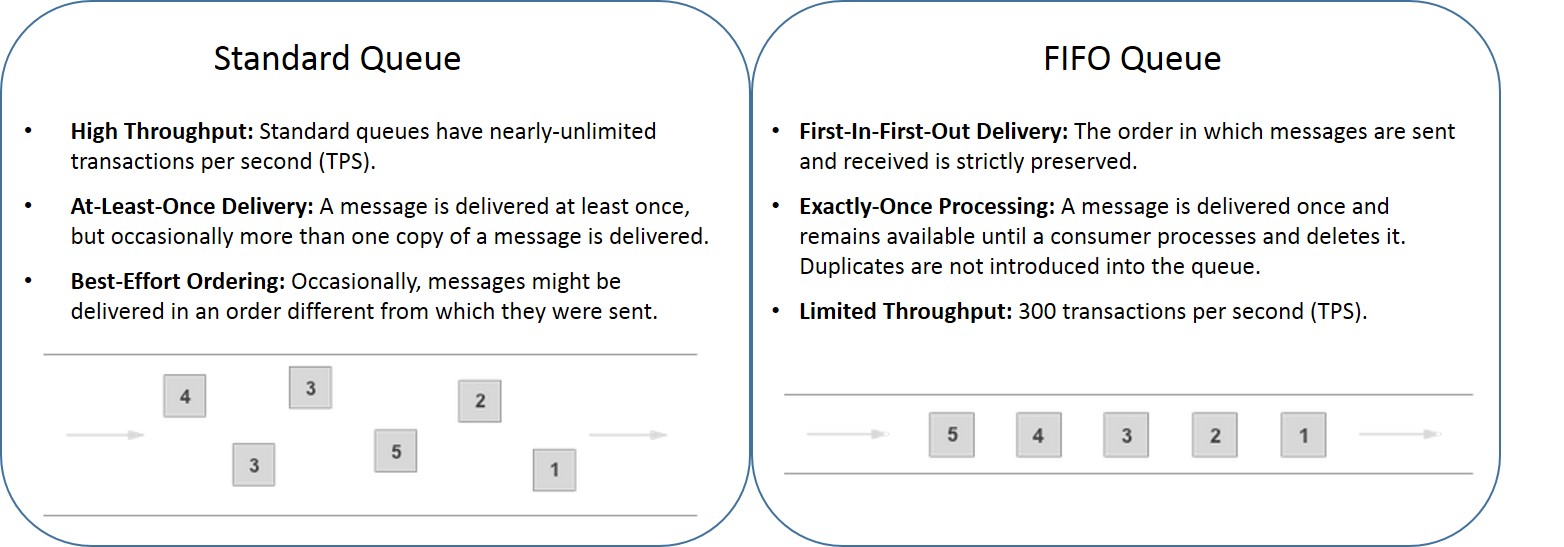
# Difference between Fifo and Standard



# SNS (Simple Notification Service)

Create SNS

\* In AWS console go to SNS

\* Click on Create Topic and enter all the required details

\* Define who can publish messages to topic and who can subscribe to topic in access policy

\* Then create subscription

\* Click on create subscription enter the ARN of topic

\* Select the endpoint like http,email or sms and enter the details under endpoint block like email-id or phone number.

**Create mobile push notification using SNS**

\* Click on mobile push notification

\* Create a application platform

\* Enter name of application

\* Select platform of the application weather its a ios application or google application

\* Then create application endpoint by entering the device token and user id

\* If the notification has to be sent to many subscribers then copy the ARN of the platform application if incase it has to be sent to single subscriber then copy the ARN of application endpoint.

# VPC(Virtual Private Cloud)

**Steps to Create VPC**

\*Click on create VPC

\*Enter cidr block details

\*Click on create vpc

**Steps to create subnets**

\*Click on create subnets in vpc

\*select the created vpc

\*Enter name for the subnets and select the availability zone

\*Enter the CIDR block

\*Click on create subnets

**Steps to create route table**

\*Go to VPC, click on create route table

\*Enter name of the route table

\*Select VPC

\*Create route table

\*Then select route table and click on subnet associations

\*Click on edit subnets associations then select the subnets save associations.

**Steps to create Internet Gateway**

\*Go to VPC and click on create internet gateway

\*Enter name for IGW

\*Click on create IGW

\*Then go to route table select route table click on routes and edit routes

\*Add routes enter 0.0.0.0/0 and enter IGW name and save changes

**Steps to create NAT Gateway**

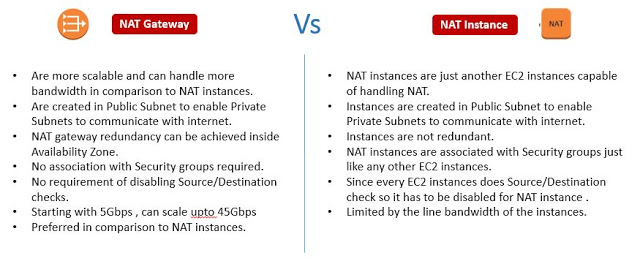
\*Go to vpc select create nat gateway

\*Enter name for nat gateway and select public select and allocate elastic ip and click on create nat gateway.

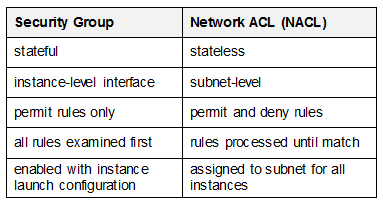
\*Go to route table and select private subnet and click on routes and edit routes

\*Enter 0.0.0.0/0 and enter nat gateway name and save changes.

# Difference between NAT gateway and NAT instance



# Difference between Nacl and security group



**Deny particular ip address in nacl**

\*Go to Nacl and select edit inbound rules

\*Under type select all traffic

\*Under source enter the ip address/32

\*And then select allow or deny and save changes

# VPC ENDPOINTS:

VPC Endpoint allows you to connect the VPC to your AWS services without the help of an Internet Gateway, NAT device, VPN or a AWS Direct Connect connection.

**Types of vpc endpoint https://www.fugue.co/blog/network-security-vpc-endpoints-101**

\***Gateway endpoint**: A [gateway endpoint](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-gateway.html) is a target for a route in a route table to connect VPC resources to S3 or DynamoDB. Traffic is then routed from instances in a subnet to one of these two services.

\***Interface endpoint**: An [interface endpoint](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-interface.html) is an elastic network interface that allows a private IP address in a subnet to connect VPC resources to a number of AWS services, such as CloudFormation, Elastic Load Balancers (ELBs), SNS.

# VPC types

\*Amazon VPC with a single public subnet only.

\*Amazon VPC with public and private subnets.

\*Amazon VPC with public and private subnets and AWS Site-to-Site VPN access.

\*Amazon VPC with a private subnet only and AWS Site-to-Site VPN access.

# VPC peering in cross region

\*Go to Peering connections and click on Create Peering connection.

\*Once provided the following information, click on Create Peering connection

\*In the region selector, provide the acceptor VPCs region.

# Transit gateway:

A transit gateway is a network transit hub that you can use to interconnect our virtual private clouds (VPCs) and on-premises networks.

**Difference between transit gateway and vpc peering**

\*Vpc peering can only be used for two vpc but transit gateway can we used to conncet multiple vpc.

\*We cannot attach vpn and aws direct gateway to vpc peering but we can attach to transit gateway.

# EBS Elastic block storage

Steps to resize ebs volume

\*Go to ebs volume

\*Click on modify🡪Under allocated storage increase the size

\*In aws cli enter resize 2fs /dev/xvdf

\*Enter df -h\* to check

# Cross region ebs attachment

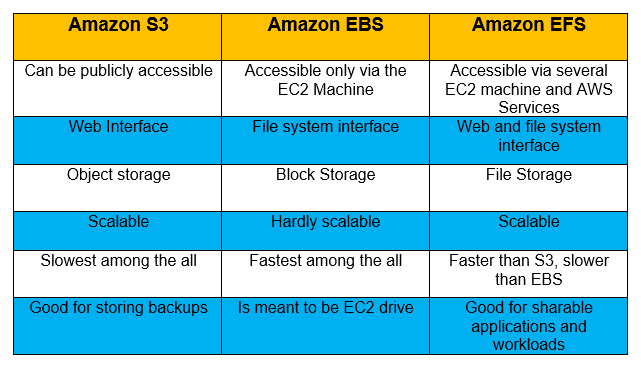
You need to create a snapshot then re-create from that snapshot in the Availability Zone you want it to run

# EFS

Amazon elastic file system is a regional service for storing data within the region and across multiple availability zones.

\*EFS cannot be attached to instance

# Difference between EBS EFS and S3



**Can you mount EBS to multiple EC2?**

**\*** With the new AWS EBS Multi-Attach option, users can now attach a single EBS volume with a maximum of 16 Amazon EC2 instances.

# ELB (Elastic load balancer)

Types of load balancer

\***Application load balancer**: It works at request level and it is used for web application which uses http and https and it operates at the 7th layer .

**How to configure application load balancer**

\*Application load balancer is used for micro service application and can be routed to the application in two ways either by mentioning the path or the port number for example if the hhtp url consists of path it will be redirected to the specific micro service or if consists of port number it will be redirected to the specific microservice.

**Steps to create application load balancer**

\*Create a ec2 instance and under configuration block enter the required script of the application

\*Then create a target group and select instance as target types and http as protocol and 80 as port number and select the vpc

\*Then enter the path of the application

\*Once the target group is created u have to register the target go to target and select the instance it will be registered to the target

\*After this go to load balancer and click on create application load balancer and give name to appliacattion load balancer and scheme should be internet facing so that it can be accessed over the internet else select internal in scheme.

\*Listeners should be hhtp and port number should be 80

\*Select the vpc and availability zone

\*Then select the existing target groups and click on create load balancer.

# NETWORK LOAD BALANCER:

It works at connection level and routes the traffic within the vpc and it can accept millions of request per second and it operates at the 4th layer.

**Steps to create application load balancer**

\*Create a ec2 instance and under configuration block enter the required script of the application

\*Then create a target group and select instance as target types and tcp as protocol and 80 as port number and select the vpc

\*Then enter the path of the application

\*Once the target group is created u have to register the target go to target and select the instance it will be registered to the target

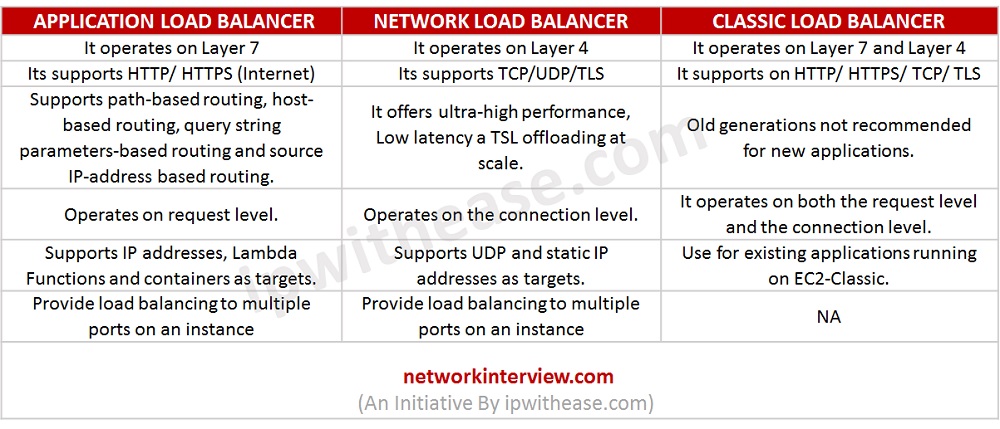
\*After this go to load balancer and click on create network load balancer and give name to application load balancer and scheme should be internet facing so that it can be accessed over the internet else select internal in scheme.

\*Listeners should be tcp and port number should be 80

\*Select the vpc and availability zone

\*Then select the existing target groups and click on create load balancer.

# Difference b/w application load balancer vs network load balancer



# ELASTIC LOAD BALANCING HEALTH CHECKS

1. HealthCheck Protocol

2. HealthCheck Port

3. HealthCheck Path

4. HealthCheck TimeoutSeconds

5. HealthCheck IntervalSeconds

6. HealthyThresholdCount

7. UnhealthyThresholdCount

8. Matcher

# Auto Scaling

Types

1.Horizontal auto scaling

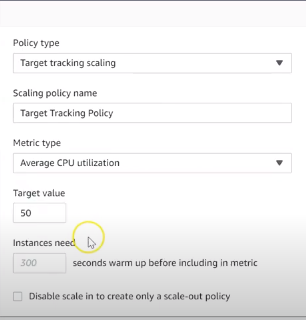
2.Vertical auto scaling

**Difference b/w load balancer and auto scaling**

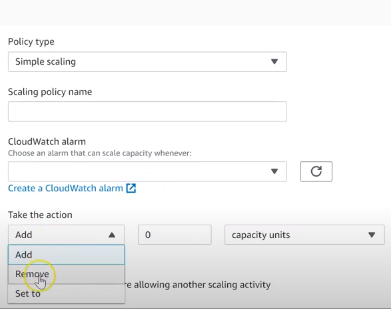
Auto Scaling is used for automatic scaling up and scaling down. Loadbalancer used to distribute the incoming traffic across multiple targets.

# Auto Scaling Policy

1. **Target tracking scaling policies**: We make use of this policy to auto scale if the cpu percent is around 50% and it a used for simple applications.



1. **Step scaling and simple scaling**: It is based out with cloud watch alarms and it will trigger the alarm if cpu utilization is more than 75% and will scale up and if we have set 25% than it will automatically scale down.



1. **Schedule Action**: In these case you can scale up the instance based on the period of time

# Auto scaling setup

\*Create a auto scaling and switch to launch configuration

\*Create a launch configuration in that select the AMI and instance type, volumes and configure the security groups

\*Come back to auto scaling and select the created launch configuration

\*Select vpc and subnets and select the load balancer type

\*Enter min and max scaling and select the scaling policies

\*Then click on create auto scaling.

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

**Launch template:** It captures all the parameters required to launch an ec2 instance

# ROUTE 53

**Latency Routing Policy**: We use this type of policy when we have resources in multiple aws region and you have to route the traffic that provides best latency

**Weighted Routing Policy**: We use this policy to route the traffic to multiple resources in proportions that we specify.

# Static Web Page

A static website consists of a series of HTML files, each one representing a physical page of a website

**Configuring Your S3 Bucket for Static Website Hosting**

* + Navigate to S3 in the AWS Console.
  + Click into your bucket.
  + Click the “Properties” section.
  + Click the “Static website hosting” option.
  + Select “Use this bucket to host a website”.
  + Enter “index.html” as the Index document

# How S3 Versioning works

Mainly used to keep the multiple Version of a object in a single bucket, and enable you to restore objects when it is accidentally deleted.

•for eg:- if you delete an object, amazon S3 inserts an delete mark which becomes as current object version and we can recover the previous version of object.

By default, when we create a S3 bucket the bucket is un versioned we need to enable it

when you enabled versioning, Amazon S3 which automatically generate the unique ID for the object is being stored.

for eg. there is a bucket you if you add an object (xyz.jpg) for the 1st time the unique ID will be created on the 2nd time. also, if you add the same object (xyz.jpeg) it will create a new unique id will be generated

If Versioning is not enabled the object will be get replaced