**AWS SNS**

SNS stands for Simple Notification Service.

It is a web service which makes it easy to set up, operate, and send a notification from the cloud.

It provides developers with the highly scalable, cost-effective, and flexible capability to publish messages from an application and sends them to other applications.

It is a way of sending messages. When you are using AutoScaling, it triggers an SNS service which will email you that "your EC2 instance is growing".

SNS can also send the messages to devices by sending push notifications to Apple, Google, Fire OS, and Windows devices, as well as Android devices in China with Baidu Cloud Push.

Besides sending the push notifications to the mobile devices, Amazon SNS sends the notifications through SMS or email to an Amazon Simple Queue Service (SQS), or to an HTTP endpoint.

SNS notifications can also trigger the Lambda function. When a message is published to an SNS topic that has a Lambda function associated with it, Lambda function is invoked with the payload of the message. Therefore, we can say that the Lambda function is invoked with a message payload as an input parameter and manipulate the information in the message and then sends the message to other SNS topics or other AWS services.

Amazon SNS allows you to group multiple recipients using topics where the topic is a logical access point that sends the identical copies of the same message to the subscribe recipients.

Amazon SNS supports multiple endpoint types. For example, you can group together IOS, Android and SMS recipients. Once you publish the message to the topic, SNS delivers the formatted copies of your message to the subscribers.

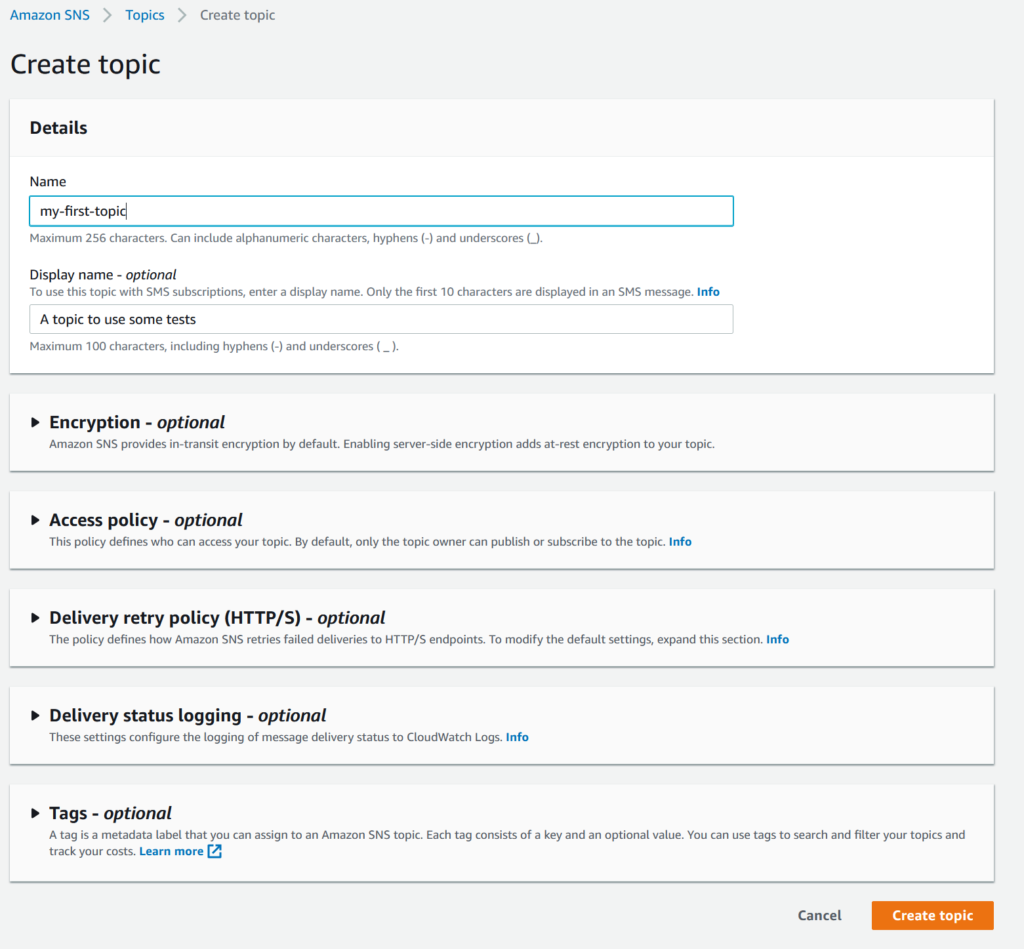
To prevent the loss of data, all messages published to SNS are stored redundantly across multiple availability zones.

**At the heart of SNS: the topic**

A topic is something like a channel, where you have one (or more) modules writing on it and one (or more) modules reading from it.

Applications reading from the module are subscribed to the topic. Instead, applications writing on it are publishing on it.

When creating a topic, it will ask you for some settings. We can leave everything to default and only provide name and display name.



**Using the topic**

Once you create a topic, you will end up on its dashboard page. From there, you can see what its subscribers are and even publish a notification in the topic.

that manages sending messages to the subscribing endpoint. There are two clients of SNS:

Subscribers

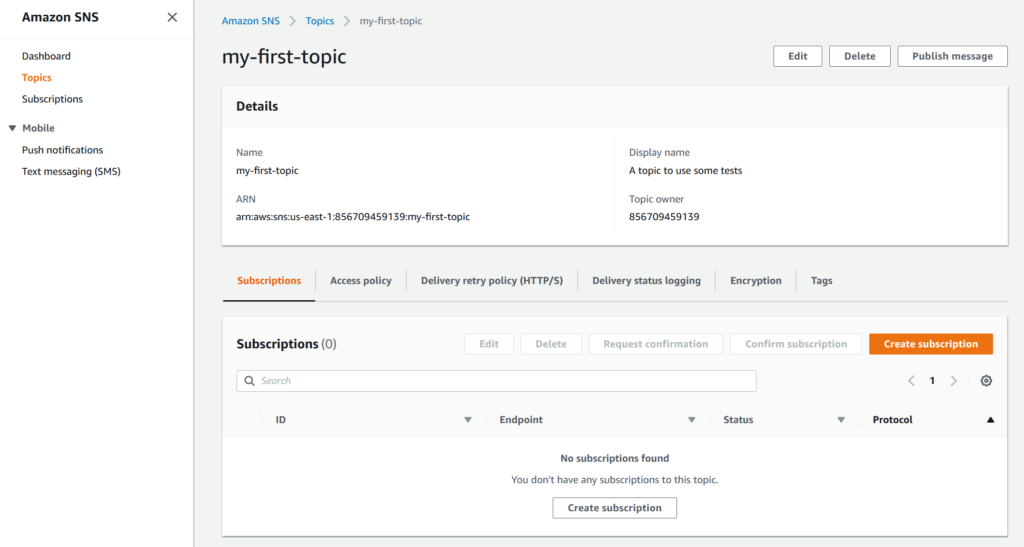
Publishers

**Publishers**

Publishers are also known as producers that produce and send the message to the SNS which is a logical access point.

**Subscribers**

Subscribers such as web servers, email addresses, Amazon SQS queues, and AWS Lambda functions receive the message or notification from the SNS over one of the supported protocols (Amazon SQS, email, Lambda, HTTP, and SMS).



**Benefits of SNS**

**Instantaneous delivery**

SNS is based on push-based delivery. This is the key difference between SNS and SQS. SNS is pushed once you publish the message in a topic and the message is delivered to multiple subscribers.

**Flexible**

SNS supports multiple endpoint types. Multiple endpoint types can receive the message over multiple transport protocols such as email, SMS, Lambda, Amazon SQS, HTTP, etc.

**Inexpensive**

SNS service is quite inexpensive as it is based on pay-as-you-go model, i.e., you need to pay only when you are using the resources with no up-front costs.

**Ease of use**

SNS service is very simple to use as Web-based AWS Management Console offers the simplicity of the point-and-click interface.

**Simple Architecture**

SNS is used to simplify the messaging architecture by offloading the message filtering logic from the subscribers and message routing logic from the publishers. Instead of receiving all the messages from the topic, SNS sends the message to subscriber-only of their interest.