Xerris Bootcamp Series

Introduction to AWS SNS

Simple Notification Service

xerris

ACCOLITE DIGITAL
Transforming The Future, Now

What is AWS Simple Notification Service (SNS)

Simple Notification Service

- Fully managed messaging system.
- Used for service-to-service communication
- Used for service-to-person communication

- Decouples your applications.
- Systems can use a pub/sub event-driven mechanism.
- SNS Allows services to scale as their demand fluctuates independently.

Simple Notification Service (SNS)

Service to Service Communications

- Send messages between applications as part of a workflow
- Services scale automatically as demand increases

Service to Person Communications

- Send SMS text messages to users
- Send email messages to users

Simple Notification Service (SNS)

SNS is highly available
- SNS Messages are stored redundantly across several servers.
- SNS provides a configurable retry policy.

SNS is durable
- Your SNS Messages are redundantly stored across Amazon SNS servers also.

Simple Notification Service (SNS)

SNS provides a Fanout mechanism
- A single message in a top can Fanout to multiple targets.

SNS Subscriptions
- Subscribers register interest in a topic through a subscription.
- When delivering a message on that topic, the subscribers receive a notification as the message is delivered.
- If the subscriber is not available, once available, the message is delivered.

Message Publisher

Amazon SNS topic

Amazon SQS queue

AWS Lambda function

HTTP endpoint
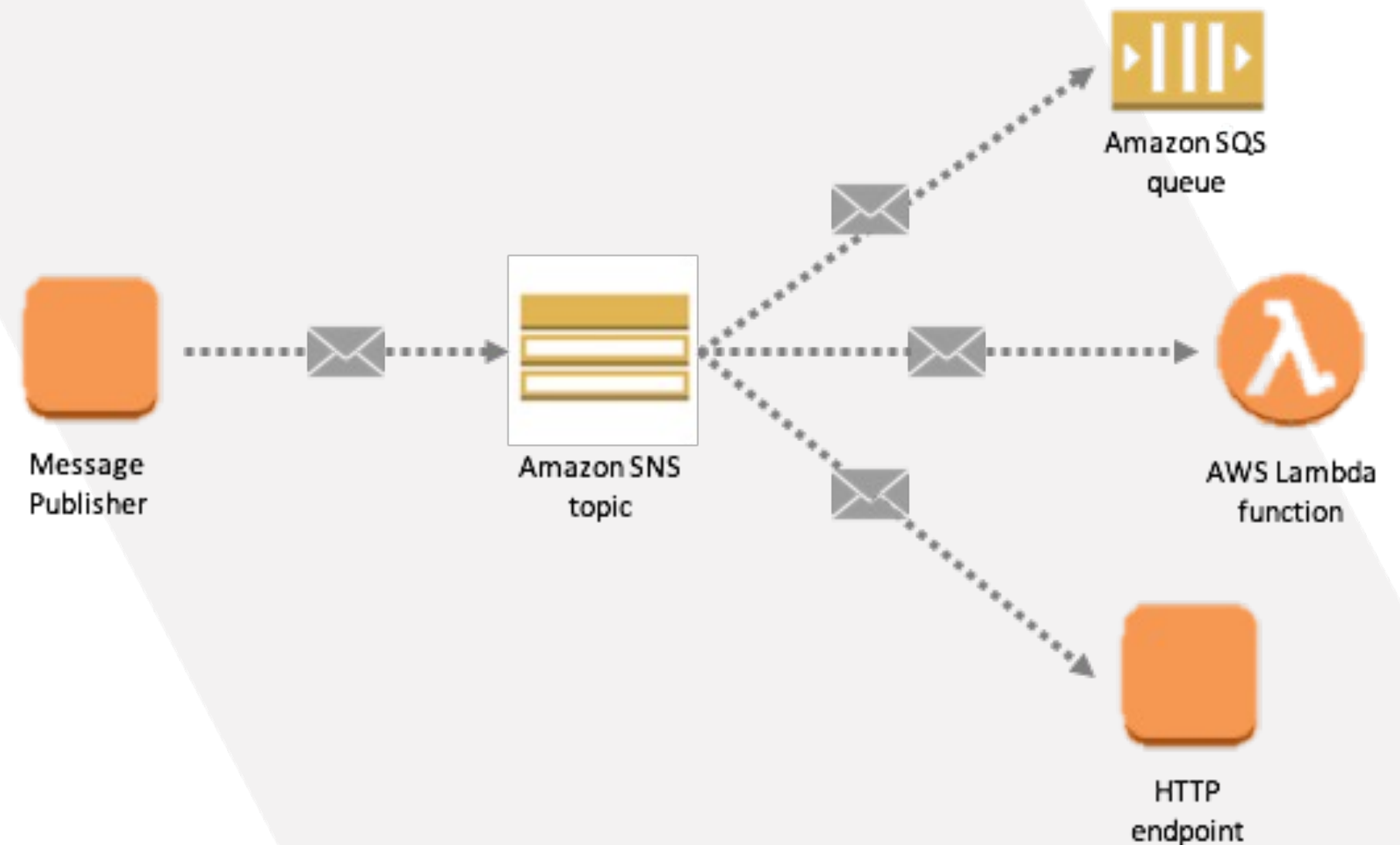
SNS using the AWS CLI

Listing topics (Mac OS)
    aws sns list-topics

Creating a Topic (Mac OS)
    aws sns create-topic --name add-song-topic

Returns:
```
{
    "TopicArn": "arn:aws:sns:us-west-2:123456789:add-song-topic"
}
```

SNS using the AWS CLI

Subscribe to a topic (Mac OS)

```
aws sns subscribe \
--topic-arn arn:aws:sns:us-west-2:123456789:add-song-topic \
--protocol email \
--notification-endpoint my-email@whatever.com
```

Results:
```
{
    "SubscriptionArn": "pending confirmation"
}
```

SNS using the AWS CLI

SNS Subscription Confirmed

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:
arn:aws:sns:us-west-2:████████:add-song-topic:f738b044-b341-4fa6-8cbd-8936ee5aebdc

If it was not your intention to subscribe, <u>click here to unsubscribe</u>.

SNS using the AWS CLI

Publish message to your topic (Mac OS)

```
 aws sns publish --topic-arn arn:aws:sns:us-west-2:123456789:add-song-topic \
--message "Hello, from SNS"
```

Results:
```
{
    "MessageId": "7dc56bff-2248-5642-b559-329972ce8129"
}
```

SNS using the AWS CLI

SNS message received

SNS using the AWS CLI

Deleting your topic (Mac OS)

```
aws sns delete-topic \
--topic-arn arn:aws:sns:us-west-2:123456789:add-song-topic
```

AWS SNS Workshop

To do:

- Create an SNS topic using the CLI.
- Create an email subscription to your topic.
- In DotNet, publish this to a topic when adding a new song.
- Verify your email was received.

Bonus:

- Create an SQS queue and subscribe to the topic.
- Create a Lambda that subscribes to the SQS queue.
- Program the Lambda to add this song to a DynamoDB Table.
- Use Terraform to create the infrastructure for this.