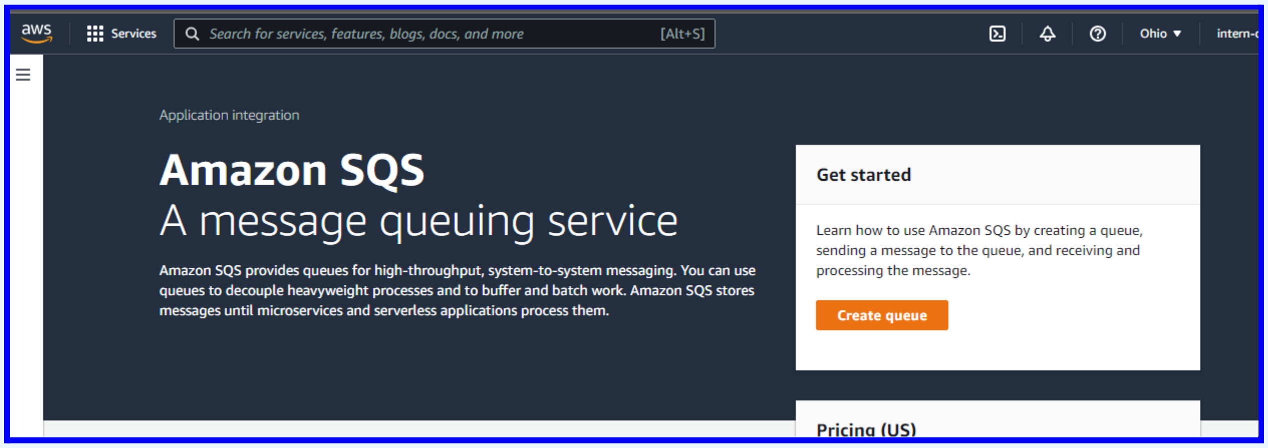
(Optional)

- **Create standard SQS.**
- **Add this SQS as target in above created Event Bridge rule (in addition to existing SNS)**
- **Add lambda trigger in SQL to sendEmail lambda function.**

SQS Home Page



Application integration

Amazon SQS
A message queuing service

Amazon SQS provides queues for high-throughput, system-to-system messaging. You can use queues to decouple heavyweight processes and to buffer and batch work. Amazon SQS stores messages until microservices and serverless applications process them.

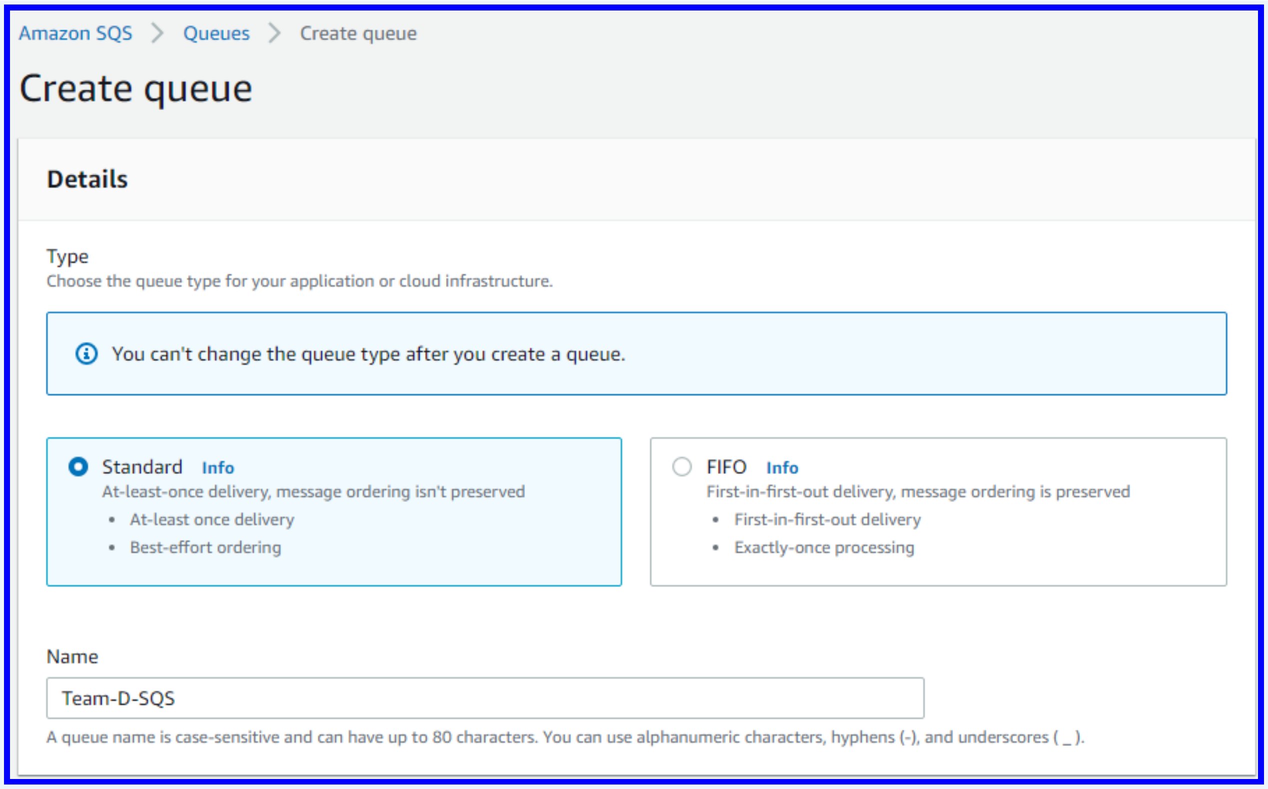
Get started

Learn how to use Amazon SQS by creating a queue, sending a message to the queue, and receiving and processing the message.

Create queue

Pricing (US)

Creating Standard Queue



Amazon SQS > Queues > Create queue

Create queue

Details

Type
Choose the queue type for your application or cloud infrastructure.

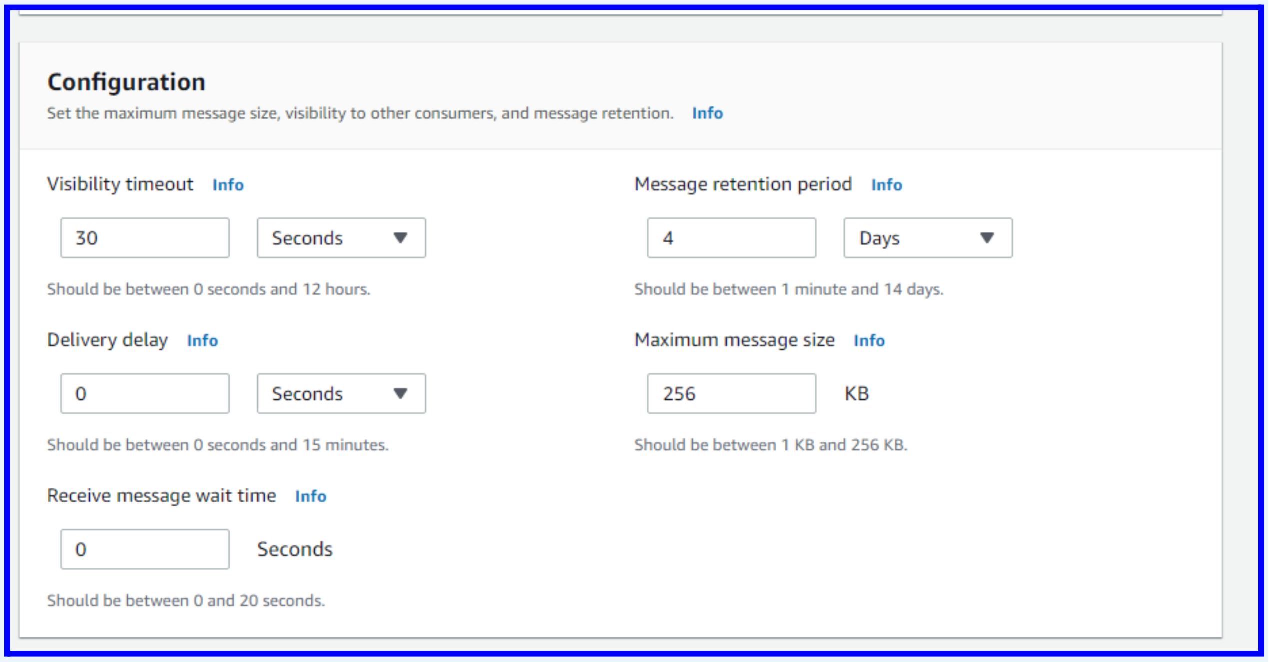
You can't change the queue type after you create a queue.

[✓ Standard] Info
At-least-once delivery, message ordering isn't preserved
- At-least once delivery
- Best-effort ordering

FIFO Info
First-in-first-out delivery, message ordering is preserved
- First-in-first-out delivery
- Exactly-once processing

Name
Team-D-SQS
A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).

Using Default COnfiguration



Configuration

Set the maximum message size, visibility to other consumers, and message retention. Info

Visibility timeout Info

30 Seconds

Should be between 0 seconds and 12 hours.

Message retention period Info

4 Days

Should be between 1 minute and 14 days.

Delivery delay Info

0 Seconds

Should be between 0 seconds and 15 minutes.

Maximum message size Info

256 KB

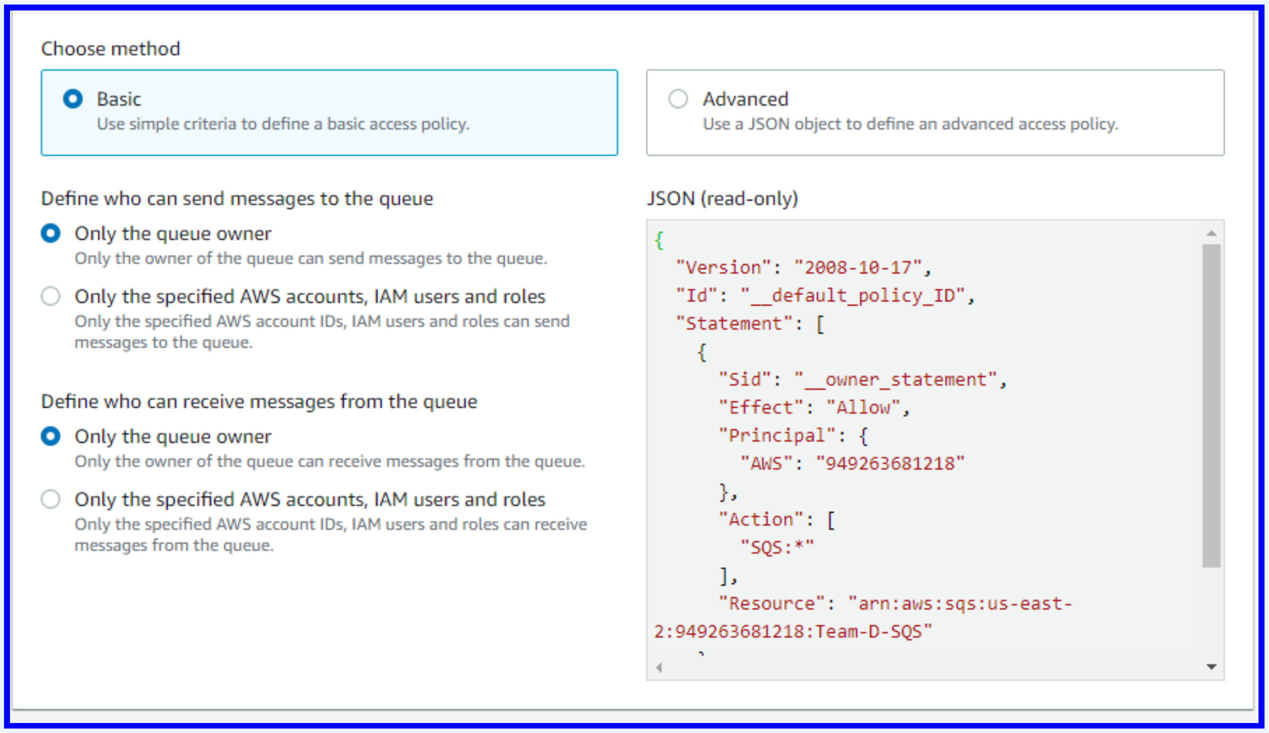
Should be between 1 KB and 256 KB.

Receive message wait time Info

0 Seconds

Should be between 0 and 20 seconds.

Using Basic Method



Choose method

[✓ Basic] Use simple criteria to define a basic access policy.

Advanced — Use a JSON object to define an advanced access policy.

Define who can send messages to the queue

[✓ Only the queue owner] Only the owner of the queue can send messages to the queue.

Only the specified AWS accounts, IAM users and roles — Only the specified AWS account IDs, IAM users and roles can send messages to the queue.

Define who can receive messages from the queue

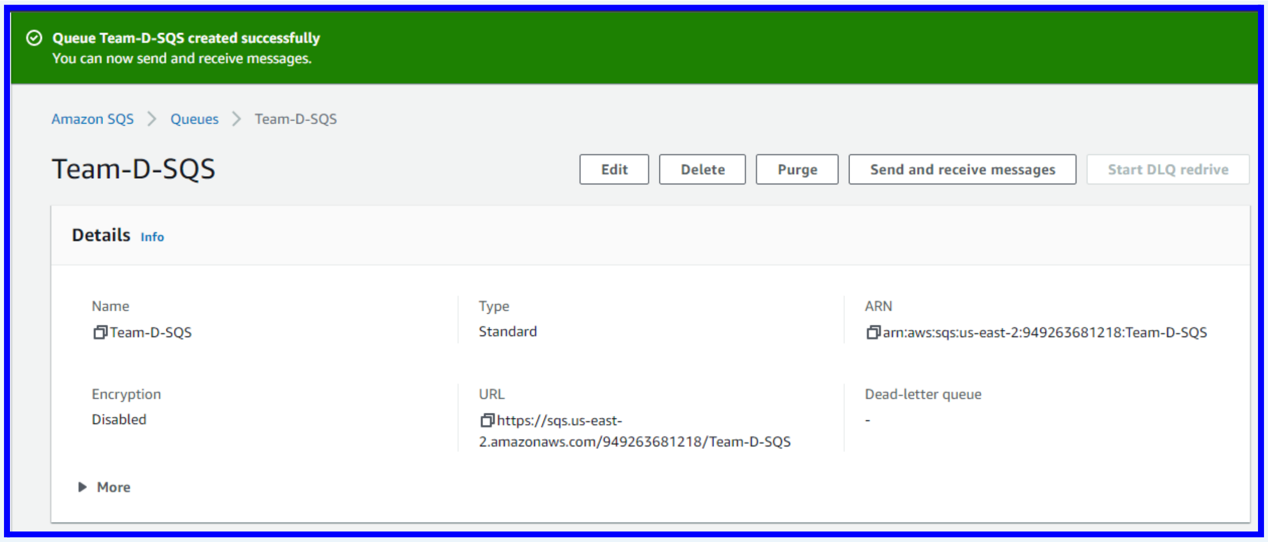
[✓ Only the queue owner] Only the owner of the queue can receive messages from the queue.

Only the specified AWS accounts, IAM users and roles — Only the specified AWS account IDs, IAM users and roles can receive messages from the queue.

JSON (read-only)

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__owner_statement",
      "Effect": "Allow",
      "Principal": {
        "AWS": "949263681218"
      },
      "Action": [
        "SQS:*"
      ],
      "Resource": "arn:aws:sqs:us-east-
2:949263681218:Team-D-SQS"
```

Standard Queue Created



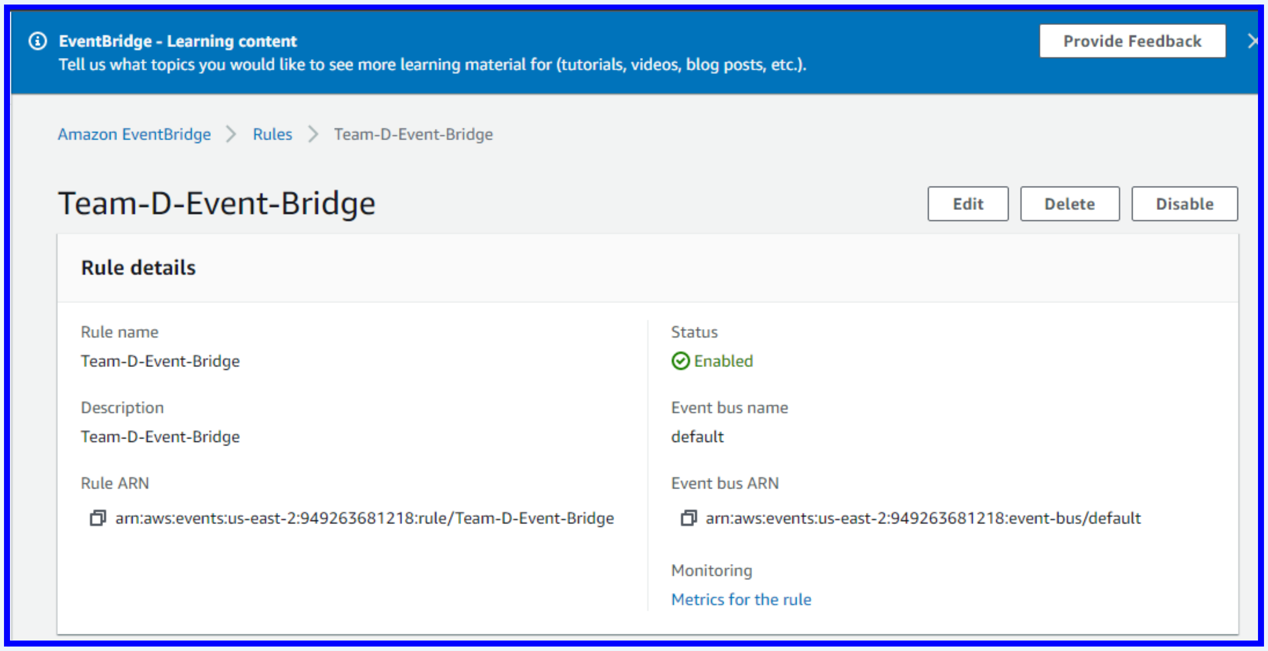
Queue Team-D-SQS created successfully
You can now send and receive messages.

Team-D-SQS

Details Info

Name	Type	ARN
Team-D-SQS	Standard	arn:aws:sqs:us-east-2:949263681218:Team-D-SQS
Encryption	URL	Dead-letter queue
Disabled	https://sqs.us-east-2.amazonaws.com/949263681218/Team-D-SQS	-

Updating Event Bridge with SQS as a target



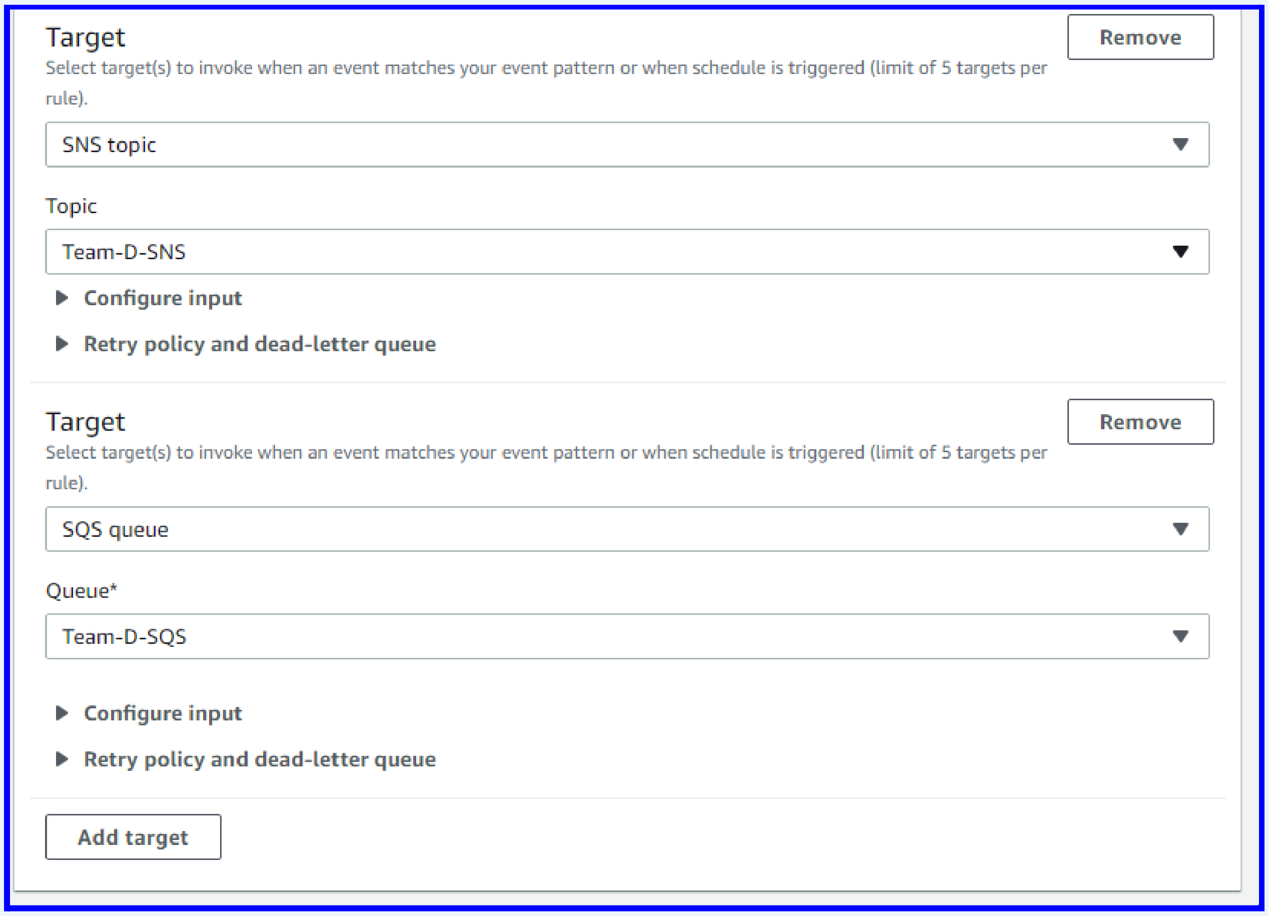
EventBridge - Learning content
Tell us what topics you would like to see more learning material for (tutorials, videos, blog posts, etc.).

Team-D-Event-Bridge

Rule details

Rule name	Status
Team-D-Event-Bridge	Enabled
Description	Event bus name
Team-D-Event-Bridge	default
Rule ARN	Event bus ARN
arn:aws:events:us-east-2:949263681218:rule/Team-D-Event-Bridge	arn:aws:events:us-east-2:949263681218:event-bus/default
	Monitoring
	Metrics for the rule

Adding SQS as Target too alongside with SNS



Target

Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule).

SNS topic

Topic

Team-D-SNS

▶ Configure input

▶ Retry policy and dead-letter queue

Target

Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule).

SQS queue

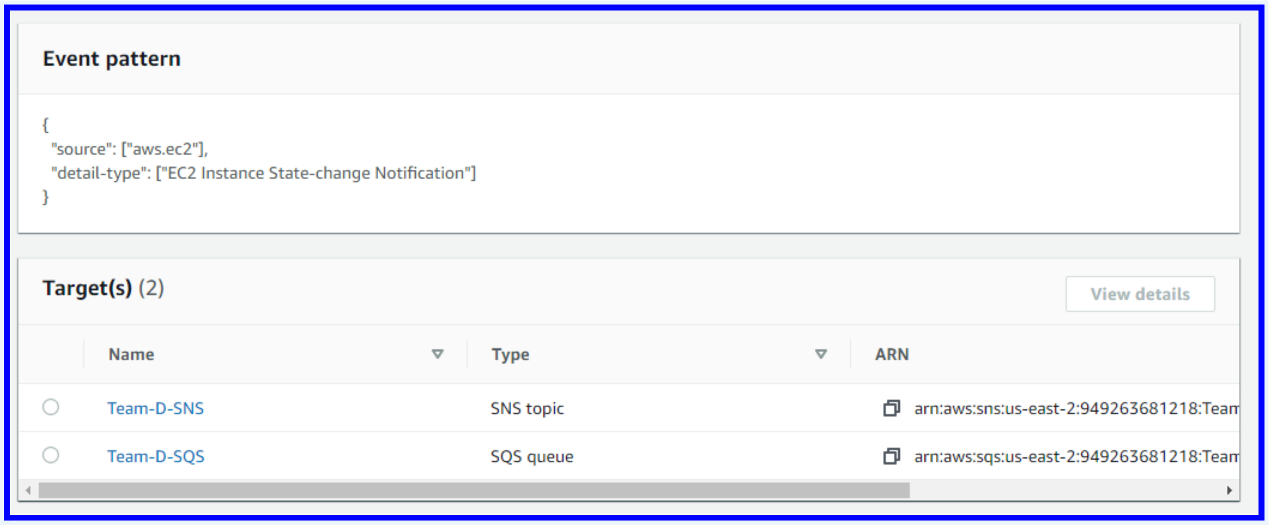
Queue*

Team-D-SQS

▶ Configure input

▶ Retry policy and dead-letter queue

We can see two targets in Event Bridge



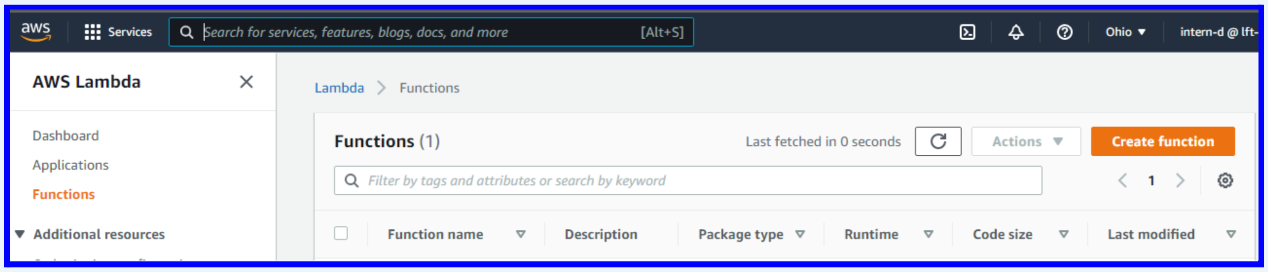
Event pattern

```
{
 "source": ["aws.ec2"],
 "detail-type": ["EC2 Instance State-change Notification"]
}
```

Target(s) (2)

Name	Type	ARN
Team-D-SNS	SNS topic	arn:aws:sns:us-east-2:949263681218:Team
Team-D-SQS	SQS queue	arn:aws:sqs:us-east-2:949263681218:Team

Lambda function Home Page

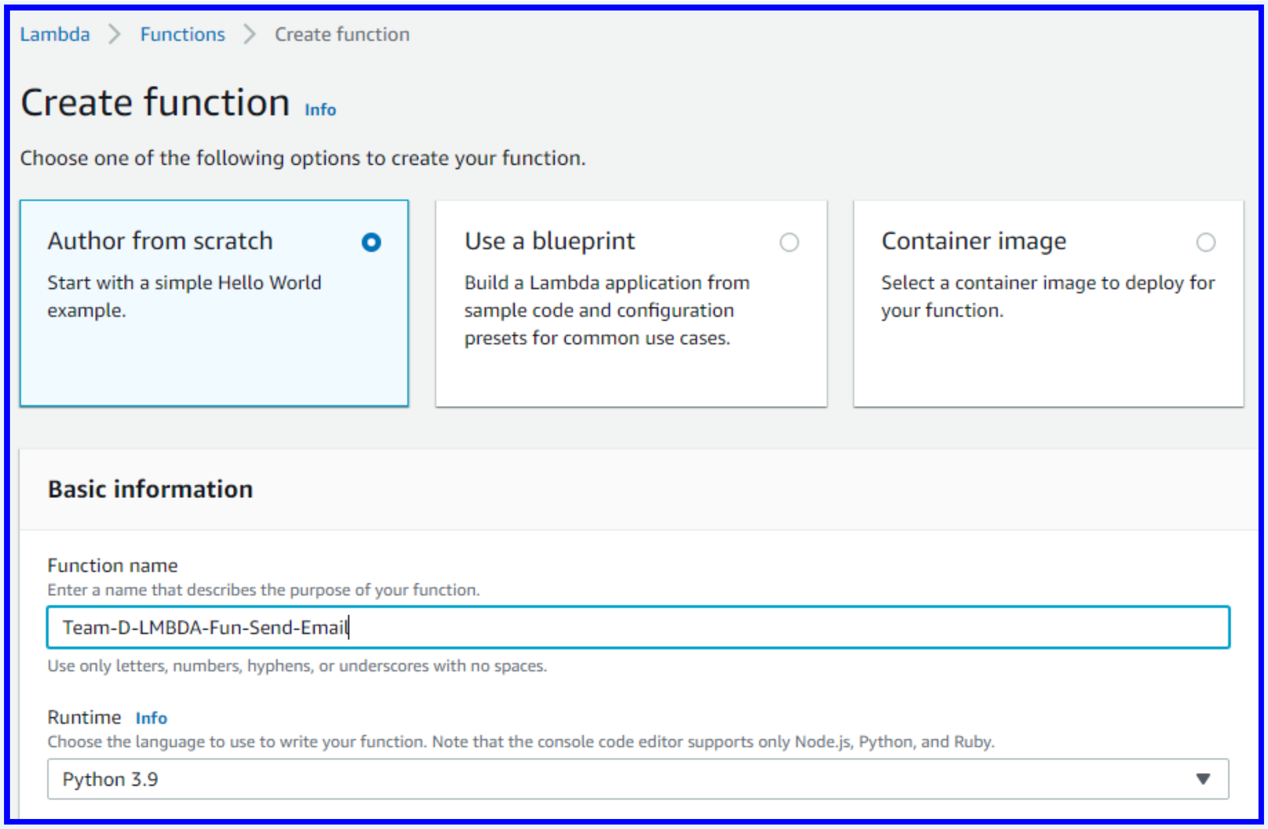


Functions (1)

Last fetched in 0 seconds

Function name	Description	Package type	Runtime	Code size	Last modified

Creating Send Email Function



Create function Info

Choose one of the following options to create your function.

[✓ Author from scratch] Start with a simple Hello World example.

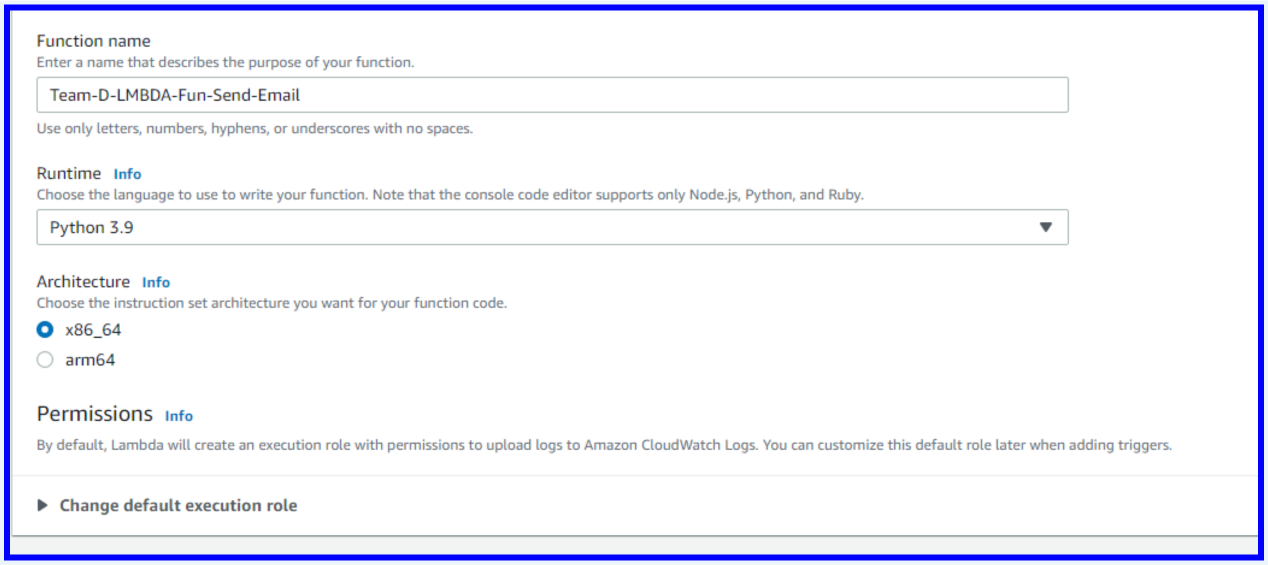
Use a blueprint — Build a Lambda application from sample code and configuration presets for common use cases.

Container image — Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
Team-D-LMBDA-Fun-Send-Email
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.9



Function name
Enter a name that describes the purpose of your function.
Team-D-LMBDA-Fun-Send-Email
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.9

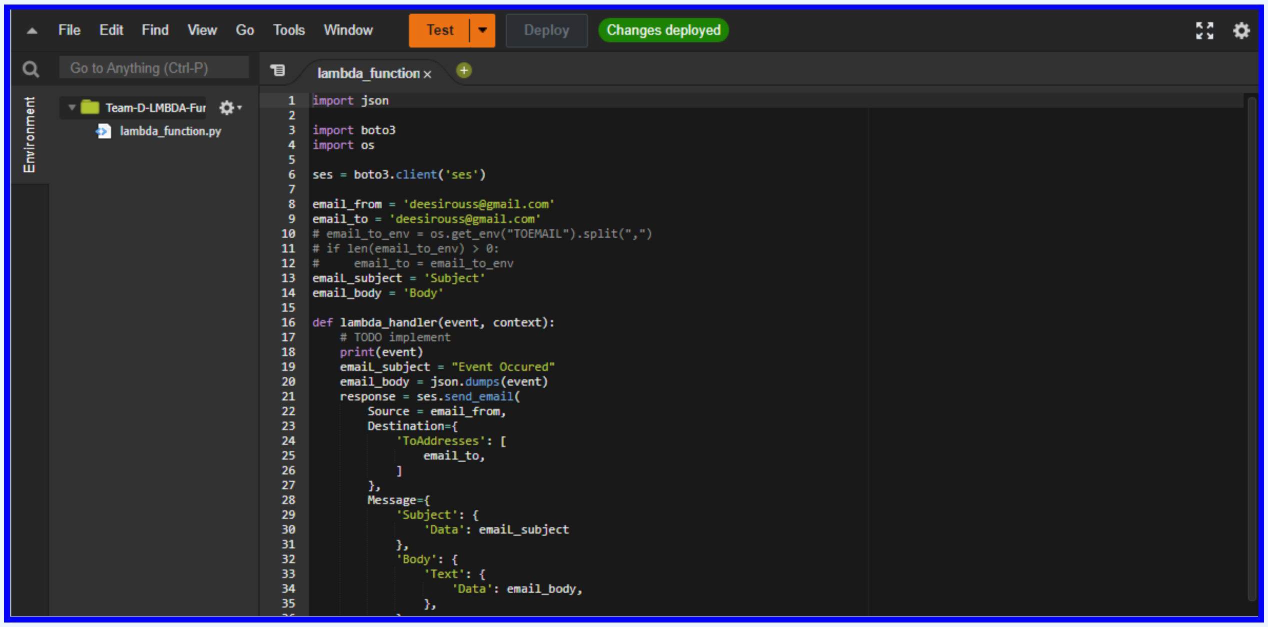
Architecture Info
Choose the instruction set architecture you want for your function code.
[✓ x86_64]
arm64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

Change default execution role

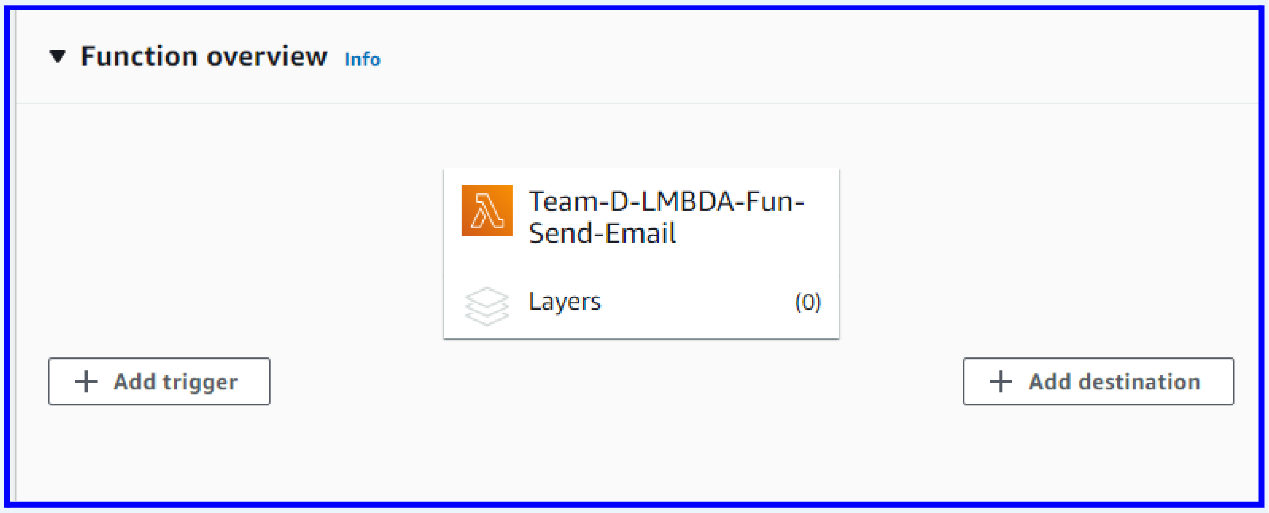
<u>Using the same code used by Amit Joshi Dai and deployed</u>

Where email address is changed

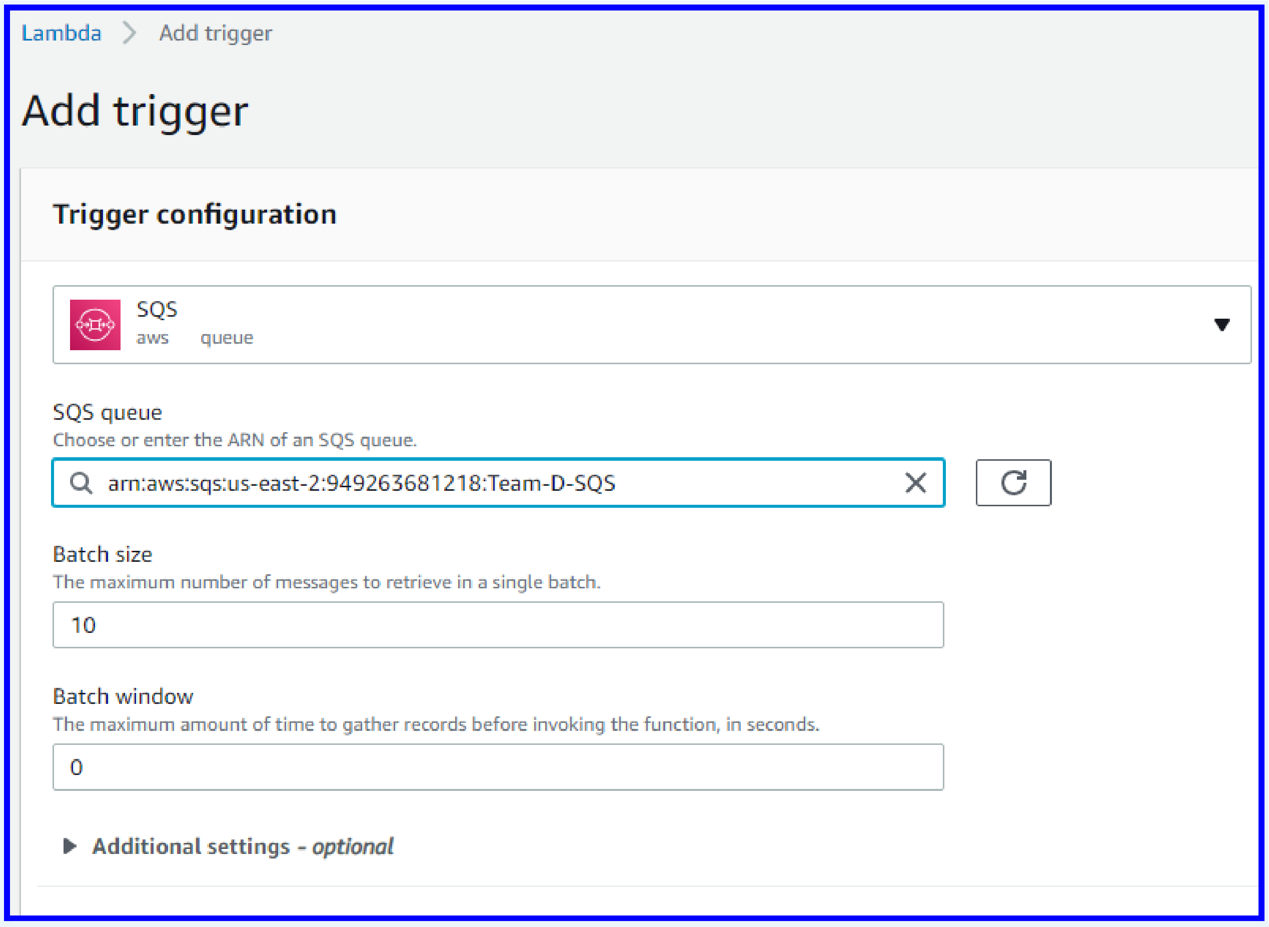


```python
import json

import boto3
import os

ses = boto3.client('ses')

email_from = 'deesirouss@gmail.com'
email_to = 'deesirouss@gmail.com'
# email_to_env = os.get_env("TOEMAIL").split(",")
# if len(email_to_env) > 0:
#     email_to = email_to_env
emaiL_subject = 'Subject'
email_body = 'Body'

def lambda_handler(event, context):
    # TODO implement
    print(event)
    emaiL_subject = "Event Occured"
    email_body = json.dumps(event)
    response = ses.send_email(
        Source = email_from,
        Destination={
            'ToAddresses': [
                email_to,
            ]
        },
        Message={
            'Subject': {
                'Data': emaiL_subject
            },
            'Body': {
                'Text': {
                    'Data': email_body,
                },
```

Adding SQS Trigger for SQS



Selecting SQS as trigger



Add trigger

Trigger configuration

SQS
aws queue

SQS queue
Choose or enter the ARN of an SQS queue.
arn:aws:sqs:us-east-2:949263681218:Team-D-SQS

Batch size
The maximum number of messages to retrieve in a single batch.
10

Batch window
The maximum amount of time to gather records before invoking the function, in seconds.
0

▶ Additional settings - *optional*