AWS SNS (SIMPLE NOTIFICATION SERVICE)

- ➤ SNS is a fast, flexible, fully managed PUSH notification service.
- ➤ It is a web service that delivery or sending of messages to subscribing endpoints or clients.
- ➤ It allows for sending individual messages of fan-out messages to a large number of recipients or to other distributed AWS services.
- ➤ Messages published to an SNS topics will be delivered to the subscriber immediately.
- ➤ Inexpensive, pay-as-you-go model with no upfront cost.
- ➤ Reliable: at least three copies of the data are store across multiple AZ in same region.
- ➤ It is a way of sending messages. When you are using auto scaling, it triggers an SNS service which will email you that "your EC2 instance is growing".

Publisher: publishers are also known as produce and send the message to the SNS which is a logical access point.

Subscriber: subscribers such as web servers, email addresses, Amazon SQS queues, AWS lambda receive the message of notification from the SNS over one of the supported protocols (Amazon SQS, email, lambda, HTTPS, SMS).

SNS Topic:
- ➤ It is a logical access point and communication channel.
- ➤ Each topic has a unique name.
- ➤ A topic name is limited to 256 alphanumeric character.
- ➤ The topic name must be unique with in the AWS account.
- ➤ Each topic is assigned an AWS ARN once it gets created.
- ➤ A topic can support subscribers and notification deliveries over multiple protocols.
- ➤ Message/ request published to a single topic can be delivered over multiple protocols as configured when creating each subscriber.
- ➤ Delivery format/ transport protocols (endpoints.), SMS, e-mail, email: JSON –for applications, HTTP/ HTTPS, SQS, AWS lambda.
- ➤ When using Amazon SNS, you (as the owner) create a topic and control access to it by defining access policies that determine which publishers and subscribers can communicate with the topic.
- ➤ Instead of including a specific destination address in each message to topics that they have created or to topics they have permission to publish to.
- ➤ Amazon SNS matches the topic to a list of subscribers who have subscribed to that topic and delivers the message to each of these subscriber.
- ➤ Each topic has a unique name that identifies the Amazon SNS endpoint for publishers to past message and subscribers to register for notification.

- ➢ Subscriber receive all messages published to the topics to which they subscribe, and all subscribers to a topic receive the same message.
- ➢ By default only the topic owner (who created it) can publish to the SNS topic.
- ➢ The owner can set/ change permission to one of more users (with valid AWS ID) to publish to his topic.
- ➢ Only the owner of the topic can grant/ change permission for the topic.
- ➢ Subscriber can be those with/ without AWS ID, only subscriber with AWS ID can request subscription.
- ➢ Both publishers and subscriber can use SSL to help secure the channel to send and receive message.
- ➢ Supported push notification platforms:
  - • Amazon Device Messaging
  - • Apple Push Notification Service.
  - • Google Cloud Messaging
  - • Windows Push Notification Service
  - • Baidu Cloud Push for Android
- ➢ SNS topic can have subscribers from any supported push notification platforms as well as any other endpoint type such as SMS or Email.
- ➢ When you publish a notification to a topic SNS will send identical copies of that, message to each endpoint subscribed to the topic.

Amazon SNS Alternatives:

a. Amazon Kinesis Data Stream
b. Amazon Managed Queue Service (AWS MQS)
c. Apache Kafka
d. Twilio
e. Pusher

Amazon SNS Pricing:

a. Publish action: each 64kb of request payload count as one request. So, 256kb payload will charged as four request.
b. Mobile push notification: for e.g: $0.50/ million request
c. SMS: price depends on country
d. E-mail: $2/100,000
e. HTTP/s notification : $ 0.60/ million
f. SQS and lambda calls are free. These are charged at SQS and lambda rates.
g. Data Transfer