MAJOR -2 PROJECT

END TERM REPORT

For

Designing a Scalable and Fault tolerant Web application

Submitted By

Specialization	SAP ID	Name
CCVT	500083440	Aahna Pandey
CCVT	500084089	Aarushi Arora
CCVT	500083149	Aditya Pilani

Department of Systemics

School Of Computer Science

UNIVERSITY OF PETROLEUM & ENERGY STUDIES,

DEHRADUN- 248007. Uttarakhand

Dr.Ambika Aggarwal
Project Guide

Dr. Neelu J. Ahuja
Cluster Head

School of Computer Science
University of Petroleum & Energy Studies, Dehradun

Synopsis Report

1. Project Title

Designing a Scalable and Fault tolerant Web application

2. Abstract

Web applications must provide outstanding performance and user experience, as well as high availability and resilience in the face of unforeseen difficulties, in order to survive in the current digital landscape. This calls for a fault-tolerant and scalable design methodology. A web application is said to be scalable if it can adapt to increases in user demand or data volume without experiencing a decline in performance. Contrarily, fault tolerance ensures that a system will continue to function even in the event of component failure or outside disturbances. In addition to meeting present needs, this presentation will cover future-proofing techniques and best practices for building web applications against growing loads and possible system failures by utilising AWS tools and resources. Through the strategic application of AWS's scalable resources and fault-tolerant architectures, developers can create web applications that not only withstand the test of time but also evolve with the technological landscape, ensuring that user demands are met with unwavering reliability and performance.

3. Introduction

- The success of a web application relies on its ability to remain operational and accessible, even when faced with unexpected challenges.
- Scalability and fault tolerance are pivotal concepts in achieving this goal.

- Scalability involves the application's capacity to adapt and perform well under increasing workloads.
- Fault tolerance ensures the application's availability even in the presence of failures or disruptions.
- Amazon Web Services (AWS) emerges as a cloud computing platform that offers a comprehensive suite of services tailored for contemporary web application development.
- The objective of this presentation is to explore how AWS resources and tools can be leveraged to design web applications that exhibit both scalability and fault tolerance.
- By effectively utilizing AWS services, developers can craft applications that seamlessly adjust to changing user demands and technological landscapes

4. Problem Statement

The difficulty lies in designing online applications that easily scale to user traffic while ensuring resilient availability in a digital environment marked by evolving user needs and technology uncertainties. Using the tactical capabilities of AWS services, this presentation discusses the requirement for designing such scalable and fault-tolerant web applications.

.

5. Objectives

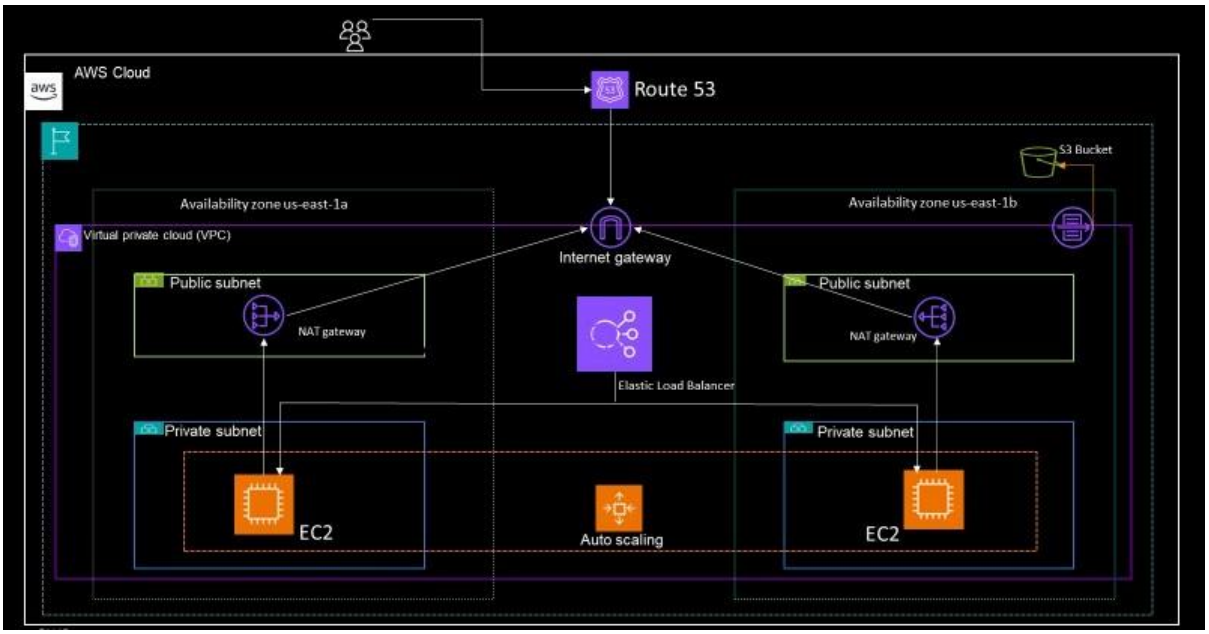
1. **Understand Principles:** Explore the fundamental concepts of scalability and fault tolerance in web application design using AWS services.
2. **AWS Tool Proficiency:** Gain proficiency in leveraging AWS services to create web applications that can scale dynamically and remain resilient.

3. **Architectural Strategies:** Discover effective architectural strategies for building applications capable of handling varying loads and ensuring fault tolerance.
4. **Real-World Application:** Learn through practical examples and case studies how to implement scalable and fault-tolerant design principles using AWS services.

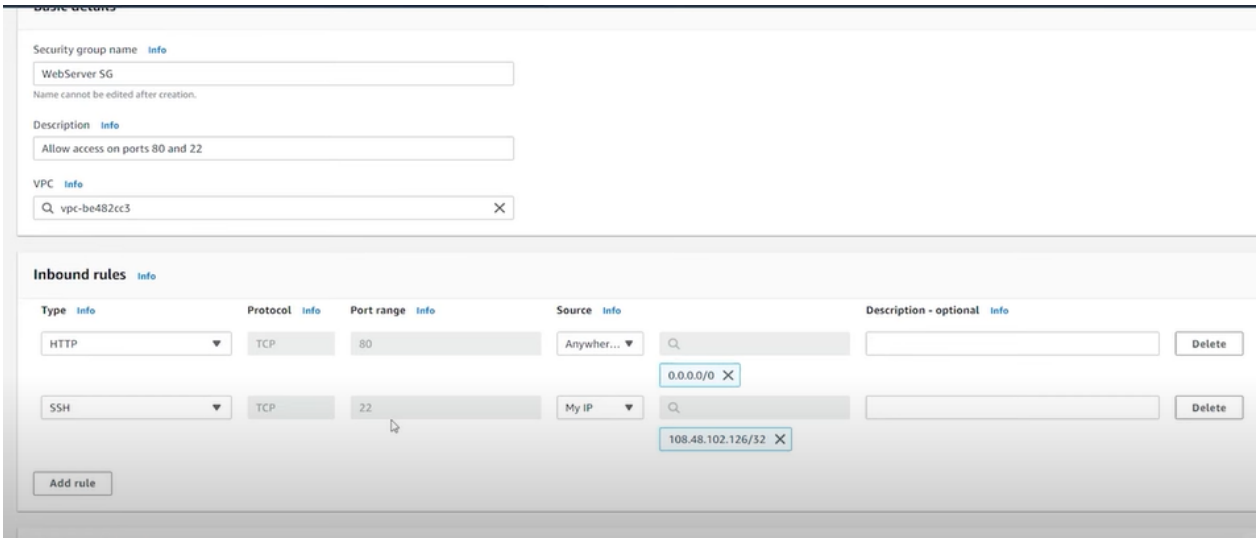
6. Methodology

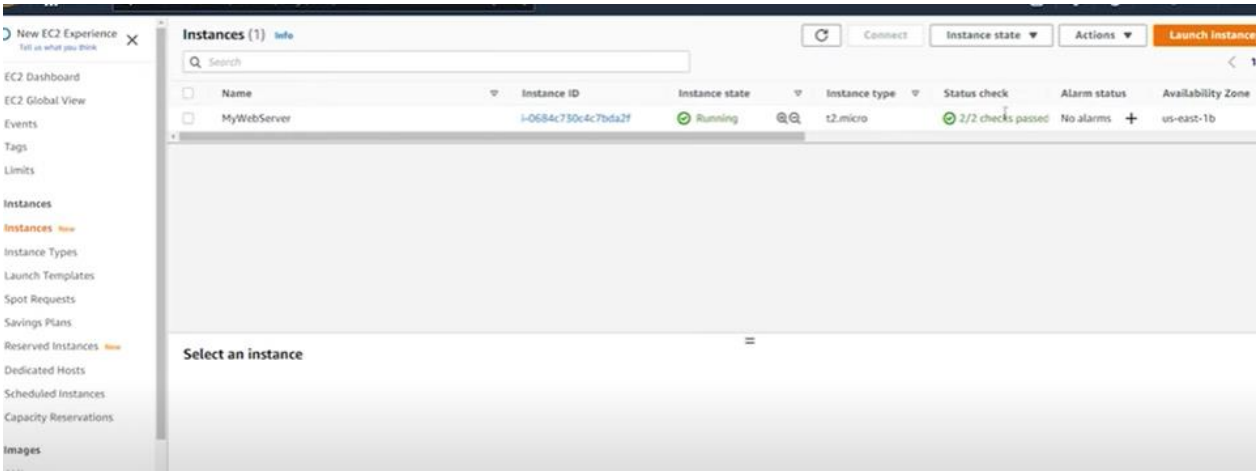
1. Create a VPC
2. Create public and private subnets
3. Create Route Table
4. Add Internet Gateway
5. Create EC2 instances
6. Create Target Groups
7. Create Elastic Load Balancer
8. Create Auto Scaling Groups

7. Flow Chart

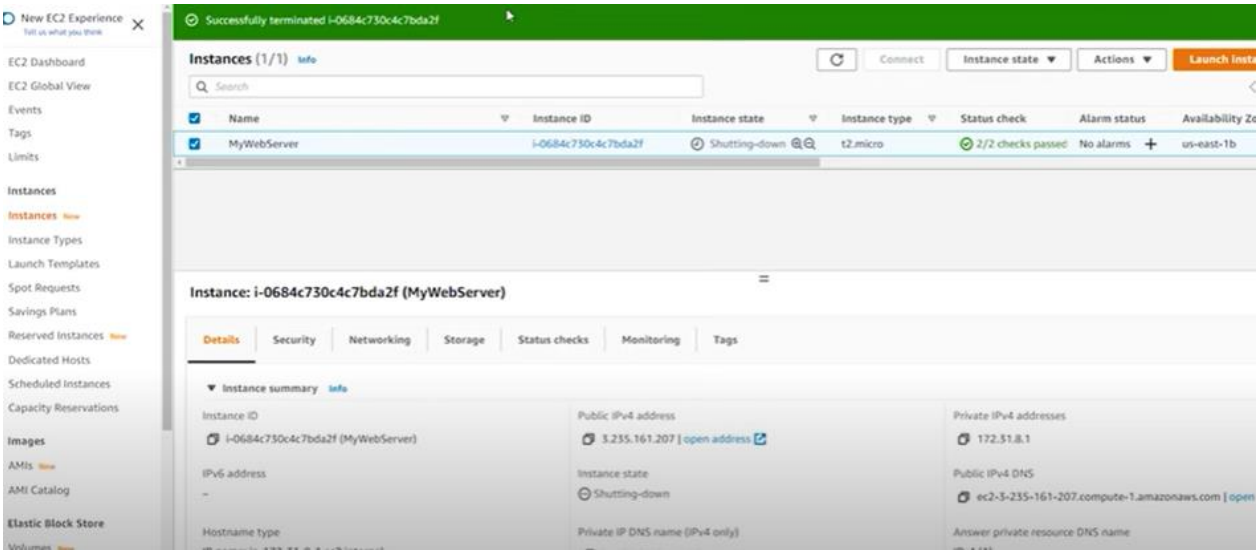


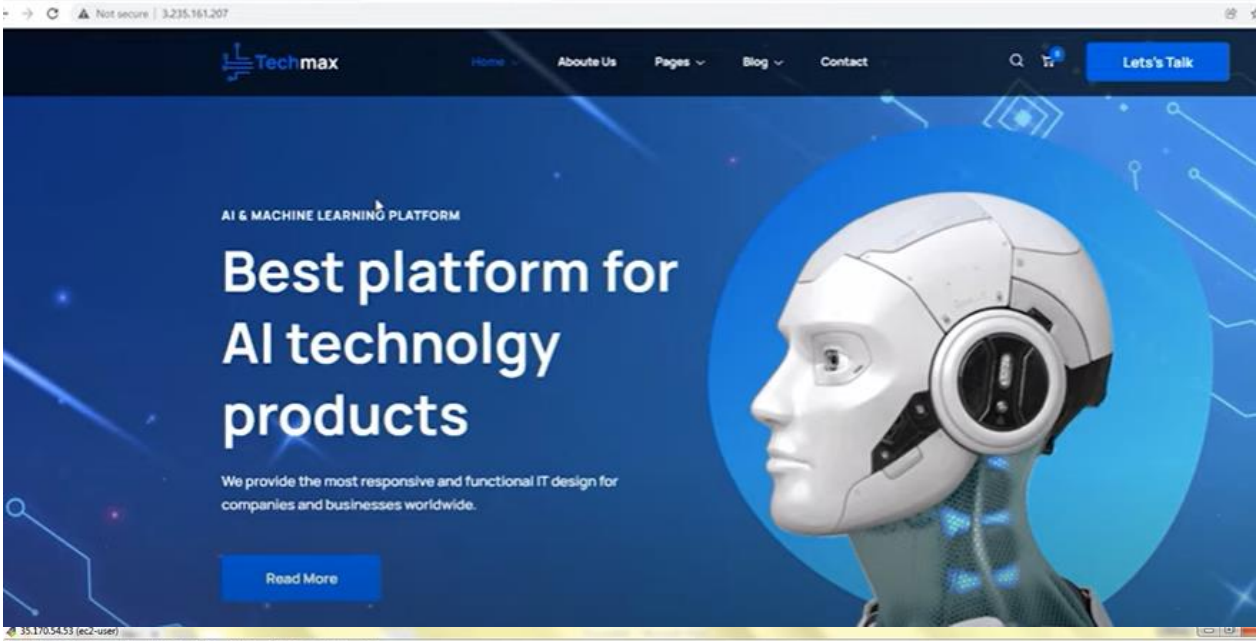
8. RESULT:

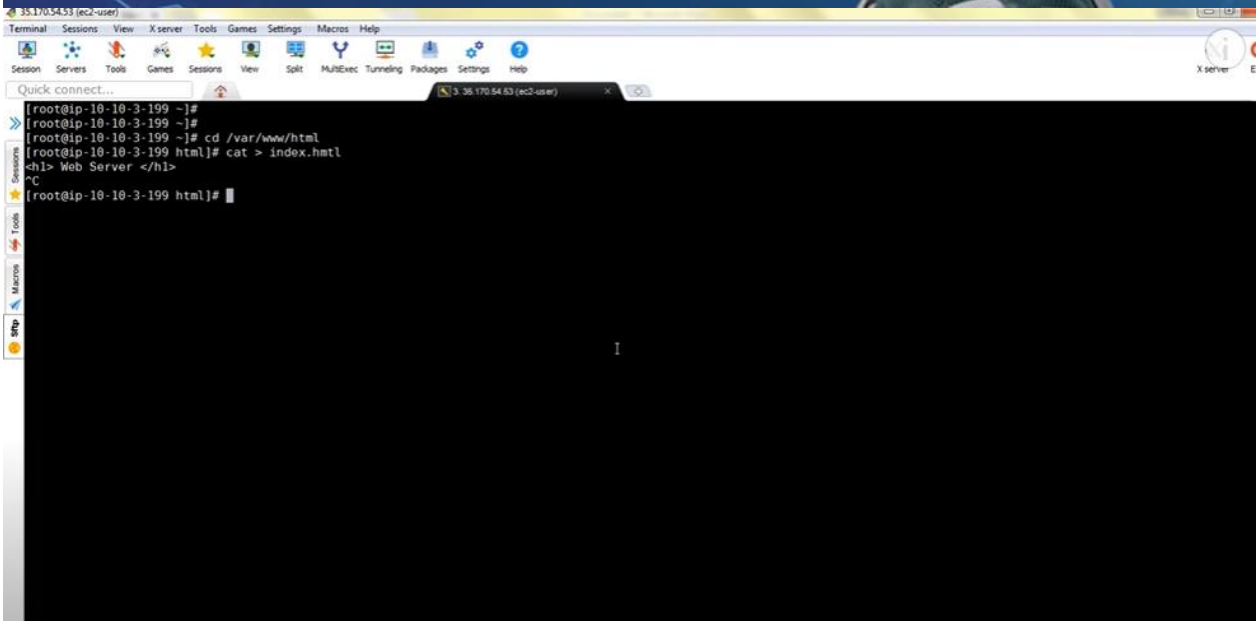

Security group name Info
WebServer SG
Name cannot be edited after creation.
Description Info
Allow access on ports 80 and 22
VPC Info
vpc-be482cc3
Inbound rules Info
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
HTTP
TCP
80
Anywher...
0.0.0.0/0
Delete
SSH
TCP
22
My IP
108.48.102.126/32
Delete
Add rule

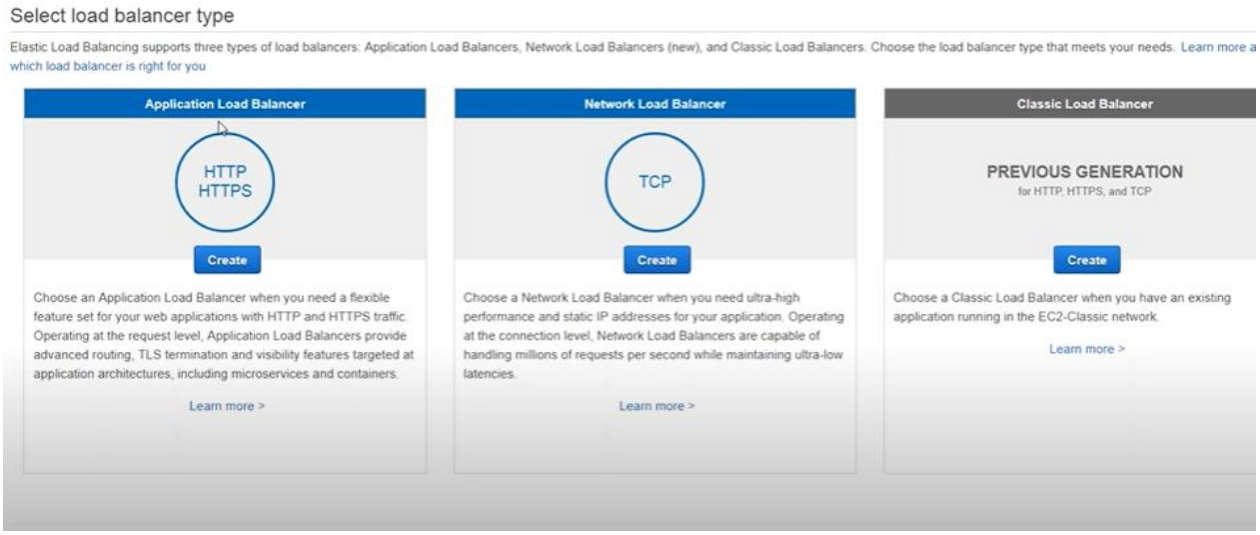

New EC2 Experience
EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
Instances (1) Info
Connect
Instance state
Actions
Launch instance
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
MyWebServer
i-0684c730c4c7bda2f
Running
t2.micro
2/2 checks passed
No alarms
us-east-1b
Select an instance

```
Using username "ec2-user".
Authenticating with public key "myec2key"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
11 package(s) needed for security, out of 15 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-8-1 ~]$ sudo su
[root@ip-172-31-8-1 ec2-user]# yum update -y
```




Not secure | 3.235.161.207
Techmax
Home
About Us
Pages
Blog
Contact
Lets's Talk
AI & MACHINE LEARNING PLATFORM
Best platform for AI technolgy products
We provide the most responsive and functional IT design for companies and businesses worldwide.
Read More


35.170.54.53 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
[root@ip-10-10-3-199 ~]#
[root@ip-10-10-3-199 ~]#
[root@ip-10-10-3-199 ~]# cd /var/www/html
[root@ip-10-10-3-199 html]# cat > index.hmtl
<h1> Web Server </h1>
^C
[root@ip-10-10-3-199 html]#


Select load balancer type
Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers (new), and Classic Load Balancers. Choose the load balancer type that meets your needs. Learn more a
which load balancer is right for you
Application Load Balancer
HTTP
HTTPS
Create
Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing, TLS termination and visibility features targeted at application architectures, including microservices and containers.
Learn more >
Network Load Balancer
TCP
Create
Choose a Network Load Balancer when you need ultra-high performance and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second while maintaining ultra-low latencies.
Learn more >
Classic Load Balancer
PREVIOUS GENERATION
for HTTP, HTTPS, and TCP
Create
Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.
Learn more >

Step 1: Define Load Balancer

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Add

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Availabil Zones to provide higher availability for your load balancer.

VPC vpc-7af8ef02 (10.10.0.0/16) | DemoVPC

Available subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
	us-east-1d	subnet-c76b02e8	10.10.4.0/24	Subnet-1D

Selected subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
	us-east-1a	subnet-ba5e68f1	10.10.1.0/24	Subnet-1A
	us-east-1b	subnet-78274a25	10.10.2.0/24	Subnet-1B
	us-east-1c	subnet-03d8fc67	10.10.3.0/24	Subnet-1C

Step 2: Assign Security Groups

'ou have selected the option of having your Elastic Load Balancer inside of a VPC, which allows you to assign security groups to your load balancer. Please select the security groups to assign to this load balancer. This can be hanged at any time.

Assign a security group: Create a **new** security group
Select an **existing** security group

Security group name: ELB-SG

Description: ELB-SG

Type	Protocol	Port Range	Source
Custom TCP F	TCP	80	Custom 0.0.0.0/0

Add Rule

Step 4: Configure Health Check

Your load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer. Customize the health check to meet your specific needs.

Ping Protocol HTTP
Ping Port 80
Ping Path /index.html

Advanced Details

Response Timeout 5 seconds
Interval 30 seconds
Unhealthy threshold 2
Healthy threshold 10

1. Define Load Balancer | 2. Assign Security Groups | 3. Configure Security Settings | 4. Configure Health Check | 5. Add EC2 Instances | 6. Add Tags | 7. Review

Step 5: Add EC2 Instances

'he table below lists all your running EC2 Instances. Check the boxes in the Select column to add those instances to this load balancer.

/PC vpc-7af8ef02 (10.10.0.0/16) | DemoVPC

Instance	Name	State	Security groups	Zone	Subnet ID	Subne
i-0fa2b0503b62e97ba	WebServer	running	WebSG	us-east-1c	subnet-03d8fc67	10.10.3

Availability Zone Distribution
instance in us-east-1c

Enable Cross-Zone Load Balancing
Enable Connection Draining 300 seconds

Load Balancer Creation Status

Successfully created load balancer

Load balancer Demo-ELB was successfully created.

Note: It may take a few minutes for your instances to become active in the new load balancer.

Close

olumes
napshots
ETWORK &
ECURITY
ecurity Groups
lastic IPs
lacement Groups
ey Pairs
etwork Interfaces
OAD BALANCING
oad Balancers
arget Groups
UTO SCALING
aunch Configurations
uto Scaling Groups

Owned by me | Filter by tags and attributes or search by keyword | 1 to 1 of 1

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date	Platfor
	Demo-Image	ami-0743717d	283398654523/...	283398654523	Private	available	January 25, 2018 at 10:52:5...	Other L

1. Choose AMI | 2. Choose Instance Type | 3. Configure Details | 4. Add Storage | 5. Configure Security Group | 6. Review

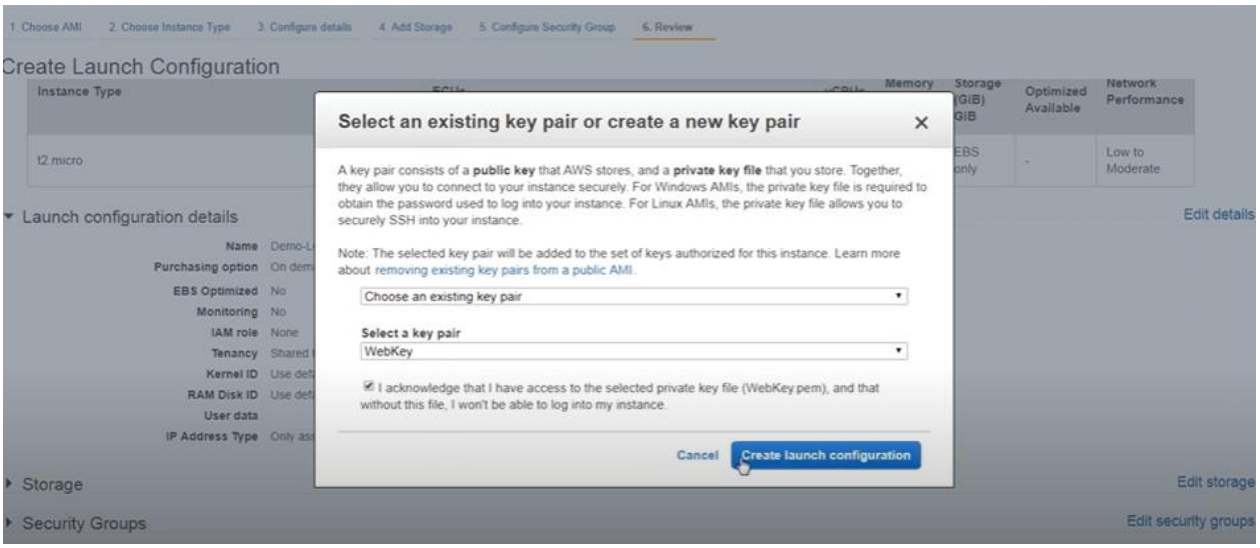
Create Launch Configuration

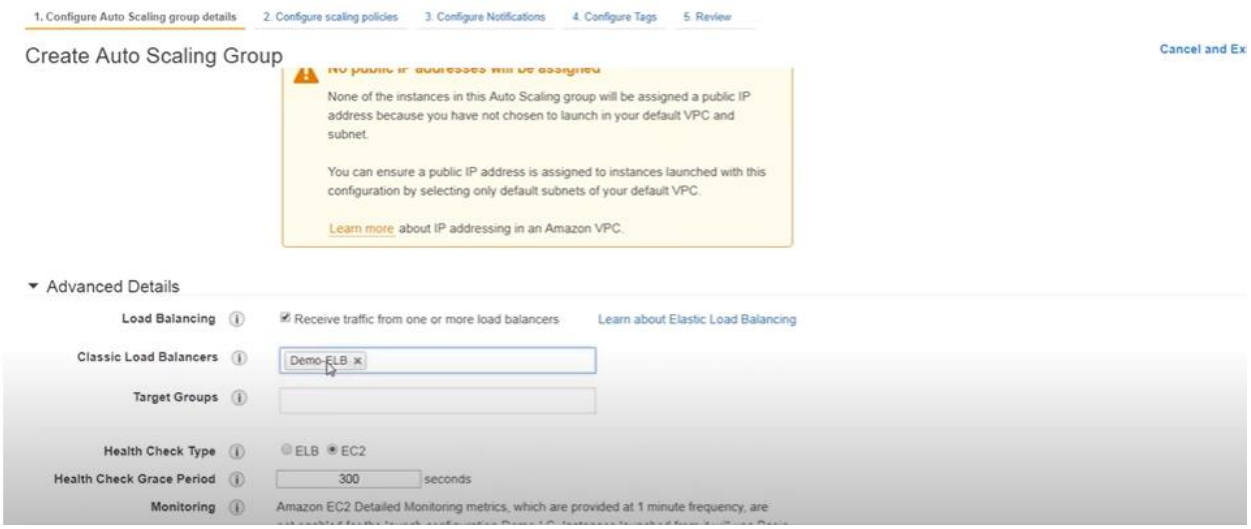
mazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and ne apacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

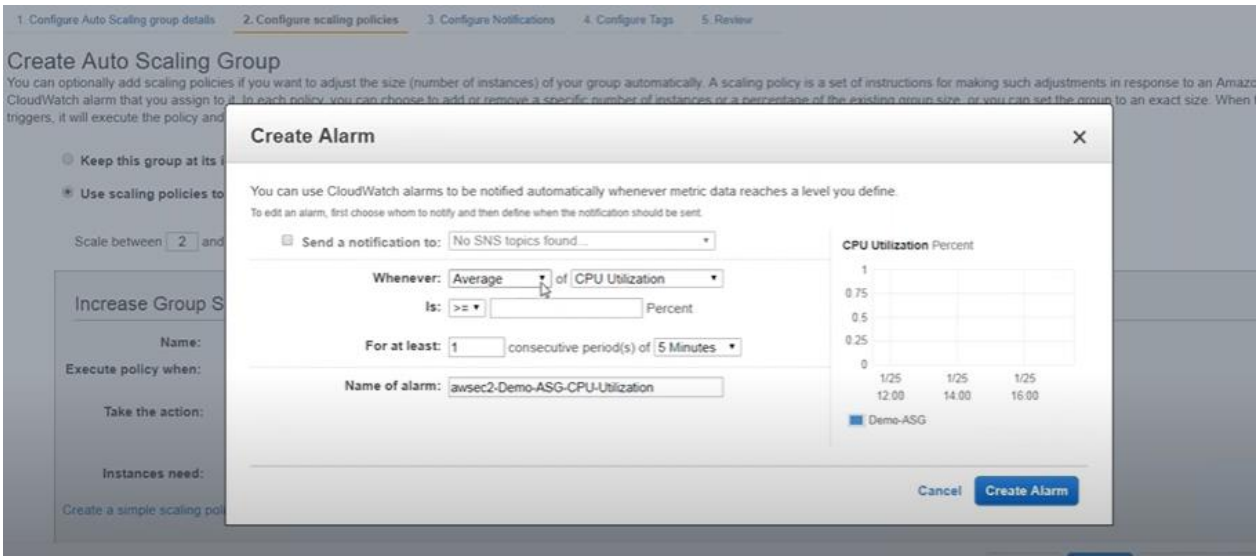
Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

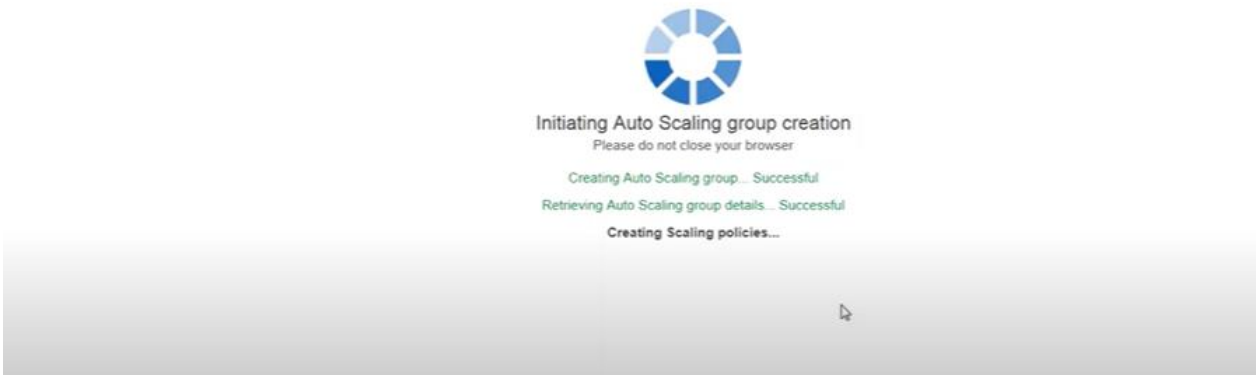
Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performan
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moder
General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moder
General purpose	t2.small	1	2	EBS only	-	Low to Moder
General purpose	t2.medium	2	4	EBS only	-	Low to Moder
General purpose	t2.large	2	8	EBS only	-	Low to Moder
General purpose	t2.xlarge	4	16	EBS only	-	Moderate
General purpose	t2.2xlarge	8	32	EBS only	-	Moderate

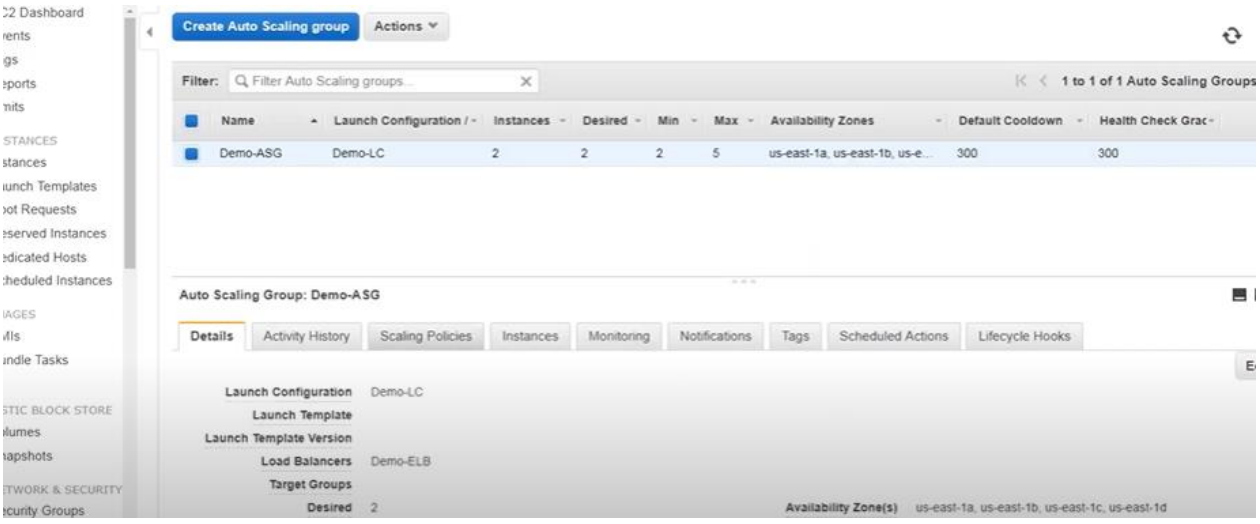

1. Choose AMI
2. Choose Instance Type
3. Configure details
4. Add Storage
5. Configure Security Group
6. Review
Create Launch Configuration
Instance Type
Memory
Storage (GiB)
Optimized Available
Network Performance
t2.micro
Low to Moderate
Launch configuration details
Edit details
Name
Purchasing option
EBS Optimized
Monitoring
IAM role
Tenancy
Kernel ID
RAM Disk ID
User data
IP Address Type
Storage
Edit storage
Security Groups
Edit security groups
Select an existing key pair or create a new key pair
A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.
Choose an existing key pair
Select a key pair
WebKey
I acknowledge that I have access to the selected private key file (WebKey.pem), and that without this file, I won't be able to log into my instance.
Cancel
Create launch configuration

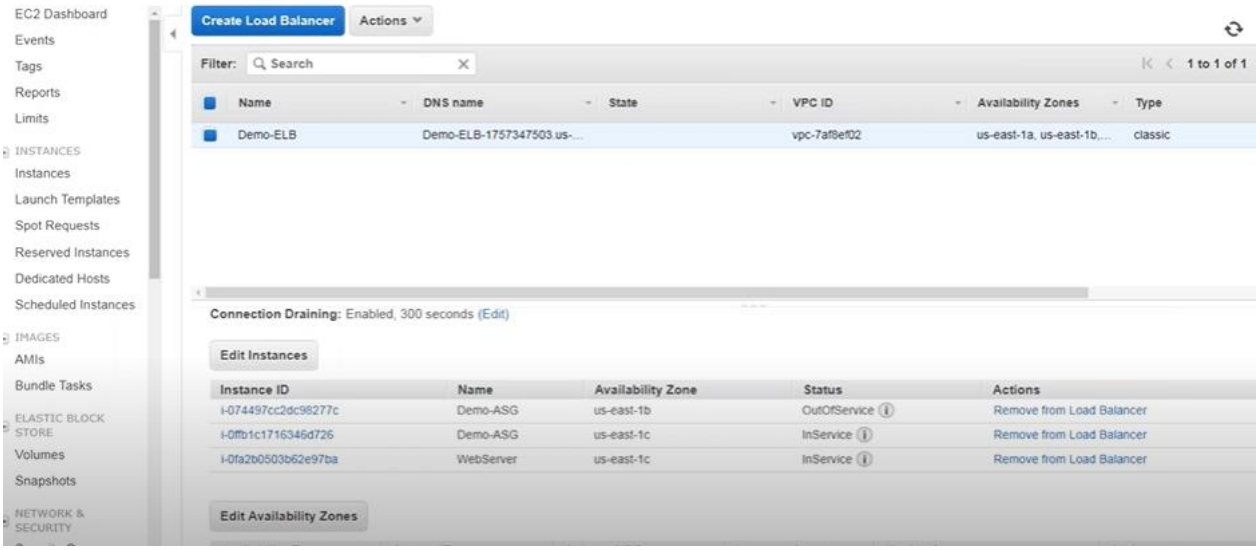

1. Configure Auto Scaling group details
2. Configure scaling policies
3. Configure Notifications
4. Configure Tags
5. Review
Create Auto Scaling Group
Cancel and Ex
None of the instances in this Auto Scaling group will be assigned a public IP address because you have not chosen to launch in your default VPC and subnet.
You can ensure a public IP address is assigned to instances launched with this configuration by selecting only default subnets of your default VPC.
Learn more about IP addressing in an Amazon VPC.
Advanced Details
Load Balancing
Receive traffic from one or more load balancers
Learn about Elastic Load Balancing
Classic Load Balancers
Demo-ELB
Target Groups
Health Check Type
ELB
EC2
Health Check Grace Period
300
seconds
Monitoring
Amazon EC2 Detailed Monitoring metrics, which are provided at 1 minute frequency, are


1. Configure Auto Scaling group details
2. Configure scaling policies
3. Configure Notifications
4. Configure Tags
5. Review
Create Auto Scaling Group
Keep this group at its i
Use scaling policies to
Scale between 2 and
Increase Group S
Name:
Execute policy when:
Take the action:
Instances need:
Create a simple scaling poli
Create Alarm
You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.
To edit an alarm, first choose whom to notify and then define when the notification should be sent.
Send a notification to:
No SNS topics found...
Whenever:
Average
of
CPU Utilization
Is:
>=
Percent
For at least:
1
consecutive period(s) of
5 Minutes
Name of alarm:
awsec2-Demo-ASG-CPU-Utilization
CPU Utilization Percent
1
0.75
0.5
0.25
0
1/25 12:00
1/25 14:00
1/25 16:00
Demo-ASG
Cancel
Create Alarm

Auto Scaling group creation status







9. References

- https://cloudknight.medium.com/building-a-fault-tolerant-web-application-on-aws-a-step-by-step-guide-to-using-cloudformation-590988a99eb8
- https://aws.plainenglish.io/create-a-scalable-resilient-fault-tolerant-3-tier-architecture-9cdb3064b2de
- https://awstip.com/designing-a-fault-tolerant-web-application-building-a-highly-available-3-tier-architecture-e740765dd069