EC2 Auto Scaling Groups

Set scaling rules which will automatically launch additional EC2 instance or shutdown instances to meet current demand

Introduction to Auto Scaling Groups

Auto Scaling Groups (**ASG**) contains a collection of EC2 instances that are treated as a group for the purposes of automatic scaling and management.

Automatic scaling can occur via:

1. **Capacity Settings**
2. **Health Check Replacements**
3. **Scaling Policies.**

ASG – Capacity Settings

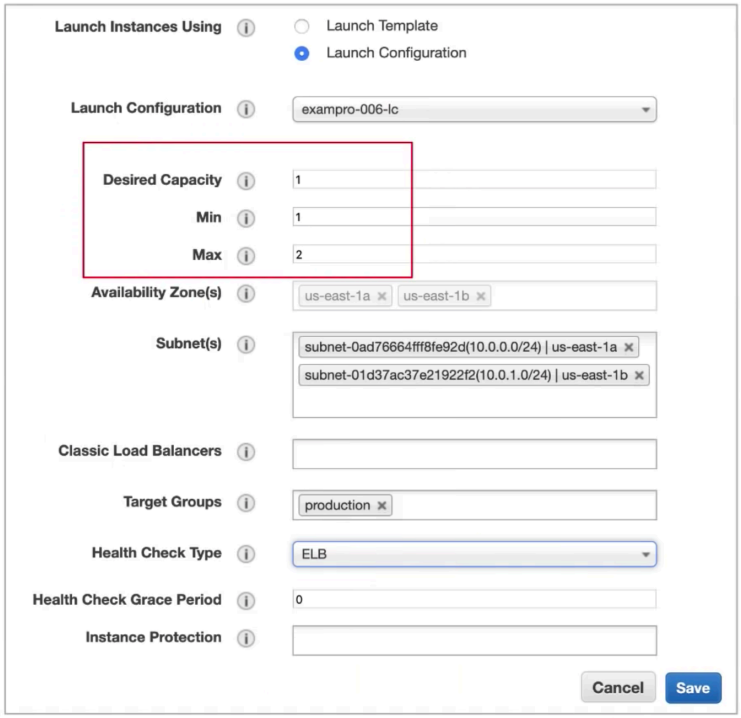
The size of an Auto Scaling Group is based on **Min**, **Max** and **Desired Capacity**.

Min is how many EC2 instances should at least be running.

Max is number EC2 instances allowed to be running.

Desired Capacity is how many EC2 instances you want to ideally run.

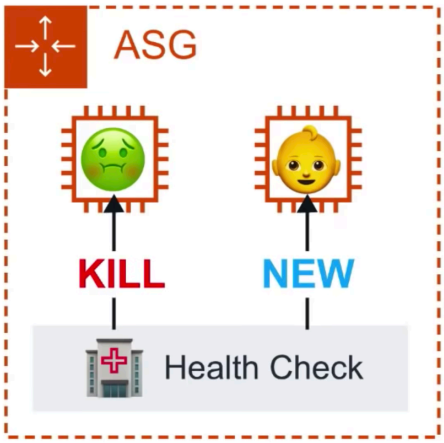
ASG will always launch instances to meet minimum capacity.

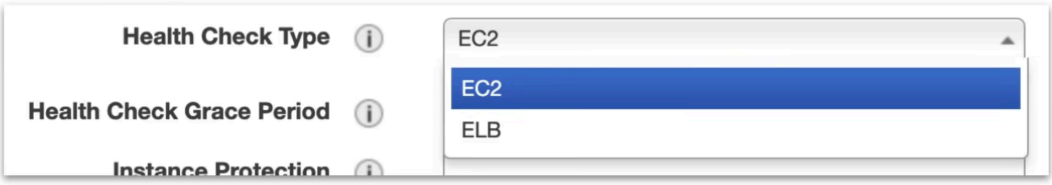


ASG – Health Check Replacements

EC2 Health Check Type

ASG will perform a health check on EC2 instances to determine if there is a software or hardware issue. This is based on the **EC2 Status Checks**. If an instance is considered unhealthy. ASG will terminate and launch a new instance.

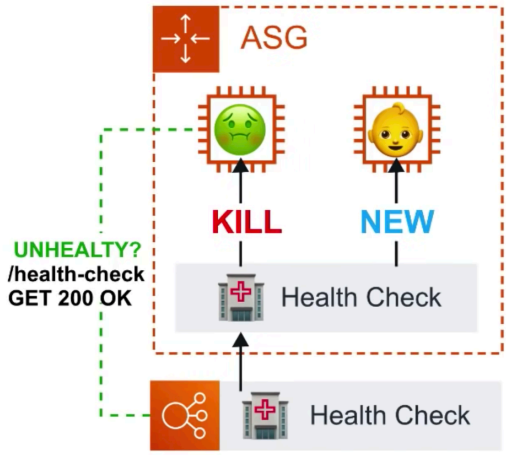


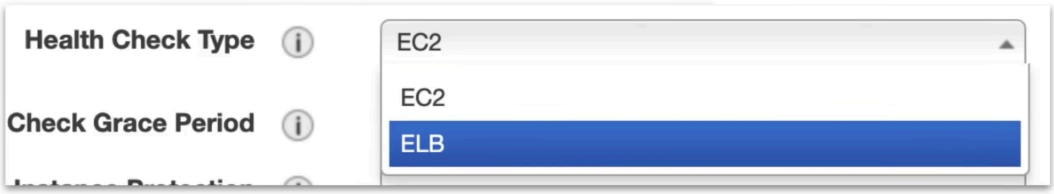


ASG – Health Check Replacements

ELB Health Check Type

ASG will perform a health check based on the ELB health check. ELB can perform health checks by pinging an HTTP(S) endpoint with an expected response. If ELB determines a instance is unhealthy it forwards this information to ASG which will terminate the unhealthy instance.





ASG – Scaling Policies

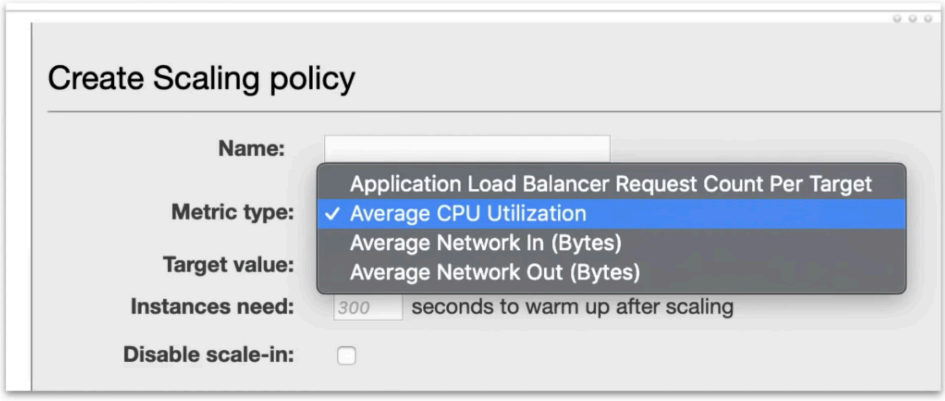
Scaling Out: Adding More Instances

Scaling In: Removing Instances

Target Tracking Scaling Policy

Maintains a specific metric at a target value.

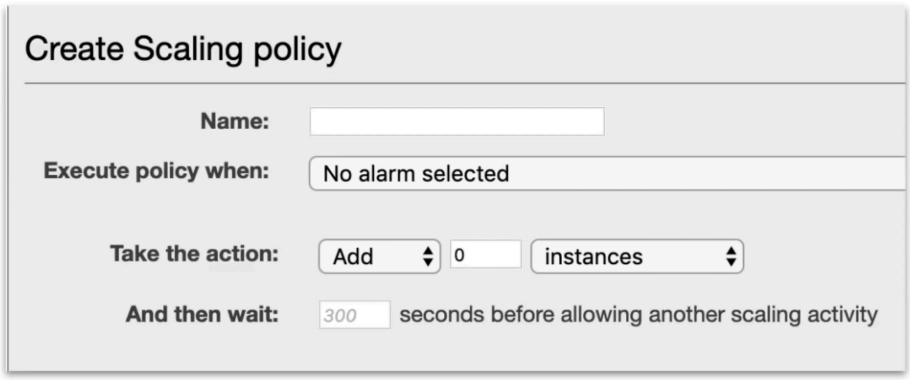
eg. If **Average CPU Utilization** exceeds 75% then add another server.



ASG – Scaling Policies

Simple Scaling Policy

Scales when an **alarm is breached**.

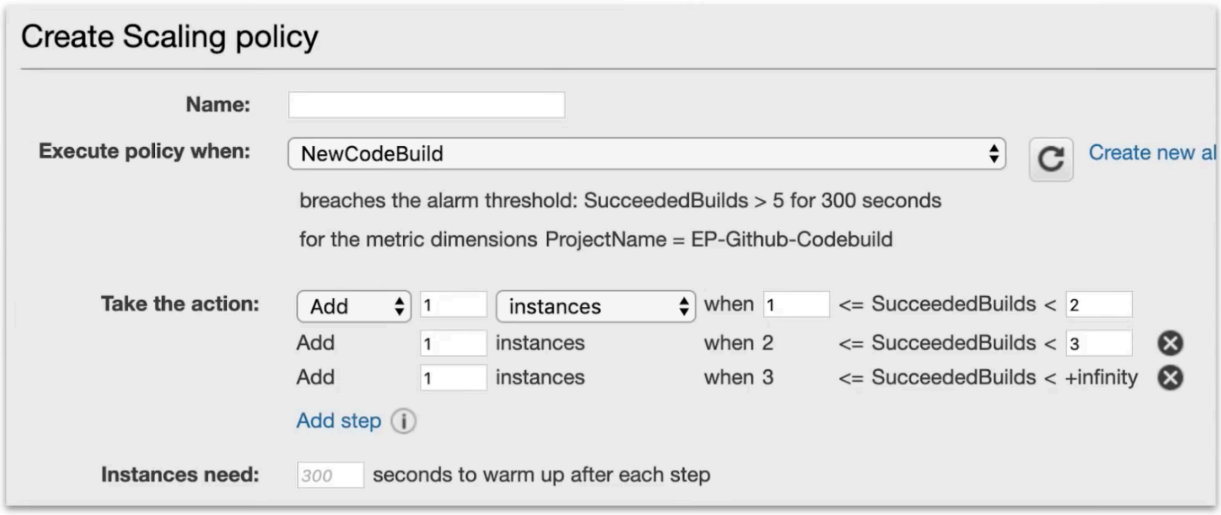


Not recommended, legacy scaling policy. Use scaling policies with steps now.

ASG – Scaling Policies

Scaling policies with steps

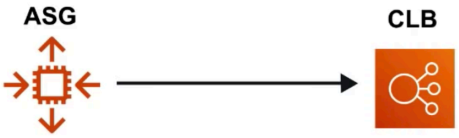
Scales when an **alarm is breached**, can **escalates based on alarm** value changing.

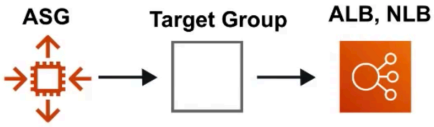


ASG - ELB Integration

ASG can be associated with Elastic Load Balancers (ELB). When ASG is associated with ELB richer health checks can be set.

Classic Load Balancers are associated **directly** to the ASG

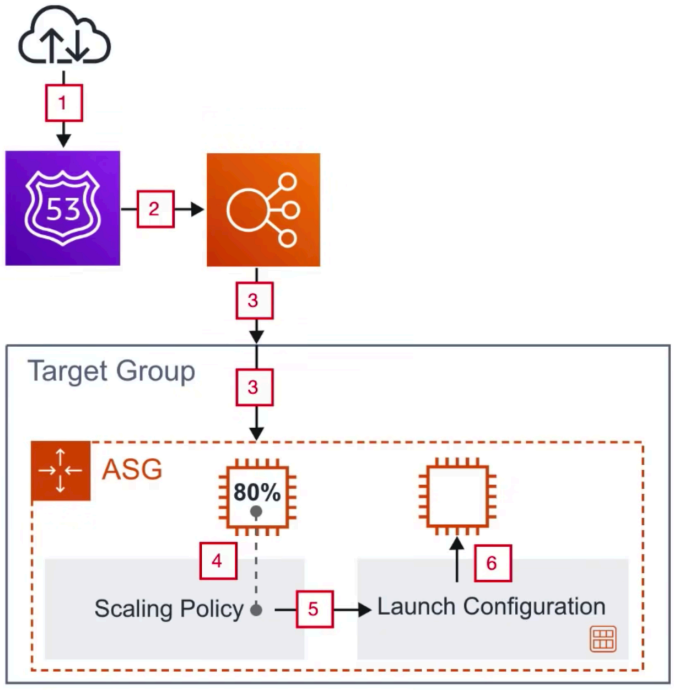




Application and Network Load Balancers are associated **indirectly** via their Target Groups.

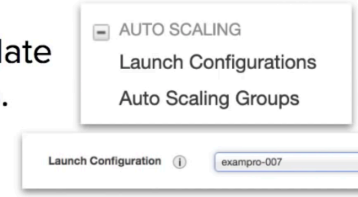
ASG – Use Case

1. Burst of traffic from the internet hits our domain.
2. Route53 points that traffic to our load balancer.
3. Our load balancer passes the traffic to its target group.
4. The target group is associated with our ASG and sends the traffic to instances registered with our ASG
5. The ASG Scaling Policy will check if our instances are near capacity.
6. The Scaling Policy determines we need another instance, and it Launches an new EC2 instance with the associated Launch Configuration to our ASG

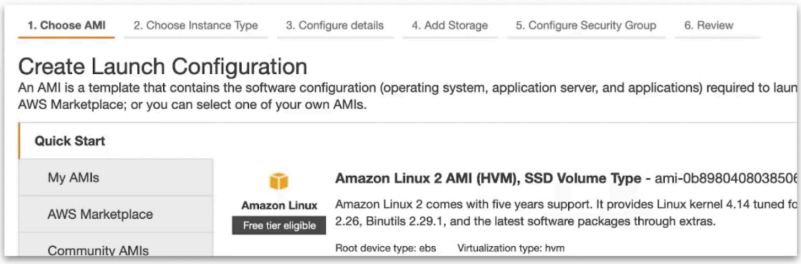


Launch Configuration

A launch configuration is an instance configuration template that an Auto Scaling group uses to launch EC2 instances.



A Launch Configuration is the same process as Launching an EC2 instance except you are saving that configuration to Launch an Instance for later. Hence “Launch Configuration.



Launch Configurations **cannot be edited**, When you need to update your Launch Configuration you create a new one or clone the existing configuration and then manually associate that new Launch Configuration

Launch Templates are Launch Configurations with Versioning, Everyone appears to still use Launch Configurations

EC2 Auto Scaling Groups *CheatSheet*

- An ASG is a collection of EC2 instances grouped for scaling and management
- Scaling Out is when add servers
- Scaling In is when you remove servers
- Scaling Up is when you increase the size of an instance (eg. updating Launch Configuration with larger size)
- Size of an ASG is based on a **Min, Max** and **Desired Capacity**
- **Target Scaling policy** scales based on when a target value for a metric is breached eg. Average CPU Utilization exceed 75%
- **Simple Scaling** policy triggers a scaling when an alarm is breached
- **Scaling Policy with Steps** is the new version of Simple Scaling policy and allows you to create steps based on eculation alarm values.
- Desired Capacity is how many EC2 instances you want to ideally run
- An ASG will always launch instances to meet minimum capacity
- Health checks determine the current state of an instance in the ASG
- Health checks can be run against either an ELB or the EC2 instances
- When an Autoscaling launches a new instance it uses a Launch Configuration which holds the configuration values for that new instance eg. AMI, InstanceType, Role
- Launch Configurations cannot be edited and must be cloned or a new one created
- Launch Configurations must be manually updated in by editing the Auto Scaling settings.