TNGS Learning Solutions
AWS Solutions Architect
Online Course

AWS Auto Scaling

AWS Auto Scaling

- AWS Auto Scaling is a service provided by Amazon Web Services (AWS) that automatically adjusts the number of compute resources in your applications to maintain performance and availability while optimizing costs.
- It allows you to ensure that your application can automatically scale up or down based on demand, traffic patterns, and other metrics.

AWS Auto Scaling

- **Scaling Groups**: The primary resource in AWS Auto Scaling is an Auto Scaling group. This group represents a collection of Amazon Elastic Compute Cloud (EC2) instances that share similar characteristics and can be scaled together. You define the group's launch configuration, which includes the Amazon Machine Image (AMI), instance type, and other settings.

AWS Auto Scaling

- **Scaling Policies**: Auto Scaling groups use scaling policies to determine when and how to add or remove instances. There are two types of scaling policies:
  - **Dynamic Scaling**: This type of scaling policy responds to changes in demand by automatically adjusting the number of instances in the group. You can configure scaling policies based on metrics like CPU utilization, network traffic, or custom CloudWatch metrics.
  - **Scheduled Scaling**: Scheduled scaling policies allow you to plan for predictable changes in capacity. You can set specific dates and times for adding or removing instances.

AWS Auto Scaling

- **Launch Configuration**: A launch configuration defines the specifications of the instances that Auto Scaling launches. You can specify the AMI, instance type, security groups, and user data. When scaling up, Auto Scaling uses this configuration to create new instances.
- **Instance Termination Policies**: You can configure termination policies to determine which instances to terminate when scaling down. This helps ensure efficient resource usage and can be based on various criteria, such as oldest launch configuration or instance ID.

AWS Auto Scaling

- **Cooldown Period**: A cooldown period is a configurable time delay between scaling actions. It prevents Auto Scaling from adding or removing instances too rapidly, which can help stabilize your application during fluctuating workloads.
- **Mixed Instances Policy**: AWS Auto Scaling allows you to use a mixed instances policy to launch a combination of On-Demand, Spot, and Reserved Instances within a single Auto Scaling group. This can help reduce costs while maintaining availability.

AWS Auto Scaling

- **Target Tracking Scaling**: You can configure target tracking scaling policies that maintain a specific metric (e.g., CPU utilization or request count per target) at a predefined target value. Auto Scaling automatically adjusts the number of instances to achieve the target.
- **Lifecycle Hooks**: You can set up lifecycle hooks to perform custom actions during instance termination or launch. For example, you can pause the termination process to back up data before an instance is terminated.

AWS Auto Scaling

- **Integration with AWS Services**: Auto Scaling integrates with other AWS services, such as Elastic Load Balancing (ELB), to automatically distribute incoming traffic to newly launched instances. It also works with AWS CloudWatch for monitoring and Amazon CloudWatch Alarms for triggering scaling actions based on metric thresholds.
- **Auto Scaling Warm Pools**: Warm pools allow you to maintain a set of pre-initialized instances in anticipation of incoming traffic spikes. This helps reduce the time it takes to scale out when demand increases.

AWS Auto Scaling

- **Predictive Scaling**: AWS Auto Scaling provides predictive scaling, which uses machine learning algorithms to forecast demand and proactively adjust capacity to meet that demand.
- AWS Auto Scaling is a powerful tool for ensuring the availability, reliability, and cost-effectiveness of your applications in AWS.
- By configuring scaling policies and parameters, you can automate the management of your compute resources and respond to changing workloads effectively.