

Module 10: Auto-Scaling and Monitoring

Thursday, September 18, 2025 4:14 PM

A. Topics and Learning Objectives (LOs):

- Topic:

- + Elastic Load Balancing
- + Amazon Cloud Watch
- + Amazon EC2 Auto Scaling

B. Elastic Load Balancing

- Distributes incoming application or network traffic across multiple targets in a single AZ or across multiple AZ
- Scales user's load balancer as traffic to user's application changes over time

| Types of load balancers | | |
|---|--|---|
| Application Load Balancer | Network Load Balancer | Classic Load Balancer (Previous Generation) |
| <ul style="list-style-type: none">• Load balancing of HTTP and HTTPS traffic• Routes traffic to targets based on content of request• Provides advanced request routing targeted at the delivery of modern application architectures, including microservices and containers• Operates at the application layer (OSI model layer 7) | <ul style="list-style-type: none">• Load balancing of TCP, UDP, and TLS traffic where extreme performance is required• Routes traffic to targets based on IP protocol data• Can handle millions of requests per second while maintaining ultra-low latencies• Is optimized to handle sudden and volatile traffic patterns• Operates at the transport layer (OSI model layer 4) | <ul style="list-style-type: none">• Load balancing of HTTP, HTTPS, TCP, and SSL traffic• Load balancing across multiple EC2 instances• Operates at both the application and transport layers. |

- Elastic Load Balancing use cases:

- + Highly available and fault-tolerance applications
- + Containerized applications
- + Elasticity and scalability
- + VPC
- + Hybrid environments
- + Invoke Lambda functions over HTTP(S)

- Load Balancing Monitoring:

- + Amazon CloudWatch metrics: Used to verify that the system is performing as expected and creates an alarm to initiate an action if a metric goes outside an acceptable range.
- + Access logs: Capture detailed information about requests sent to user's load balancer
- + AWS CloudTrail logs: Capture the who, what, when, and where of API interactions in AWS services

C. Amazon CloudWatch

- Monitoring AWS resources: To use AWS efficiently, user needs insight into AWS resources:

- + How do you know when users should launch more Amazon EC2 instances?
- + Is your application's performance or availability being affected by a lack of sufficient capacity?
- + How much of your infrastructure is actually being used?

- Amazon CloudWatch:

- + Monitors: AWS resources and Applications that run on AWS
- + Collects and tracks: Standard metrics and Custom metrics
- + Alarms: Send notifications to an Amazon SNS topic and Perform Amazon EC2 Auto Scaling or Amazon EC2 actions.
- + Events: Define rules to match changes in AWS environment and route these events to one or more target functions for processing.

- CloudWatch alarms:
 - + Create alarms based on:
 - * Statuc threshold
 - * Anomaly detection
 - * Metric math expression
 - + Specify
 - * Namespace
 - * Metric
 - * Statistic
 - * Period
 - * Conditions
 - * Additional configurations
 - * Actions

D. Amazon EC2 Auto Scaling

- Helps user to maintain application availability
- Enables user to automatically add or remove EC2 instances according to conditions that user define
- Detects impaired EC2 instances and unhealthy applications and replaces the instances without user intervention
- Provides several scaling options - Manual, sheduled, dynamic, or on-demand, and predictive
- Auto Scaling Groups: An Auto Scaling group is a collection of EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management
- Monitors user's applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost
- Provides a simple, powerful user interface that enables user to build scaling plans for resoirces, including:
 - + Amazon EC2 instances and Spot Fleets
 - + Amazon Elastic Container Service (Amazon ECS) Tasks
 - + Amazon DynamoDB tables and indexes
 - + Amazon Aurora Replicas