

Day-13

.....

Today I learned about two very important services that work together to keep performance balanced and make AWS cost-effective. Autoscaling automatically adjusts the number of resources (like servers) based on demand. This prevents overload during high usage and saves costs by scaling down during low demand. Load balancing distributes traffic to multiple servers. This prevents any single server from being overload and keeps performance balanced and highly available. The process for integrating Autoscaling and Load balancing in an AWS are as follows:

Creating a Target group:

The screenshot shows the 'Specify group details' page in the AWS Management Console. On the left, a navigation pane shows 'Step 1: Specify group details' as the active step, with 'Step 2: Register targets' below it. The main content area is titled 'Specify group details' and includes a sub-header 'Basic configuration'. Below this, it says 'Settings in this section can't be changed after the target group is created.' The 'Choose a target type' section has three radio buttons: 'Instances' (selected), 'IP addresses', and 'Lambda function'. Each option has a list of bullet points describing its capabilities. For 'Instances', it mentions supporting load balancing to instances within a specific VPC and facilitating the use of Amazon EC2 Auto Scaling. For 'IP addresses', it mentions supporting load balancing to VPC and on-premises resources, facilitating routing to multiple IP addresses, and offering flexibility with microservice-based architectures. For 'Lambda function', it mentions facilitating routing to a single Lambda function and being accessible to Application Load Balancers only.

Eg: (Creating a target group here for instances)

Creating a Load Balancer:

Create Application Load Balancer [info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

► How Application Load Balancers work

Basic configuration

Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

LabELB

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)

Scheme can't be changed after the load balancer is created.

Internet-facing

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name is publicly resolvable.
- Requires a public subnet.

Internal

- Serves internal traffic.
- Has private IP addresses.
- DNS name is publicly resolvable.
- Compatible with the IPv4 and Dualstack IP address types.

Load balancer IP address type [info](#)

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

IPv4

Includes only IPv4 addresses.

Mapping a network and Selecting Target group for load balancer:

Network mapping [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC [Info](#)

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm VPC for your targets, view [target groups](#). For a new VPC, [create a VPC](#).

Lab VPC

vpc-0ed1efc6f6257e4c5

IPv4 VPC CIDR: 10.0.0.0/16

Mappings [Info](#)

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones

☒ us-east-1a (use1-az2)

Subnet

subnet-0590bd8ab352d7bce

Public Subnet 1

IPv4 subnet CIDR: 10.0.0.0/24

IPv4 address

Assigned by AWS

☒ us-east-1b (use1-az4)

Subnet

subnet-06304fddf57096ef2

Private Subnet 2

IPv4 subnet CIDR: 10.0.3.0/24

Listeners and routing [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Remove

Protocol

Port

Default action [Info](#)

HTTP

:

80

Forward to

LabGroup

HTTP

1-65535

Target type: instance, IPv4

[Create target group](#)

Listener tags - optional

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

Creating Autoscaling group in template:

LabConfig (lt-061c3c8f174048169)

Launch template details

Launch template ID

lt-061c3c8f174048169

Launch template name

LabConfig

Default version

1

assumed-2031030051@silvero

Details

Versions

Template tags

Actions

Delete template

Launch instance from template

Modify template (Create new version)

Delete template version

Set default version

Manage tags

Create Spot Fleet

Create Auto Scaling group

Note: Here a template was pre-created by me to attach it with AutoScaling.

Integrating Load Balancer to AutoScaling:

- Step 1
Choose launch template or configuration
- Step 2
Choose instance launch options
- Step 3 - optional
Integrate with other services
- Step 4 - optional
Configure group size and scaling
- Step 5 - optional
Add notifications
- Step 6 - optional
Add tags
- Step 7
Review

Integrate with other services - optional [Info](#)

Use a load balancer to distribute network traffic across multiple servers. Enable service-to-service communications with VPC Lattice. Shift resources away from impaired Availability Zones with zonal shift. You can also customize health check replacements and monitoring.

Load balancing [Info](#)

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 an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

☒ Choose from your load balancer target groups

This option allows you to attach Application, Network, or Gateway Load Balancers.

☐ Choose from Classic Load Balancers

Existing load balancer target groups

Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

LabGroup | HTTP
Application Load Balancer: LabELB

Configuring AutoScaling:

Configure group size and scaling - optional [Info](#)

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size [Info](#)

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling [Info](#)

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity

2

Equal or less than desired capacity

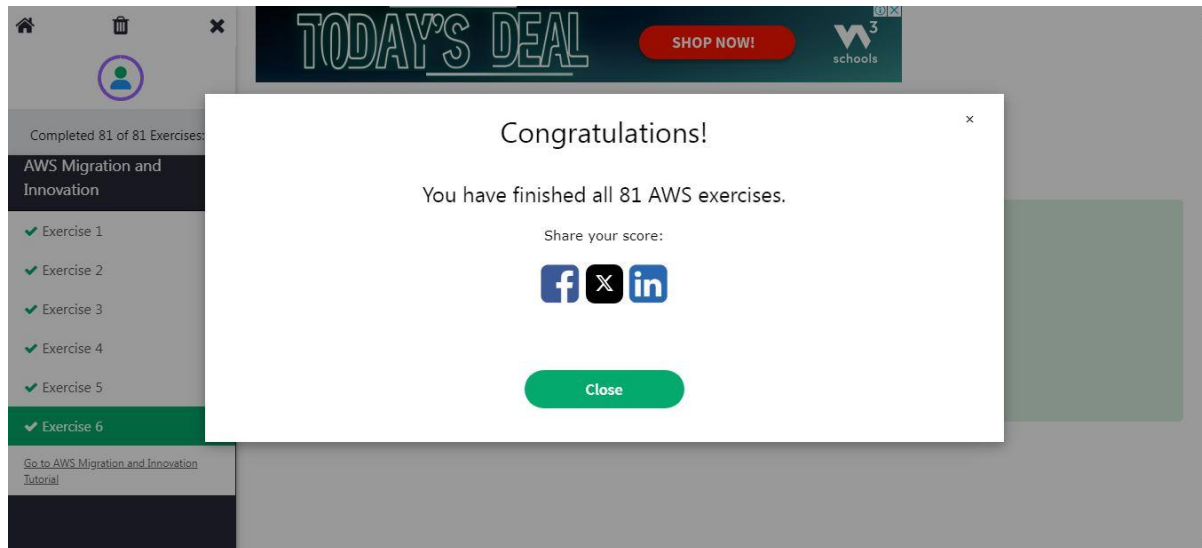
Max desired capacity

6

Equal or greater than desired capacity

And this is how AutoScaling is created with the integration of Load Balancing.

In addition to that, I also completed series of exercises form the “W3Schools” platform on AWS cloud, its services and its properties. It helped me a lot to enhance my understanding of cloud and its services. I have attached the image of completion below:



I was also given a set of question to solve by myself. I got a score of 15 out of 20 and after a detailed evaluation by Sir, I was told to focus more on the S3 service. Also Read and understand the question carefully. I have attached the set of questions below:

Sr. No.	Subject	Question	Option A	Option B	Option C	Option D	Correct Answer	My Answer
1	AWS EC2	What are the key features of Amazon EC2?	Scalable compute capacity	Managed database services	Data analytics services	Content delivery network	A	Option A
2	AWS EC2	Which statement is true about security groups in EC2?	Security groups allow all traffic by default	Security groups act as virtual firewalls	Security groups block all traffic by default	Security groups allow only inbound traffic	B	Option B
3	AWS S3	What are the storage classes available in S3?	Standard, Intelligent-Tiering	Infrequent Access, Glacier	One Zone-IA, Cold Archive	Temporary Storage, Long-Term Storage	A	Option D
4	AWS S3	How does S3 ensure data durability?	Through encryption	Replication across regions	Automatic backups	Through user-defined configurations	B	Option C
5	AWS VPC	What is the primary purpose of a VPC?	Connect isolated cloud resources	Store backup data	Improve app performance	Automate application deployment	A	Option A
6	AWS VPC	What is a characteristic of subnets in a VPC?	Subnets can span multiple Availability Zones	Subnets are restricted to a single AZ	Subnets are shared across accounts	Subnets require manual configuration	B	Option A
7	AWS EBS	What is the key difference between SSD and HDD EBS volumes?	SSD offers higher throughput	HDD is more expensive	HDD offers lower latency	SSD is less durable	A	Option A
8	AWS EBS	How can you back up data from an EBS volume?	Use AWS Backup	Create a snapshot	Clone the volume	Use Elastic Block Store Vault	B	Option B
9	AWS Instance Type	What are the main categories of EC2 instance types?	General Purpose, Compute Optimized	Memory Optimized, Storage Optimized	Burstable Performance	GPU Instances	B	Option B
10	AWS Instance Type	Which factor is important when choosing an EC2 instance type?	Number of vCPUs only	Network bandwidth	Instance pricing model	Region location	B	Option A
11	AWS EC2	What is an AMI in EC2?	A virtual server	A pre-configured virtual machine image	A storage service	An EC2 configuration file	B	Option B
12	AWS EC2	How does EC2 Auto Scaling improve performance?	By reducing latency	By dynamically adjusting instances	By providing backups	By managing security groups	B	Option B
13	AWS S3	What is S3 bucket versioning?	Allows bucket replication	Tracks changes to objects	Enables faster data transfer	Deletes old versions automatically	B	Option D
14	AWS S3	How can you restrict access to an S3 bucket?	Using IAM policies	Through bucket policies	Using S3 Access Points	Through CloudFront policies	B	Option B
15	AWS VPC	What is an Internet Gateway in a VPC?	Routes private IPs to the internet	Connects a VPC to the internet	Allows external apps to connect to VPCs	Facilitates internal VPC communication	B	Option B
16	AWS VPC	How is a NAT Gateway used in a VPC?	Allows instances in private subnets to access the internet	For routing traffic between VPCs	To enforce access restrictions	To connect multiple subnets	A	Option A
17	AWS EBS	What is the maximum size of an EBS volume?	1 TB	16 TB	64 TB	32 TB	C	Option D
18	AWS EBS	What happens when you delete an EBS volume?	It is archived	It is permanently deleted	It is backed up automatically	It is detached	B	Option B
19	AWS Instance Type	What is the purpose of burstable performance instances?	For consistent workloads	For spiky workloads	For web servers	For GPU processing	B	Option B
20	AWS Instance Type	Which workload is best suited for Compute Optimized instances?	Batch processing	High-performance computing	Data analytics	Streaming applications	B	Option B
							15/20	