

INTERNSHIP PROJECT – CLOUD COMPUTING.

For my internship assignment at Animemangatoon.com, I developed and deployed a cloud-based web application inspired by 'The 05 Best Fantasy Manhwa You Must Read Now.' Utilizing **Amazon Web Services (AWS)**, I ensured that the application was highly available, scalable, and secure. AWS S3 was used for static web hosting, making it cost-effective and efficient for serving content. Additionally, the web app was optimized to handle high traffic using **Auto-Scaling**, providing a global distribution network for faster content delivery.

With AWS's robust suite of services, the project was designed with front-end technologies, avoiding the use of backend frameworks while still ensuring a responsive and reliable experience. The app was version-controlled and deployed via **GitHub**, following cloud best practices for easy deployment and accessibility.

Github link :- [Prashant-8113/cloud-assignment - GitHub](https://github.com/Prashant-8113/cloud-assignment)

Steps for complete site setup and deployment.

1. Project Planning & Research

- Researched the content for the web application based on 'The 05 Best Fantasy Manhwa' You Must Read Now.
- Focused on creating a front-end using HTML, CSS technologies.

2. Setting Up the AWS Environment

- Created an **AWS account** and set up the necessary permissions.
- Planned for a cloud-based deployment using **Amazon EC2, S3, Elastic Load Balancer**, and **Auto Scaling**.

3. Developing the Web Application

- Built the front-end of the application using HTML, CSS.
- Organized all files (HTML, CSS, images) in preparation for deployment on EC2.
- Site is fulfilled with all the requirements as per animemangatoon.com specified.

4. Configuring EC2 for Deployment

- Launched an **Amazon EC2** instance (Virtual Machine) to serve as the host for the application.

- Installed a web server (like Apache or Nginx) on the EC2 instance to serve the static web content.
- Uploaded the front-end files (HTML, CSS) to the EC2 instance using **SSH**.
- Allowed HTTP and HTTPS access for web access.

Launch template name and description

Launch template name - *required*

web-template

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

1

Max 255 chars

Type	Info	Protocol	Info	Port range	Info
HTTP		TCP		80	
Source type	Info	Source	Info	Description - optional	Info
Anywhere		0.0.0.0/0		e.g. SSH for admin desktop	
Security group rule 3 (TCP, 443, 0.0.0.0/0)					
Type	Info	Protocol	Info	Port range	Info
HTTPS		TCP		443	
Source type	Info	Source	Info	Description - optional	Info
Anywhere		0.0.0.0/0		e.g. SSH for admin desktop	

Remove

Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...read more
ami-0ef29ab52f72213b

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Region in which

EC2 > ... > Launch instance from template

Launch instance from template

Launching from a template allows you to launch from an instance configuration that you would have saved in the past. These saved configurations can be reused and shared with other users to standardize launches across an organisation.

Choose a launch template

Source template

web-template
ID: lt-09aef370e4bb2d6c7

1 (Default)
1

Instance details

Your instance details are listed below. Any fields that are not specified as part of the configuration below will use the template or default values for those fields. Ensure that you have permissions to override these parameters or your instance launch will fail.

Application and OS Images (Amazon Machine Image) Info

Summary

Number of instances Info
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...read more
ami-0ef29ab52f72213b

Virtual server type (instance type)
t2.micro

Firewall (security group)
my-sec

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Region in which

Cancel Launch instance

- Code for ec2 deployment.

```
[root@ip-172-31-34-196 html]# history
 1  yum install httpd -y
 2  systemctl restart httpd
 3  sudo yum install nginx -y
 4  systemctl start nginx
 5  systemctl enable nginx
 6  yum install git -y
 7  cd /var/www/html/
 8  git clone https://github.com/Prashant-8113/cloud-assignment.git
 9  mv /var/www/html/your-repo/* /var/www/html/
10  sudo chown -R ec2-user:ec2-user /var/www/html/
11  sudo chmod -R 755 /var/www/html/
12  history
[root@ip-172-31-34-196 html]#
```

5. Load Balancer Configuration

- Configured an **Elastic Load Balancer (ELB)** to distribute incoming traffic across multiple EC2 instances, ensuring high availability and fault tolerance.
- Set up the ELB to route requests to your EC2 instance hosting the web application.

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

☐ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

☐ Attach to an existing load balancer
Choose from your existing load balancers.

☒ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to a new load balancer

Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, [visit the Load Balancing console](#).

☒ Application Load Balancer
HTTP, HTTPS

☐ Network Load Balancer
TCP, UDP, TLS

Load balancer name
Name cannot be changed after the load balancer is created.

my-group-1

Load balancer scheme
Scheme cannot be changed after the load balancer is created.

☐ Internal

☒ Internet-facing

[EC2](#) > [Target groups](#)

Target groups (1/1) [Info](#)

Filter target groups

< 1 >

☒

Name

☐

ARN

☐

Port

☐

Protocol

☐

Target type

☐

Load balancer

☒

[my-target-group](#)

arn:aws:elasticloadbalanci...

80

HTTP

Instance

my-alb

Target group: my-target-group

Registered targets (1) [Info](#)

Anomaly mitigation: **Not applicable**

Deregister

Register targets

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

< 1 >

☐

Instance ID

☐

Name

☐

Port

☐

Zone

☐

Health status

☐

Health status details

☐

Launch...

☐

[i-01ec40e78d9bde505](#)

web-app

80

ap-northeast-1a

Initial

Target registration is i...

October 9...

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks

Anomaly mitigation: **Always enabled**

Additional health check types - optional [Info](#)

☒ Turn on Elastic Load Balancing health checks

Recommended

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Anomaly mitigation: **EC2 Auto Scaling will start to detect and act on health checks performed by Elastic Load Balancing. To avoid unexpected terminations, first verify the settings of these health checks in the [Load Balancer console](#)**

☐ Turn on VPC Lattice health checks

VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

☐ Turn on Amazon EBS health checks

EBS monitors whether an instance's root volume or attached volume stalls. When it reports an unhealthy volume, EC2 Auto Scaling can replace the instance on its next periodic health check.

Health check grace period [Info](#)

This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

200

seconds

6. Enabling Auto Scaling for EC2

- Configured **AWS Auto Scaling** to automatically add or remove EC2 instances based on traffic demand.
- This ensures that the application can scale dynamically, handling high traffic loads during peak times and reducing costs during low-traffic periods.

Automatic scaling - optional

Choose whether to use a target tracking policy [Info](#)

You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

☐ **No scaling policies**
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

☒ **Target tracking scaling policy**
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name

Target Tracking Policy

Metric type [Info](#)

Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization ▼

Target value

40

Instance warmup [Info](#)

300 seconds

(Optional step for giving sms system for getting real-time information.)

Add notifications - optional [Info](#)

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

Add notification

Cancel Skip to review Previous **Next**

- Real time monitoring and autoscaling.

Failed to start the instance i-01ec40e78d9bde505

You have requested more vCPU capacity than your current vCPU limit of 1 allows for the instance bucket that the specified instance type belongs to. Please visit <http://aws.amazon.com/contact-us/ec2-request> to request an adjustment to this limit.

Instances (1/2) Info

Last updated less than a minute ago

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv
<input checked="" type="checkbox"/>	web-app	i-01ec40e78d9bde505	Stopped	t2.micro	-	View alarms +	ap-northeast-1a	-
<input type="checkbox"/>		i-05cb47bcc42b7e9ae	Running	t2.micro	2/2 checks passed	View alarms +	ap-northeast-1a	ec2-18-18

EC2 > Target groups > my-target-group

my-target-group

Actions

Details

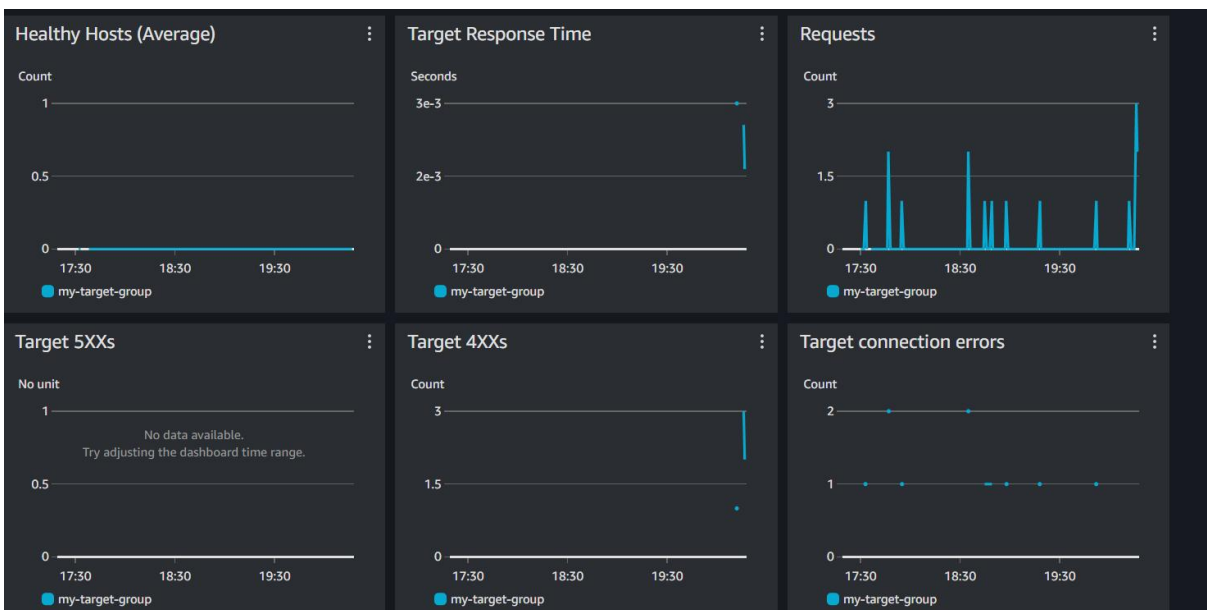
arn:aws:elasticloadbalancing:ap-northeast-1:058264234387:targetgroup/my-target-group/5a47a2b578345b48

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-007679ae818aa0e23
IP address type	Load balancer		
IPv4	my-alb		

2	0 Healthy	1 Unhealthy	1 Unused	0 Initial	0 Draining
Total targets	0 Anomalous				

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.



7. Setting Up Amazon S3 for File Storage

- Used **Amazon S3** to store large static assets, such as images, for the website.
- Configured the web server on EC2 to retrieve these assets from the S3 bucket to reduce load on the server and increase performance.

General configuration

AWS Region
Asia Pacific (Tokyo) ap-northeast-1

Bucket type [Info](#)

☒ **General purpose**
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

cloud-bucket

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

☐ **Block *all* public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region
Asia Pacific (Tokyo) ap-northeast-1

Bucket type Info

☒ **General purpose**
 Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**
 Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info

cloud-bucket-intern

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

- **Public Access to S3 Bucket:** To enable public read access for all objects in the S3 bucket, configure the bucket's permissions by applying a policy that allows the :- s3:GetObject action to all users (Principal: *).

This ensures that anyone on the internet can access and download files from the bucket.

Bucket ARN

arn:aws:s3:::cloud-bucket-intern

Policy

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "PublicReadGetObject",
6       "Effect": "Allow",
7       "Principal": "*",
8       "Action": [
9         "s3:GetObject"
10      ],
11      "Resource": [
12        "arn:aws:s3:::cloud-bucket-intern/*"
13      ]
14    }
15  ]
16 }
17

```

- All the images and other static code assets added to S3 bucket

Files and folders (8 Total, 215.8 KB)			
All files and folders in this table will be uploaded.			
<input type="text" value="Find by name"/> < 1 >			
<input type="checkbox"/>	Name	Folder	Type
<input type="checkbox"/>	attack on titan.jpg	images/	image/jpeg
<input type="checkbox"/>	death note.jpeg	images/	image/jpeg
<input type="checkbox"/>	demon slayer.jpeg	images/	image/jpeg
<input type="checkbox"/>	my hero academia.jpeg	images/	image/jpeg
<input type="checkbox"/>	naruto.jpeg	images/	image/jpeg
<input type="checkbox"/>	one piece.jpeg	images/	image/jpeg
<input type="checkbox"/>	sword art online.jpeg	images/	image/jpeg
<input type="checkbox"/>	tokyo ghoul.jpeg	images/	image/jpeg

8. Version Control with GitHub

- Managed the project's code through **GitHub**, enabling version control and easy collaboration.
- Uploaded the web app to a GitHub repository and linked it for submission.
- Github link :- [Prashant-8113/cloud-assignment - GitHub](https://github.com/Prashant-8113/cloud-assignment)

9. Testing & Validation

- Conducted thorough testing of the web application on the EC2 instance, ensuring that it responds properly under different load conditions.
- Verified the performance of the Auto Scaling and Load Balancer configurations.
- Web page link :- <http://cloud-bucket-intern.s3-website-ap-northeast-1.amazonaws.com/>

