dopt-05a56071b1eebb653	Main route table rtb-0667de94290c90e36
Main network ACL -	Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool -
IPv6 CIDR (Network border group) -	Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups -	Owner ID 550239528263

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

III. Setup Security Groups and Network ACL:

Creating Security Groups for web server EC2 instance. It acts as virtual firewall for the AWS resources. It defines what kind of network traffic is allowed to and from instance.

Here, we are creating Network ACL:

Network ACLs (NACLs) control traffic at the subnet level. While Security Groups work at the instance level, NACLs provide additional security for the overall network.

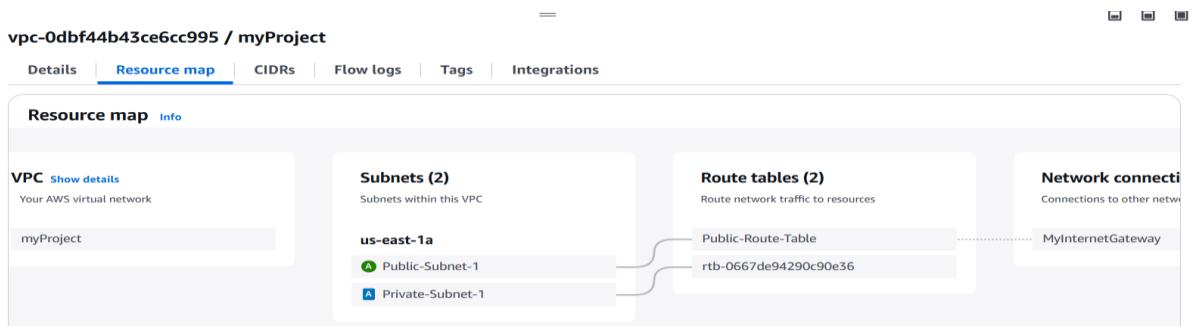
The screenshot shows the 'Create network ACL' page in the AWS VPC console. The 'Name' field is set to 'Public-Subnet-NACL'. The 'VPC' dropdown is set to 'vpc-0dbf44b43ce6cc995 (myProject)'. Under 'Tags', a single tag 'Name: Public-Subnet-NACL' is added. The 'Create network ACL' button is visible at the bottom right.

The screenshot shows the 'Network ACLs (1/4)' page in the AWS VPC console. A success message indicates rules were updated for 'Public-Subnet-NACL'. The table lists four Network ACLs, with 'Public-Subnet-NACL' selected. The 'Outbound rules' tab is active, showing two rules: one allowing all traffic to 0.0.0.0/0 with 'Allow' status, and another allowing all traffic to 0.0.0.0/0 with 'Deny' status. The 'Edit outbound rules' button is located at the top right of the rules table.

Name	Network ACL ID	Associated with	Default	VPC ID	Inbound
acl-0cccd702644209be5	subnet-0e9735e41c9dcc3c / Work Public Sub...	Yes	vpc-0075218acb3170c23 / Work VPC	2 Inb	
acl-0402e95342010e156	6 Subnets	Yes	vpc-09fd33d434805a8da	2 Inb	
acl-013ea442c8a03d8a7	2 Subnets	Yes	vpc-0dbf44b43ce6cc995 / myProject	2 Inb	
Public-Subnet-NACL	acl-0ec3369b9daa08f6b	-	No	vpc-0dbf44b43ce6cc995 / myProject	4 Inb

IV. VPC resource map:

It is the resource map of an AWS Virtual Private Cloud (VPC) named "**myProject**". It includes two subnets within the **us-east-1a** availability zone: a **Public-Subnet-1** for internet-facing resources, and a **Private-Subnet-1** for internal services. These subnets are associated with a **Public Route Table**, which routes internet traffic through an **Internet Gateway** named **MyInternetGateway**, enabling resources in the public subnet to access and be accessed from the internet. This setup forms the foundation for deploying a secure and scalable web application in AWS.



2. Step 2: Launching EC2 Instance:

I. Launch an Instance:

In this step, an EC2 instance named **MyProjectEC2** is being created to serve as the web server for hosting the website.

NOTE: Used t2.micro service as it is a free tier service. If I have to use the paid service, I would have used **m5.large or t3.large** as it is suitable to manage high-traffic.

The screenshot shows the AWS EC2 Launch Instance wizard. The process is at the 'Launch an instance' step. In the 'Name and tags' section, the instance name is set to 'MyProjectEC2'. In the 'Application and OS Images (Amazon Machine Image)' section, 'Amazon Linux 2023 AMI' is selected. On the right, the 'Summary' section shows the configuration: 1 instance, t2.micro virtual server type, EC2-Security-Group, and 1 volume (8 GiB). A callout box highlights the 'Free tier' information: 'In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.'

Instance type

t2.micro
Family: t2 - 1 vCPU, 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

Additional costs apply for AMIs with pre-installed software

Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required
vokey [Create new key pair](#)

Network settings Info

VPC - required | [Info](#)
vpc-0dbf44b43ce6cc995 (myProject)
10.0.0.0/16

Subnet | [Info](#)
subnet-00252643f633c4ba5 Public-Subnet-1
VPC: vpc-0dbf44b43ce6cc995 Owner: 550239528263 Availability Zone: us-east-1a Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24

Auto-assign public IP | [Info](#)
Enable

Summary
Number of instances: 1
Virtual server type (instance type): t2.micro
Firewall (security group): EC2-Security-Group
Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

[Launch instance](#) [Preview code](#)

Network settings Info

VPC - required | [Info](#)
vpc-0dbf44b43ce6cc995 (myProject)
10.0.0.0/16

Subnet | [Info](#)
subnet-00252643f633c4ba5 Public-Subnet-1
VPC: vpc-0dbf44b43ce6cc995 Owner: 550239528263 Availability Zone: us-east-1a Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24

Auto-assign public IP | [Info](#)
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
 Create security group Select existing security group

Common security groups | [Info](#)
Select security groups
EC2-Security-Group sg-09612f95a3699c874 X
VPC: vpc-0dbf44b43ce6cc995

Compare security group rules

Advanced network configuration

Network interface 1
Device index: [Info](#) Network interface: [Info](#) Description: [Info](#)
0 New interface

Summary
Number of instances: 1
Virtual server type (instance type): t2.micro
Firewall (security group): EC2-Security-Group
Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

[Launch instance](#) [Preview code](#)

II. Connecting EC2 instance

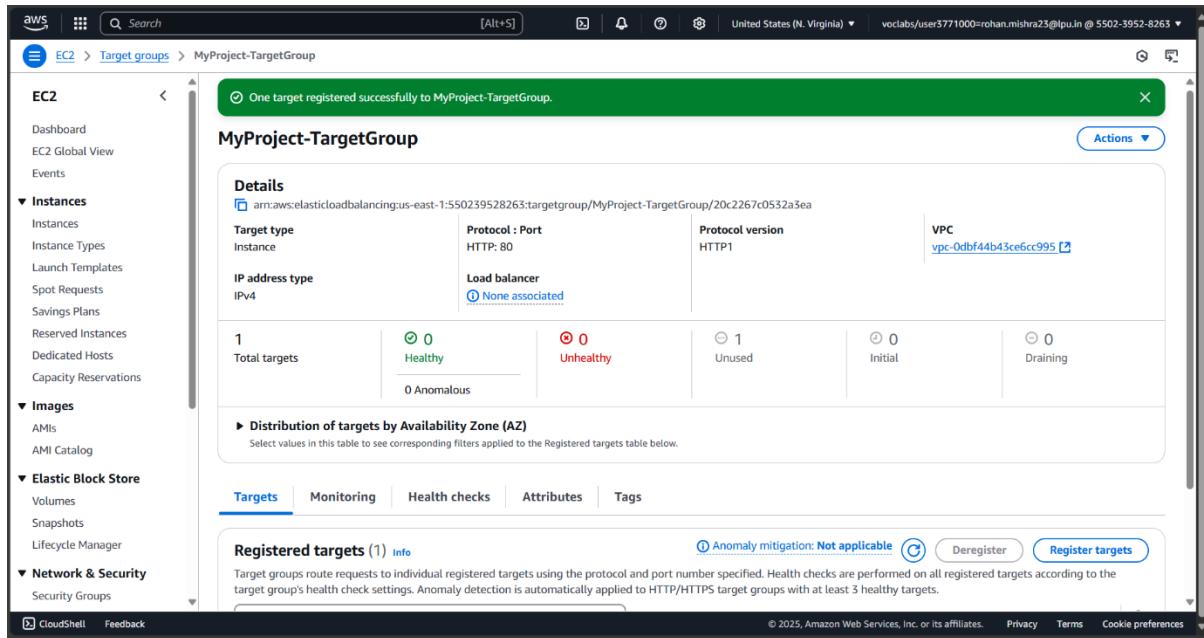
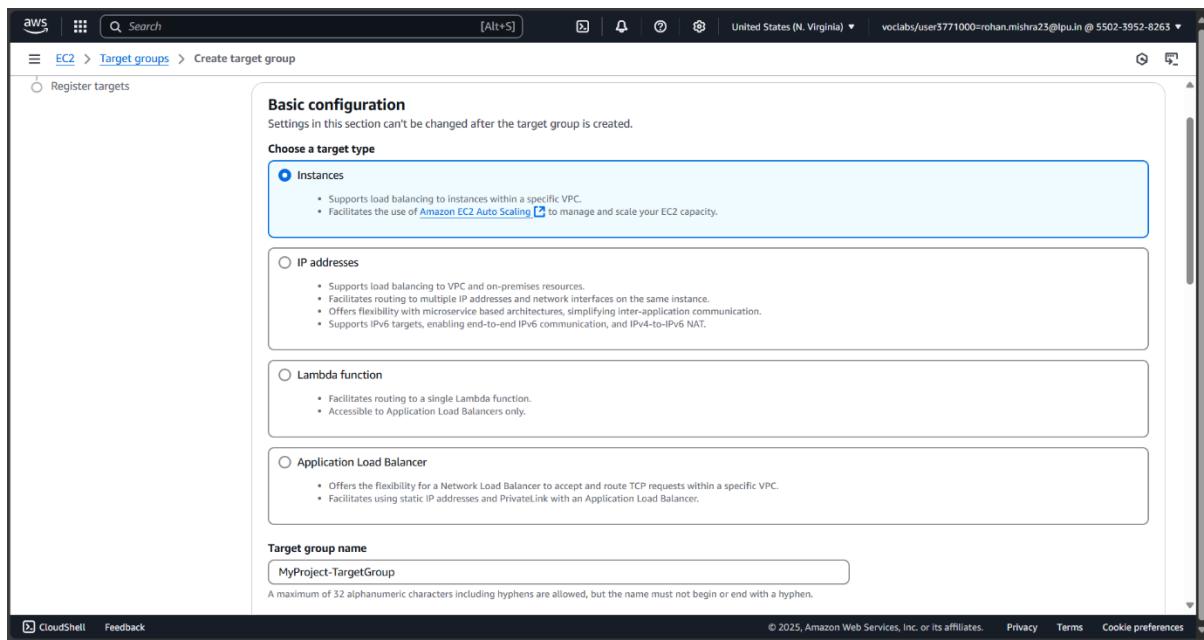
i-061d465c3da8bc920 (MyProjectEC2)

PublicIPs: 3.91.100.147 PrivateIPs: 10.0.1.174

Establishing Connection ...

3. Step 3: Amazon ELB (Elastic Load Balancer):

- I. In this step, a **Target Group** named **MyProject-TargetGroup** is being created as part of setting up an **Elastic Load Balancer (ELB)** for the website. The target type selected is **Instances**, which means the load balancer will route incoming traffic to one or more EC2 instances within the VPC.



- II. In this step, an Application Load Balancer (ALB) named **MyProject-ALB** has been successfully created. The ALB is **internet-facing**, meaning it is designed to accept and distribute incoming web traffic from the internet to EC2 instances across multiple availability zones. This enhances the website's **fault tolerance and availability**, ensuring it remains accessible even if one availability zone experiences issues. The load balancer automatically distributes incoming traffic to the registered targets in the target group, helping to balance the load and improve performance. The DNS name provided by AWS can now be used to access the website through the ALB.

4. Step 4: Configure AWS Auto-Scaling:

- I. Create Launch Template:

In this step, a **Launch Template** is being created to automate the deployment of EC2 instances for the website. The user data script provided installs and starts the Apache web server (`httpd`) on each new instance, and sets a custom welcome message ("Welcome to My Learning Platform") as the homepage.

II. Create Auto Scaling Group:

In this step, an **Auto Scaling Group** named **MyProject-AutoScaling** is being reviewed before creation. It uses a previously defined **Launch Template** (MyTemplate) that includes user data for setting up Apache and serving the website content. The Auto Scaling Group is configured to launch EC2 instances across multiple **Availability Zones** within the specified **VPC**, enhancing both **high availability** and **fault tolerance**. With this setup, AWS can automatically increase or decrease the number of instances based on traffic demand, ensuring the website remains responsive during peak loads and cost-effective during low usage.

Screenshot of the AWS CloudShell interface showing the creation of an Auto Scaling group. The user is on Step 1: Choose launch template.

Step 1: Choose launch template

- Choose launch template
- Choose instance launch options
- Integrate with other services
- Configure group size and scaling
- Add notifications
- Add tags
- Review

Group details

Auto Scaling group name: MyProject-AutoScaling

Launch template

Launch template	Version	Description
MyTemplate [L] lt-0c74f9b58c5815b4b	Default	For the MyProject

Step 2: Choose instance launch options

Network

VPC: vpc-0dbf44b43ce6cc995 [L]

Availability Zone	Subnet	Subnet CIDR range
us-east-1a	subnet-00252643f633c4ba5 [L]	10.0.1.0/24
us-east-1a	subnet-05339bc987779f7df [L]	10.0.2.0/24
us-east-1b	subnet-0fda8107e0714178a [L]	10.0.3.0/24

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS CloudShell interface showing the creation of an Auto Scaling group. The user is on Step 3: Integrate with other services.

Instance type requirements

This Auto Scaling group will adhere to the launch template.

Step 3: Integrate with other services

Load balancing

Load balancer 1

Name	Type	Target group
MyProject-ALB [L]	Application/HTTP	MyProject-TargetGroup [L]

VPC Lattice integration options

VPC Lattice target groups: -

Application Recovery Controller (ARC) zonal shift

ARC zonal shift: Disabled

Health checks

Health check type	Health check grace period
None	None

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS CloudShell interface showing the created Auto Scaling group.

Auto Scaling groups (1/1) Info

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
MyProject-AutoScaling	MyTemplate Version Default	0	Updating capacity...	1	1	1	us-east-1a, us-east-1b

Auto Scaling group: MyProject-AutoScaling

Details Integrations - new Automatic scaling Instance management Instance refresh Activity Monitoring

MyProject-AutoScaling Capacity overview

Desired capacity	Scaling limits (Min - Max)	Desired capacity type	Status
1	1 - 1	Units (number of instances)	Updating capacity

Date created: Mon Mar 31 2025 19:20:40 GMT+0530 (India Standard Time)

Launch template

Launch template	AMI ID	Instance type	Owner
MyTemplate [L]	ami-0c74f9b58c5815b4b	t2.micro	rohan.mishra23@ipu.in

5. Step 5: Create Amazon S3 Bucket:

- I. In this step, an **Amazon S3 bucket** named **MyProject-content** is being created to store static content such as images, documents, or any other files used by the website. The bucket uses the **General-purpose** bucket type, which supports a variety of storage classes and ensures high availability by replicating data across multiple availability zones. **ACLs are disabled** for improved security, meaning access is controlled strictly through bucket policies. This setup is ideal for hosting static website assets or integrating with services like CloudFront for fast and secure content delivery.

The screenshot shows the 'Create bucket' wizard in the AWS S3 console. In the 'General configuration' section, the 'AWS Region' is set to 'US East (N. Virginia) us-east-1'. The 'Bucket type' dropdown is open, with 'General purpose' selected (indicated by a blue border). A tooltip explains that general purpose buckets are the original S3 bucket type, supporting a mix of storage classes across multiple Availability Zones. The 'Bucket name' field contains 'MyProject-content'. Below it, a note specifies bucket names must be 3 to 63 characters and unique. The 'Copy settings from existing bucket - optional' section shows a 'Choose bucket' button and a note about copied settings. In the 'Object Ownership' section, 'ACLs disabled (recommended)' is selected (blue border), indicating all objects are owned by the account and access is controlled via bucket policies.

The screenshot shows the 'Buckets' page in the AWS S3 console. A green success message at the top states 'Successfully created bucket "myproject-content"'. Below it, an 'Account snapshot - updated every 24 hours' section provides storage usage information. The main table lists 'General purpose buckets (1/1)'. The single entry is 'myproject-content', which was created on March 31, 2025, at 19:28:49 UTC+05:30. The table includes columns for Name, AWS Region, IAM Access Analyzer, and Creation date. Action buttons for each row include 'View details', 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://myproject-content	1 file, 20.2 KB (100.00%)	0 files, 0 B (0%)

Files and folders Configuration

Files and folders (1 total, 20.2 KB)

Name	Folder	Type	Size	Status	Error
CA2 INT330 updated.docx	-	application/vnd.openxmlform...	20.2 KB	Succeeded	-

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Edit bucket policy Info

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

Bucket ARN arn:aws:s3:::myproject-content

Policy

```
1 ▼ {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Principal": {
7         "Service": "cloudfront.amazonaws.com"
8       },
9       "Action": "s3:GetObject",
10      "Resource": "arn:aws:s3:::mylearningplatform-content/*",
11      "Condition": {
12        "StringEquals": {
13          "AWS:SourceArn": "arn:aws:cloudfront::550239528263:distribution/E3BKUVUM0K8J7D"
14        }
15      }
16    }
17  ]
18}
```

Policy examples Policy generator

Edit statement Remove

Add actions Choose a service Q Filter services

Included S3

Available AI Operations AMP API Gateway API Gateway V2

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon S3

General purpose buckets

- Directory buckets
- Table buckets
- Access Grants
- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

- Dashboards
- Storage Lens groups
- AWS Organizations settings

Feature spotlight [1]

▶ AWS Marketplace for S3

Account snapshot - updated every 24 hours All AWS Regions View Storage Lens dashboard

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

General purpose buckets (1 / 1) info All AWS Regions

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
myproject-content	US East (N. Virginia) us-east-1	View analyzer for us-east-1	March 31, 2025, 19:28:49 (UTC+05:30)

Copy ARN Empty Delete Create bucket

Find buckets by name

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- II. After setting up the S3 bucket for storing static content, a **CloudFront distribution** (ID: **EDIIJDS1H613X**) was created to efficiently deliver this content to users around the world. CloudFront acts as a global **Content Delivery Network (CDN)**, caching data at edge locations to reduce latency and improve load times. By connecting the distribution to the S3 bucket, the website ensures faster and more secure access to static files such as images, videos, and documents.

The screenshot shows the AWS CloudFront 'Distributions' page with the distribution ID 'EDIIJDS1H613X' selected. The 'General' tab is active. The 'Details' section shows the distribution domain name as 'd10u04s0s2elak.cloudfront.net'. The ARN is listed as 'arn:aws:cloudfront::550239528263:distribution/EDIIJDS1H613X'. The 'Last modified' status is 'Deploying'. The 'Settings' section includes fields for 'Description' (empty), 'Price class' (set to 'Use all edge locations (best performance)'), and 'Supported HTTP versions' (set to 'HTTP/2, HTTP/1.1, HTTP/1.0'). The 'Alternate domain names' field is empty. On the right, there are sections for 'Standard logging' (off), 'Cookie logging' (off), and 'Default root object' (empty). A 'Create staging distribution' button is visible. The bottom navigation bar includes links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.