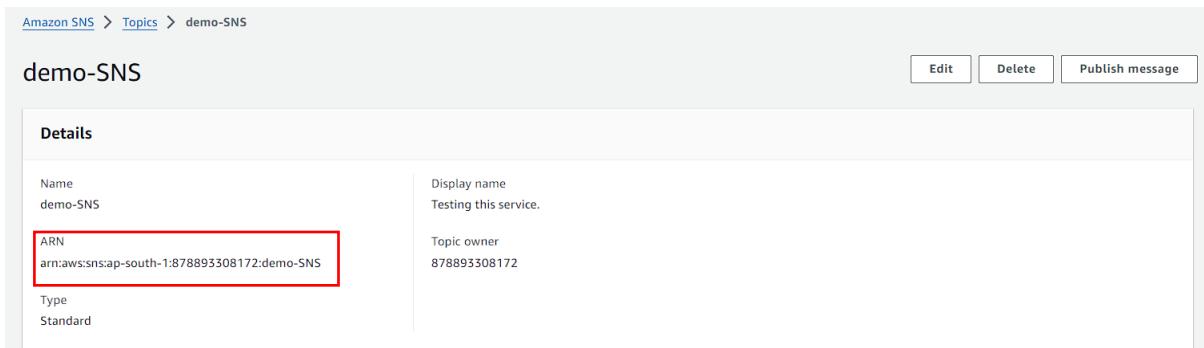


Simple Notification Service

1. Now in this you are going to setup program for SNS.
2. For that again you need to download the zip file from GitHub and unzip it then open it in Visual Studio.
3. For this lab you should be having a topic in SNS then copy its ARN and paste it in visual studio.



Amazon SNS > Topics > demo-SNS

demo-SNS

Details

Name demo-SNS	Display name Testing this service.
ARN arn:aws:sns:ap-south-1:878893308172:demo-SNS	Topic owner 878893308172
Type Standard	

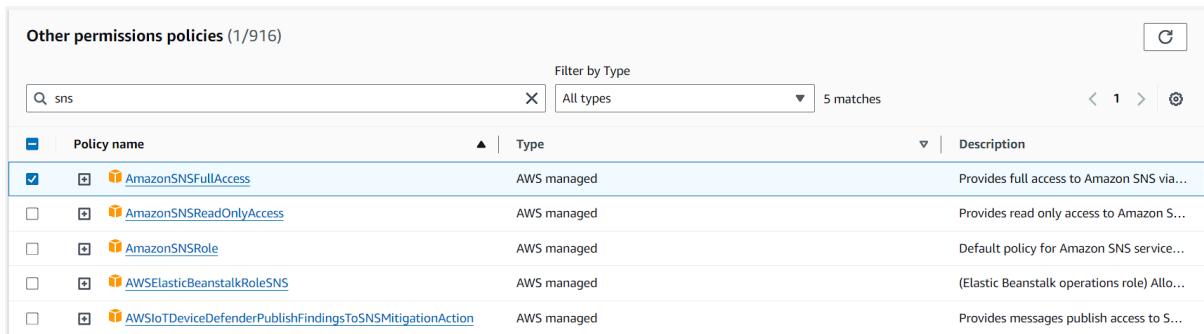
Edit Delete Publish message

```
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Newtonsoft.Json;
using SendMessage_SNS;

string topicARN = "arn:aws:sns:ap-south-1:878893308172:demo-SNS";

IAmazonSimpleNotificationService snsClient = new AmazonSimpleNotificationServiceClient();
```

4. Now open your IAM and add policy for SNS in your role which is running in your Lambda function which you used in previous lab for SQS.
5. So, you need to add this permission in both you IAM role and User.



Other permissions policies (1/916)

Filter by Type

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonSNSFullAccess	AWS managed	Provides full access to Amazon SNS via...
<input type="checkbox"/> AmazonNSReadOnlyAccess	AWS managed	Provides read only access to Amazon S...
<input type="checkbox"/> AmazonSNSRole	AWS managed	Default policy for Amazon SNS service...
<input type="checkbox"/> AWS>ElasticBeanstalkRoleSNS	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/> AWSIoTDeviceDefenderPublishFindingsToSNSMitigationAction	AWS managed	Provides messages publish access to S...

6. No go back to SNS and create a new subscription.
7. Choose protocol as amazon SQS and create your subscription.
8. Here you are subscribing your SQS queue onto your SNS topic.

Create subscription

Details

Topic ARN

Protocol
The type of endpoint to subscribe

Endpoint
Only Amazon SQS standard queues will be listed and can receive notifications from an Amazon SNS standard topic.

Enable raw message delivery

Info After your subscription is created, you must confirm it. [Info](#)

9. Afterwards go to SQS and there scroll down and select your subscription ARN and subscribe to SNS topic.

SNS subscriptions | Lambda triggers | EventBridge Pipes | Dead-letter queue | Monitoring | Tagging | Access policy | Encryption | Dead-letter queue redrive tasks

Subscription region

SNS subscriptions (1) [Info](#)

Subscription ARN Topic ARN

arn:aws:sns:ap-south-1:878893308172:demo-SNS arn:aws:sns:ap-south-1:878893308172:demo-SNS

10. Now select your topic and click on Save.

Subscribe to Amazon SNS topic [Info](#)

Amazon SNS topic
To allow your queue to receive messages from an Amazon SNS topic, subscribe it to an Amazon SNS topic.

Specify an Amazon SNS topic available for this queue.

11. Once this is done go to the program and run the program locally.
12. Then go back to AWS Console and navigate to CloudWatch and open your logs for your lambda function.
13. You will see multiple logs and if you will open them, you can see your data in them.

Log streams Tags Anomaly detection Metric filters Subscription filters Contributor Insights Data protection

Log streams (5)

Filter log streams or try prefix search Delete Create log stream Search all log streams

<input type="checkbox"/> Log stream	Last event time
<input type="checkbox"/> 2024/02/13/[\$LATEST]0d2d00b8967f4802880784cd0c8d2fb	2024-02-13 12:04:20 (UTC+05:30)
<input type="checkbox"/> 2024/02/13/[\$LATEST]bc0730cd6668459ca06db9a7e556a8d	2024-02-13 12:04:20 (UTC+05:30)
<input type="checkbox"/> 2024/02/13/[\$LATEST]ad4f464d44564cc2a7e27913636de159	2024-02-13 12:04:20 (UTC+05:30)
<input type="checkbox"/> 2024/02/13/[\$LATEST]aeaba23fbcbfc4f27997f46c7414272a5	2024-02-13 12:04:20 (UTC+05:30)
<input type="checkbox"/> 2024/02/13/[\$LATEST]a9dd9723263342a5afaf6a96ad544e15	2024-02-13 12:04:20 (UTC+05:30)

▶	Timestamp	Message
No older events at this moment. Retry		
▶	2024-02-13T12:04:20.200+05:30	INIT_START Runtime Version: python:3.12.v18 Runtime Version ARN: arn:aws:lambda:ap-south-1:runtime:776a3759221679a634181f858871d5514dc74a176f78bc535f822e93..
▶	2024-02-13T12:04:20.284+05:30	START RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844 Version: \$LATEST
▶	2024-02-13T12:04:20.284+05:30	1
▶	2024-02-13T12:04:20.284+05:30	Product1
▶	2024-02-13T12:04:20.284+05:30	5
▶	2024-02-13T12:04:20.286+05:30	END RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844
▶	2024-02-13T12:04:20.286+05:30	REPORT RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844 Duration: 1.00 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 33 MB Init Duration: 82..
▶	2024-02-13T12:04:20.304+05:30	START RequestId: 63466179-acd4-5b52-86ac-3f3ca30264c6 Version: \$LATEST
▶	2024-02-13T12:04:20.305+05:30	2
▶	2024-02-13T12:04:20.305+05:30	Product2
▶	2024-02-13T12:04:20.305+05:30	10
▶	2024-02-13T12:04:20.324+05:30	END RequestId: 63466179-acd4-5b52-86ac-3f3ca30264c6
▶	2024-02-13T12:04:20.324+05:30	REPORT RequestId: 63466179-acd4-5b52-86ac-3f3ca30264c6 Duration: 20.00 ms Billed Duration: 20 ms Memory Size: 128 MB Max Memory Used: 34 MB
▶	2024-02-13T12:04:20.327+05:30	START RequestId: 3f93677a-ecc5-56fd-bae5-393fe362c310 Version: \$LATEST
▶	2024-02-13T12:04:20.328+05:30	5
▶	2024-02-13T12:04:20.328+05:30	Product5
▶	2024-02-13T12:04:20.328+05:30	25

[Back to top](#) ^