Auto scaling

in AWS

Creating group of ec2 instance that cann scale up or down depending on conditions you set

Enable elasticity by scaling horizontally through adding or terminating EC2 instance

Autoscaling ensures that you have the right number of AWS ec2 instance when not needed and scaling out to add more instance only when it is required

Auto scaling component

1. **Launch configuration**:- like instance type, AMI, Keypair security group
2. **Autoscaling group** :- Group name, group size, vpc, subnet, health check period
3. **Scaling policy:-** metric type, target value

**PART-02**

* If auto scaling finds that the number of ec2 instances launched by ASG into subject AZs is not balanced (Ec2 instances are not evenly distributed) autoscaling do rebalancing activity by itself
* As always tries to balance the instance distribution across AZs
* While rebalancing ,ASG launches new EC2 instance where there are less EC2 at present , and there terminates the instances form the AZ, that had more instance
* What causes imbalance of Ec2
* If we add or remove some subnet form autoscaling group
* If we manually request for EC2 terminations from our ASG
* An AZ that did not have enough EC2 capacity now has enough capacity and it is one of our autoscaling group AZ
* We can attach a running ec2 instances to an auto scaling group by using aws console or CLI if the below conditions are met
* Instance must be in running state (not terminated or stopped)
* AMI used ot launch the EC2 still exist
* Instances is not part of another autoscaling group
* Instance is in the same az of the some group
* If the existing EC2 instance under the autoscaling group, plus the one to be needed, exceed the max capacity of the autoscaling group, the request will fail, ec2 instance would not be added
* You can manually remove ec2 instance from an autoscaling group using AWS console or CLI
* When you detach on instance, you have the option do document the autoscaling group desired capacity
* If you do not the autoscaling group will launch another instance to replace the one detached
* When you delete an autoscaling group, its parameters like maximum minimum and desired capacity are all set to zero hence it terminates all its ec2 instances
* If you want to keep the ec2 instances and manage there independently you can manually detach them first, then delete the AS
* We can attach one or more elastic load balancer to our autoscaling group
* The elastic load balancer must be in the same region as the autoscaling group
* Once you do this any ec2 instance existing or added by the autoscaling group will be automatically reqistered with the autoscaling group defined ELB
* Instance and the ELB must be in the same VPC
* **Auto scaling health check**
* Auto scaling classifies its EC2 instance health status as either healthy or unhealthy
* By default , as uses ec2 status checks only to determine the health stauts of an instance
* When you have one or more ELB defined with the autoscaling group you can confiture autoscaling to use ‘both’ the EC2 health check and the ELB health check to determine the instances health check
* Health check grace period is 300 sec by default
* If we set zero in grace period , the instance health is checked once it is in service
* Until the grace period timer expires any unhealthy stauts reported by EC2 status check or the ELB attached to the autoscaling group will not be acted upon
* After grace period expires autoscaling group would consider an instance unhealthy in any of the following cases:-
* EC2 status check report to auto-scaling group an instance status other then running
* If ELB health check are configured to be used by the autoscaling then if the ELB report the instance as ‘out of services’
* Unlike AZ rebalancing termination of unhealthy instances happen first, then autoscaling attempt to launch new instance to replace the ones terminated
* Elastic IP and EBS volumes gets detached from the terminated instances you need to manually attach there to the new instances

**In four situation, ASG sends a SNS email notifications:-**

1. **An instance is launched**
2. **An instance is terminated**
3. **An instance fails to launch**
4. **An instance fails to terminate**

**Merging autoscaling group:-**

* Can only be done from the CLI (not AWS console)
* You can merge multiple, single AZ autoscaling group insto a single, one multi-AZ auto scaling group
* Scale-out means launching more ec2 instance
* Scale-in means terminating one or more ec2 instances ty scaling policy
* It is always recommend to create a scale-in event for each scale-out event you create
* Aws EC2 services sends EC2 metrics to cloudwatch about the ASG instances
* Basic monitoring (every 300 secs enabled by default & free of cost)
* You can enabled detailed (every 60 sec – chargeable)
* When the lauch configuration is done by AWS CLI detailed monitoring for EC2 instances in enabled by defaults

Stand by state

* You can manually move an instance form an ASG and put it in standby state
* Instance in standby stable one still managed by autoscaling
* Instance in standby state are changed as normal, in service instances
* They do not count towards available EC2 instance for workload / app use
* Autoscaling does not perform health check an instance in standby state

Auto-Scaling policy

Manual Dynamic step scaling policy

Min max Desired Target Tracking policy Simple scaling policy

Scaling policy :-

* Define how much you want to scale based on defined conditons
* Auto scaling group uses alarms and polices to determine scaling
* For simple or step scaling adjustment can’t change capacity of the group above the max group size or below the min group size

Predictive scaling :- it looks at historic pattern and forecast them into the future to schedule change in the no of ec2 instance it uses machine leraing model to forecast daily and weekly pattern

Target tracking polices:- Increase or decrease the current capcity of the group based on a target value for a specific metric this is similar to the way that your thermostat maintain the temp of your home

Step scaling :- Increase or decrease the current capacity of the group based on a set of scaling adjustment known as step adjustment that vary based on the size of the alarm breach

* Does not support / wait for a cost-down timer
* Support a warm-uptimer time taken by newly launch instance to be ready and contribute to the watched metric

Simple scaling :-

* Single adjustment (up or down) in response to an alarm (cooldown timer – 300 sec defaults)

Schedule scaling:-

Use for predictable load change

* You need to configure a schedule action for a scale out at a specific date/time and to a required capacity
* A schedule action must have a unique date/time you cannot config two schedule activities at the same time/data