

Hands-on tutorials

Send Fanout Event Notifications



Send Fanout Event Notifications: Hands-on tutorials

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Send Fanout Event Notifications	i
Overview	1
Implementation	1
Congratulations!	20

Send Fanout Event Notifications

Cost to Complete	Free Tier
Services Used	Amazon SNS Amazon SQS
Sending Fanout Event Notifications on AWS Requires an Account	<p>AWS Free Tier includes 1,000,000 publishes for Amazon Simple Notification Service and 1,000,000 requests of Amazon Simple Queue Service.</p> <p>View AWS Free Tier Details »</p> <p>Create a Free Account in Minutes</p>

Overview

In this tutorial, you will implement a fanout messaging scenario using [Amazon Simple Notification Service \(SNS\)](#) and [Amazon Simple Queue Service \(SQS\)](#). In this scenario, messages are "pushed" to multiple subscribers, which eliminates the need to periodically check or poll for updates and enables parallel asynchronous processing of the message by the subscribers.

To illustrate this, we will assume that you are developing a cloud-native application that sends an Amazon SNS message to a topic whenever an order is placed on an online store. The Amazon SQS queues that are subscribed to that topic will each receive identical notifications for the new order.

The AWS services you use in this tutorial are within the [AWS Free Tier](#).

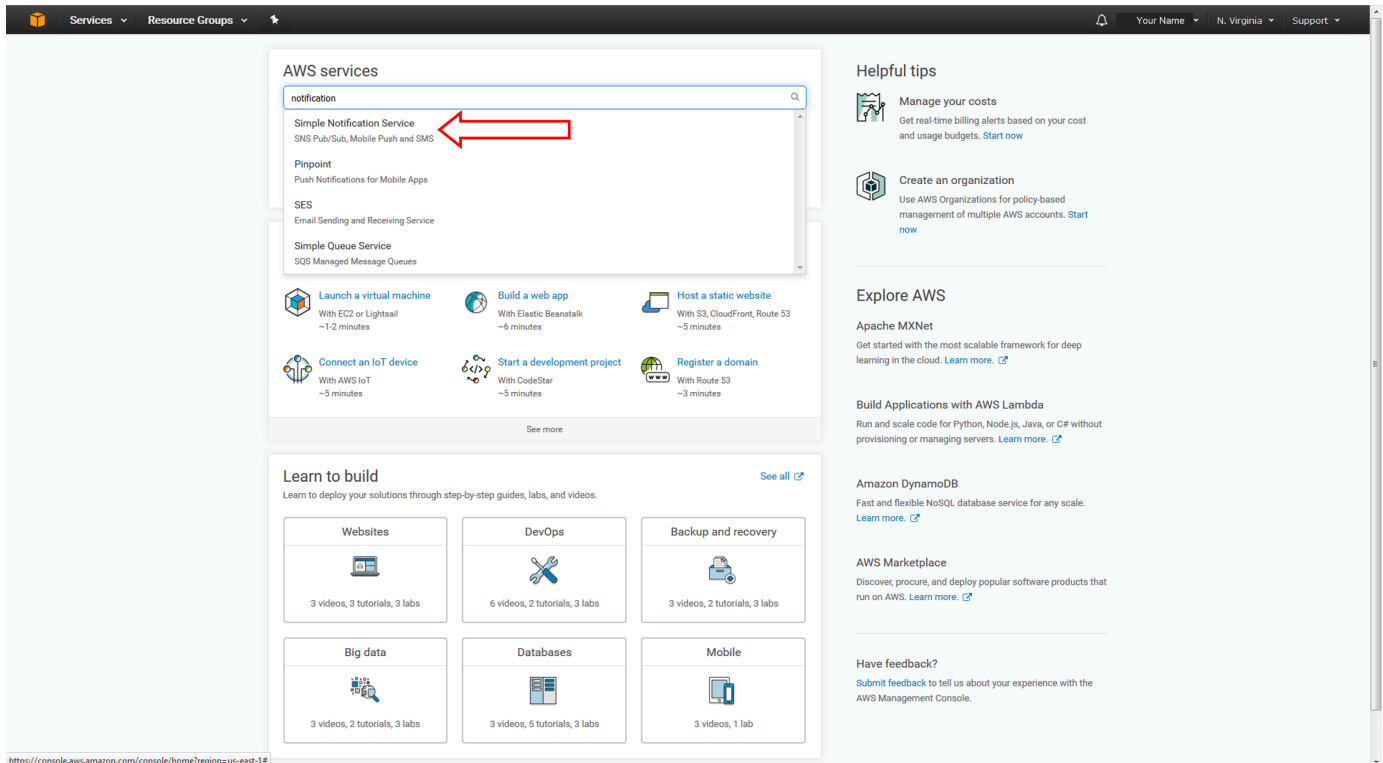
Implementation

Step 1: Open the Amazon SNS console

1. Launch the AWS Management Console

When you [click here](#), the AWS Management Console will open in a new browser window, so you can keep this step-by-step guide open. When the screen loads, enter your user name

and password to get started. Then type **notification** in the search bar and select **Simple Notification Service** to open the service console.



2. Start the Amazon SNS Console

If the SNS console landing page appears, click **Next step**.

aws Services Resource Groups

Application Integration

Amazon Simple Notification Service

Pub/sub messaging for microservices and serverless applications.

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

Create topic

Topic name
A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

Next step

[Start with an overview](#)

Benefits and features

Reliably deliver messages with durability

Amazon SNS uses cross availability zone message storage to provide high message durability. Amazon SNS reliably delivers messages to valid AWS endpoints, such as Amazon SQS queues and AWS Lambda functions.

Automatically scale your workload

Amazon SNS leverages the proven AWS cloud to dynamically scale with your application. Amazon SNS is a fully managed service, taking care of the heavy lifting related to capacity planning, provisioning, monitoring, and patching.

Simplify your architecture with Message Filtering

Amazon SNS helps you simplify your pub/sub messaging architecture by offloading the message filtering logic from your subscriber systems, and message routing logic from your publisher systems.

Keep messages private and secure

Amazon SNS topic owners can set topic policies that restrict who can publish and subscribe to a topic. Amazon SNS also ensures that data is encrypted in transit and at rest, and provides VPC endpoints for message privacy.

Pricing


Amazon SNS has no upfront costs. You pay based on the number of messages you publish, the number of messages you deliver, and any additional API calls for managing topics and subscriptions. Delivery pricing varies by endpoint type.

[Learn more](#)


Documentation

- [Developer Guide](#)
- [API Reference](#)
- [FAQs](#)
- [Support forums](#)

Use cases



The NASA image & video library provides easy access to thousands of images, audio recordings and videos, documenting NASA's more than half a century of achievements in exploring the vast unknown. [Learn more](#)



PlayOn! operates a comprehensive technology platform and serverless video processing pipeline, enabling high-quality, low-cost productions of live sporting events. [Learn more](#)

Explore AWS

Amazon RDS

Set up, operate, and scale your relational database in the cloud. [Learn more](#)

Run Serverless Containers with AWS Fargate

AWS Fargate runs and scales your containers without having to manage servers or clusters. [Learn more](#)

Related services

Amazon SQS

Fully managed message queues for microservices, distributed systems, and serverless applications.

Amazon MQ

Managed message broker service for Apache ActiveMQ

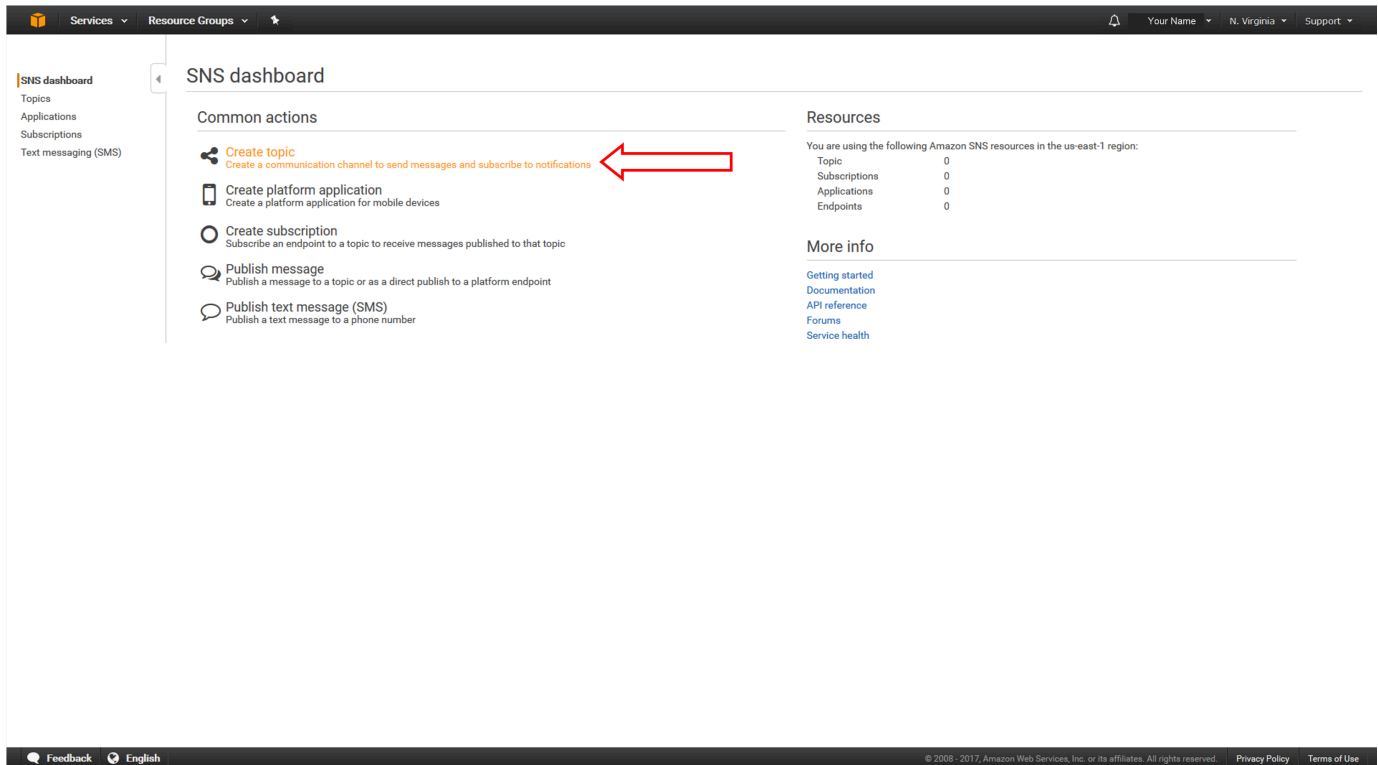
Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Step 2: Create an Amazon SNS Topic

In this step, you will create an Amazon SNS topic. A topic is a communication channel to send messages and subscribe to notifications. In this example, a sample ecommerce application will push a message to an Amazon SNS topic whenever a new order is placed on the online store.

1. Create an SNS topic

In the Create topic page, type **New-Orders**, in the topic name box, then click **Create topic**.



The screenshot shows the Amazon SNS dashboard. The left sidebar contains navigation links: SNS dashboard, Topics, Applications, Subscriptions, and Text messaging (SMS). The main content area is titled 'SNS dashboard' and features a 'Common actions' section. The first action, 'Create topic', is highlighted with a red arrow and includes the subtext 'Create a communication channel to send messages and subscribe to notifications'. Other actions include 'Create platform application', 'Create subscription', 'Publish message', and 'Publish text message (SMS)'. The right sidebar shows 'Resources' (Topic: 0, Subscriptions: 0, Applications: 0, Endpoints: 0) and 'More info' links (Getting started, Documentation, API reference, Forums, Service health). The footer includes 'Feedback', 'English', and copyright information.

2. Verify topic creation

The Topic details page confirms the topic is successfully created.

The screenshot shows the AWS Management Console interface for Amazon SNS. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information. A green notification banner at the top states: 'Topic New-Orders created successfully. You can create subscriptions and send messages to them from this topic.' Below this, the console shows the 'New-Orders' topic details. The 'Details' section includes fields for Name (New-Orders), ARN (arn:aws:sns:us-west-2:123456789000:New-Orders), Display name (-), and Topic owner (123456789000). Below the details are tabs for 'Subscriptions', 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. The 'Subscriptions' tab is active, showing a table with columns for ID, Endpoint, Status, and Protocol. The table is currently empty, with a message stating 'No subscriptions found. You don't have any subscriptions to this topic.' and a 'Create subscription' button. The console footer includes 'Feedback', 'English (US)', and copyright information.

Step 3: Create the Amazon SQS Queues

Now that you have created the topic with Amazon SNS, you will create Amazon SQS queues that will subscribe to the topic.

When you subscribe multiple queues to a topic, each queue receives identical notifications every time a message is pushed to the topic. Services attached to those queues can then process the orders asynchronously and in parallel.

For example, an Amazon EC2 server instance attached to one of the queues could handle the processing or fulfillment of the order, while the other server instance could be attached to a data warehouse for analysis of all orders received.

To keep things simple, we won't actually attach EC2 instances to the queues in this tutorial.

1. Open the Amazon SQS console

[Click here](#) to open the Amazon SQS console in a new browser window. If the SQS landing page appears, click **Get Started Now**. Otherwise, proceed to the next step.

The screenshot shows the Amazon SQS console landing page. At the top, there is a navigation bar with 'Services' and 'Resource Groups' dropdowns, and a user profile section with 'Your Name', 'Ohio', and 'Support'. The main content area features the AWS logo, the title 'Simple Queue Service', and a description: 'Amazon Simple Queue Service (SQS) is a reliable, scalable, fully-managed message queuing service.' A red arrow points to a blue 'Get Started Now' button. Below this are three sections: 'Ensure high availability', 'Scale with your business', and 'Reduce your cost', each with an icon and a brief description. At the bottom, there is a link for 'AWS SQS Documentation and Support'.

2. Create Orders-for-Inventory queue

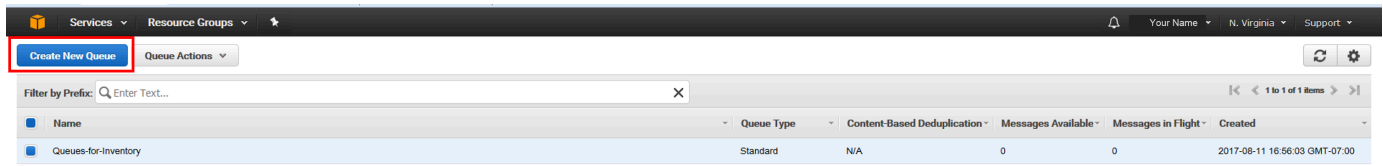
Our first queue will store orders for a fictional Inventory Service that keeps track of products, adding and deleting them as needed from inventory with each order.

On the **Create New Queue** page, enter **Orders-for-Inventory** in the **Queue Name** field. Leave **Standard Queue** selected and click **Quick-Create Queue**.

3. Create Orders-for-Analytics queue

Your new queue is created and selected in the queue list. Next, you'll create a second queue to handle order analytics.

Click **Create New Queue** to create another queue to store orders for the Analytics Service.



Services Resource Groups

Create New Queue Queue Actions

Filter by Prefix: Enter Text...

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
Queues-for-Inventory	Standard	N/A	0	0	2017-08-11 16:56:03 GMT-07:00

1 SQS Queue selected

Details Permissions Redrive Policy Monitoring Encryption

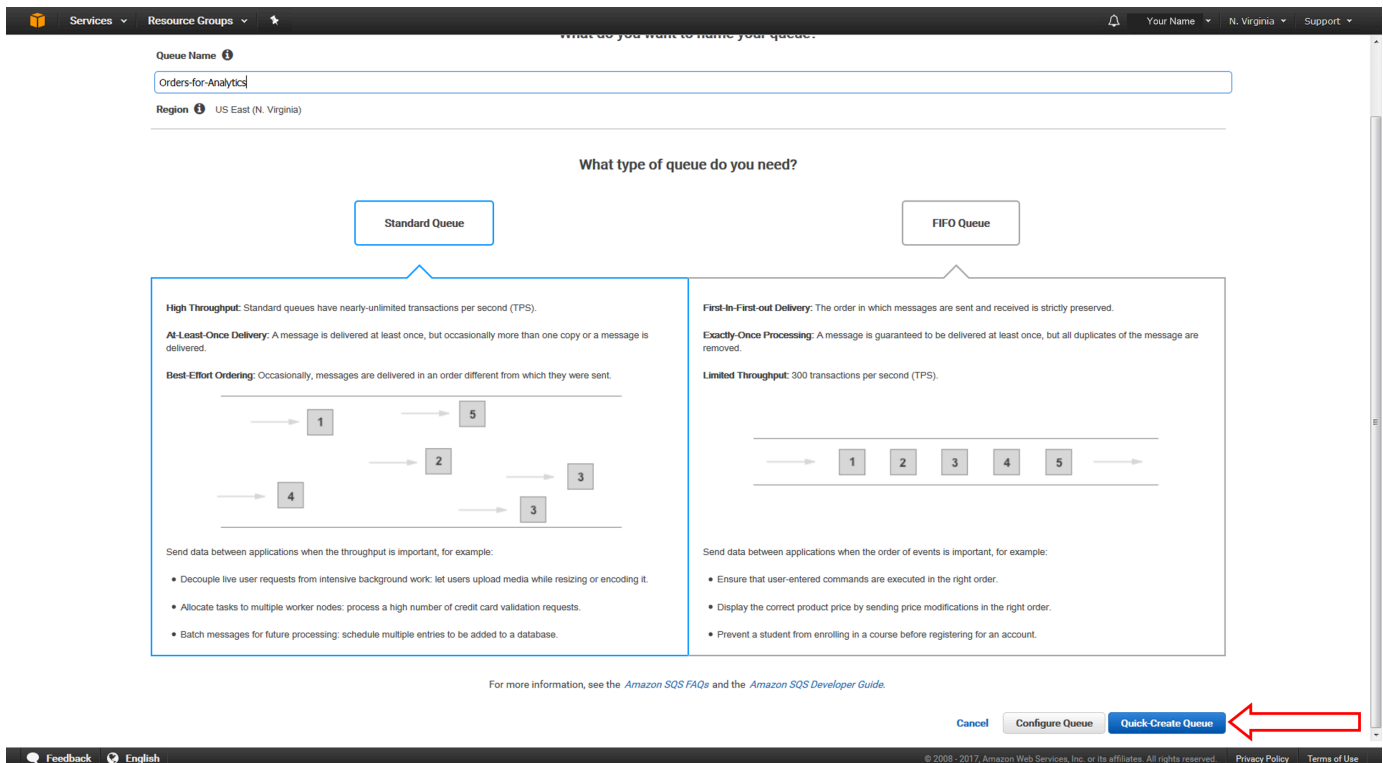
Name: Queues-for-Inventory
 URL: https://sqs.us-east-1.amazonaws.com/585534329928/Queues-for-Inventory
 ARN: arn:aws:sqs:us-east-1:585534329928:Queues-for-Inventory
 Created: 2017-08-11 16:56:03 GMT-07:00
 Last Updated: 2017-08-11 16:56:03 GMT-07:00
 Delivery Delay: 0 seconds
 Queue Type: Standard
 Content-Based Deduplication: N/A

Default Visibility Timeout: 30 seconds
 Message Retention Period: 4 days
 Maximum Message Size: 256 KB
 Receive Message Wait Time: 0 seconds
 Messages Available (Visible): 0
 Messages in Flight (Not Visible): 0
 Messages Delayed: 0

Feedback English © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

4. Enter queue name

Enter **Orders-for-Analytics** in the **Queue Name** field, and click **Quick-Create Queue**.



Queue Name

Region

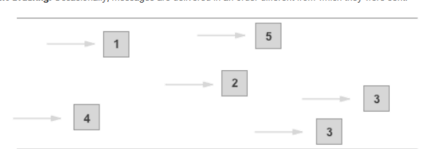
What type of queue do you need?

Standard Queue

High Throughput: Standard queues have nearly-unlimited transactions per second (TPS).

At Least-Once Delivery: A message is delivered at least once, but occasionally more than one copy or a message is delivered.

Best-Effort Ordering: Occasionally, messages are delivered in an order different from which they were sent.



Send data between applications when the throughput is important, for example:


- Decouple live user requests from intensive background work: let users upload media while resizing or encoding it.
- Allocate tasks to multiple worker nodes: process a high number of credit card validation requests.
- Batch messages for future processing: schedule multiple entries to be added to a database.

FIFO Queue

First-in-First-out Delivery: The order in which messages are sent and received is strictly preserved.

Exactly-Once Processing: A message is guaranteed to be delivered at least once, but all duplicates of the message are removed.

Limited Throughput: 300 transactions per second (TPS).



Send data between applications when the order of events is important, for example:

- Ensure that user-entered commands are executed in the right order.
- Display the correct product price by sending price modifications in the right order.
- Prevent a student from enrolling in a course before registering for an account.

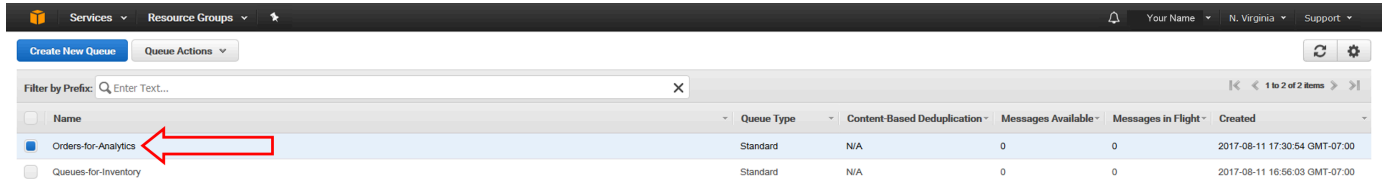
For more information, see the [Amazon SQS FAQs](#) and the [Amazon SQS Developer Guide](#).

Cancel Configure Queue **Quick-Create Queue**

Feedback English © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

5. Verify queue creation

The new queue now appears in the queue list.



The screenshot shows the AWS Management Console interface for the SQS console. At the top, there are navigation tabs for 'Services' and 'Resource Groups'. Below that, there are buttons for 'Create New Queue' and 'Queue Actions'. A search bar labeled 'Filter by Prefix' is present. The main content area displays a table of SQS queues. The table has columns for 'Name', 'Queue Type', 'Content-Based Deduplication', 'Messages Available', 'Messages in Flight', and 'Created'. Two queues are listed: 'Orders-for-Analytics' and 'Queues-for-Inventory'. The 'Orders-for-Analytics' queue is selected, and a red arrow points to its name.

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
Orders-for-Analytics	Standard	N/A	0	0	2017-08-11 17:30:54 GMT-07:00
Queues-for-Inventory	Standard	N/A	0	0	2017-08-11 16:56:03 GMT-07:00



The screenshot shows the details page for the 'Orders-for-Analytics' SQS queue. The page has tabs for 'Details', 'Permissions', 'Redrive Policy', 'Monitoring', and 'Encryption'. The 'Details' tab is active. The details are organized into two columns. The left column contains: Name: Orders-for-Analytics, URL: https://sqs.us-east-1.amazonaws.com/565534329928/Orders-for-Analytics, ARN: arn:aws:sqs:us-east-1:565534329928:Orders-for-Analytics, Created: 2017-08-11 17:30:54 GMT-07:00, Last Updated: 2017-08-11 17:30:54 GMT-07:00, Delivery Delay: 0 seconds, Queue Type: Standard, Content-Based Deduplication: N/A. The right column contains: Default Visibility Timeout: 30 seconds, Message Retention Period: 4 days, Maximum Message Size: 256 KB, Receive Message Wait Time: 0 seconds, Messages Available (Visible): 0, Messages in Flight (Not Visible): 0, Messages Delayed: 0.

Property	Value
Name	Orders-for-Analytics
URL	https://sqs.us-east-1.amazonaws.com/565534329928/Orders-for-Analytics
ARN	arn:aws:sqs:us-east-1:565534329928:Orders-for-Analytics
Created	2017-08-11 17:30:54 GMT-07:00
Last Updated	2017-08-11 17:30:54 GMT-07:00
Delivery Delay	0 seconds
Queue Type	Standard
Content-Based Deduplication	N/A
Default Visibility Timeout	30 seconds
Message Retention Period	4 days
Maximum Message Size	256 KB
Receive Message Wait Time	0 seconds
Messages Available (Visible)	0
Messages in Flight (Not Visible)	0
Messages Delayed	0

Step 4: Subscribe the Queues to the Topic

Now that you have created your two Amazon SQS queues, you need to subscribe them to the Amazon SNS topic that broadcasts notifications of new orders.

1. Subscribe the queues

From the list of queues, select the **Orders-for-Inventory** and **Orders-for-Analytics** queues. From **Queue Actions**, select **Subscribe Queues to SNS Topic**.

The screenshot shows the AWS Management Console interface for SQS queues. A dropdown menu is open over a table of queues, with the 'Subscribe Queues to SNS Topic' option highlighted by a red arrow. The table below shows two queues with columns for Name, Queue Type, Content-Based Deduplication, Messages Available, Messages in Flight, and Created.

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
Orders-for-Analytic	Standard	N/A	0	0	2017-08-11 17:30:54 GMT-07:00
Queues-for-Invento	Standard	N/A	0	0	2017-08-11 16:56:03 GMT-07:00

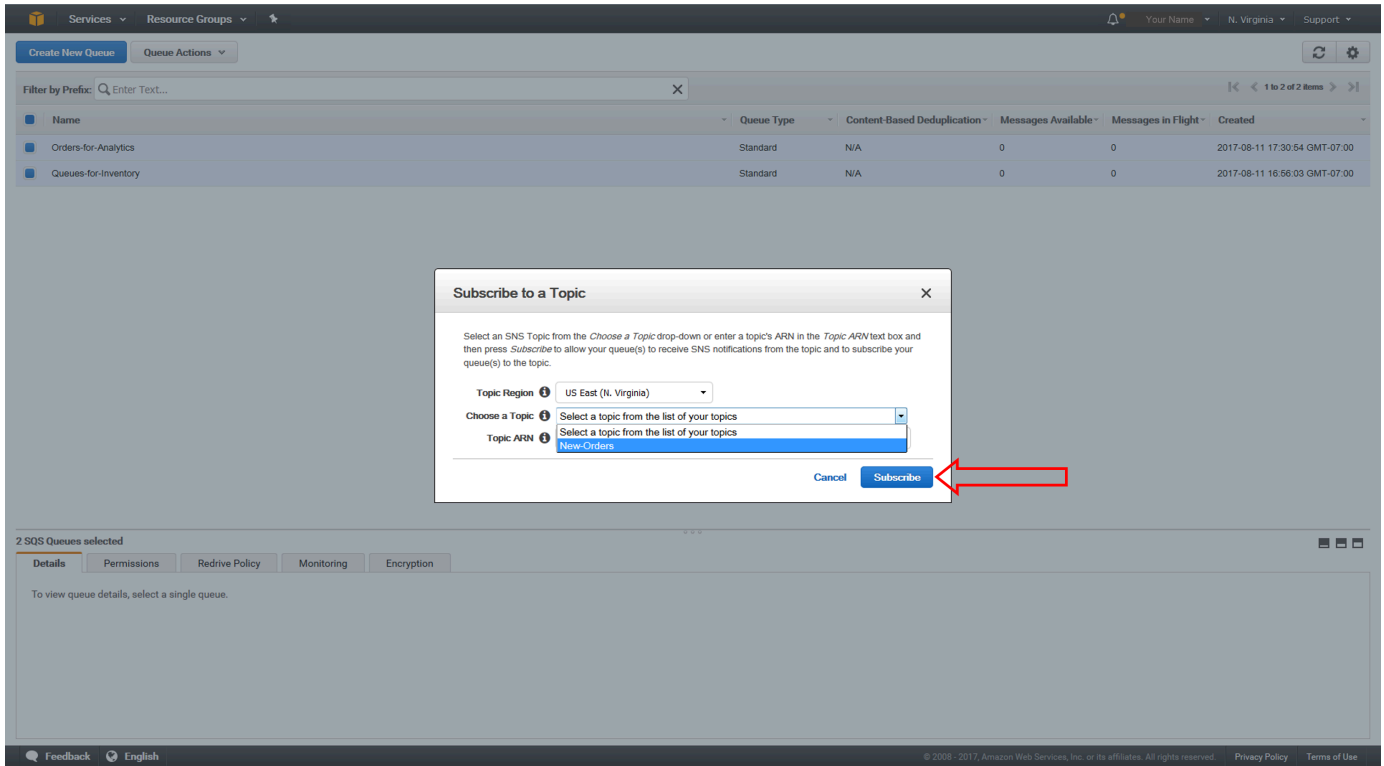
Below the table, a dialog box titled '2 SQS Queues selected' is shown with tabs for 'Details', 'Permissions', 'Redrive Policy', 'Monitoring', and 'Encryption'. The 'Details' tab is active, and the text 'To view queue details, select a single queue.' is displayed.

2. Select topic

The **Subscribe to a Topic** dialog box is displayed. From the **Choose a Topic** drop-down list, select your **New-Orders** Amazon SNS topic.

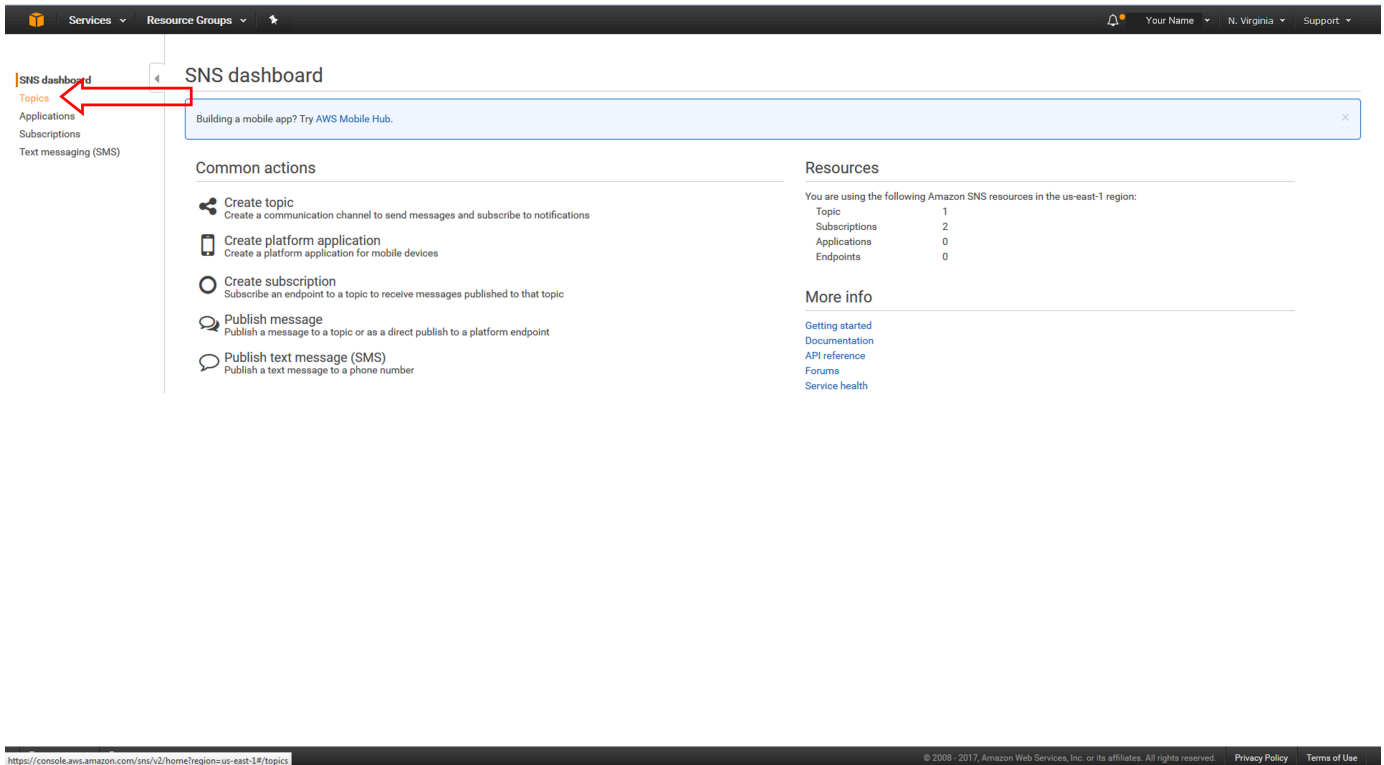
Your SNS topic appears in the list because you created it from the same account that you used to create your Amazon SQS queues. If the SNS topic was made by another account, you could subscribe to it by using the Topic ARN. For more details, see the [Amazon SNS documentation](#).

Leave the **Topic Region** unchanged, and click **Subscribe**.



3. Confirm subscription

The **Topic Subscription Result** dialog box is displayed. Click **OK**.

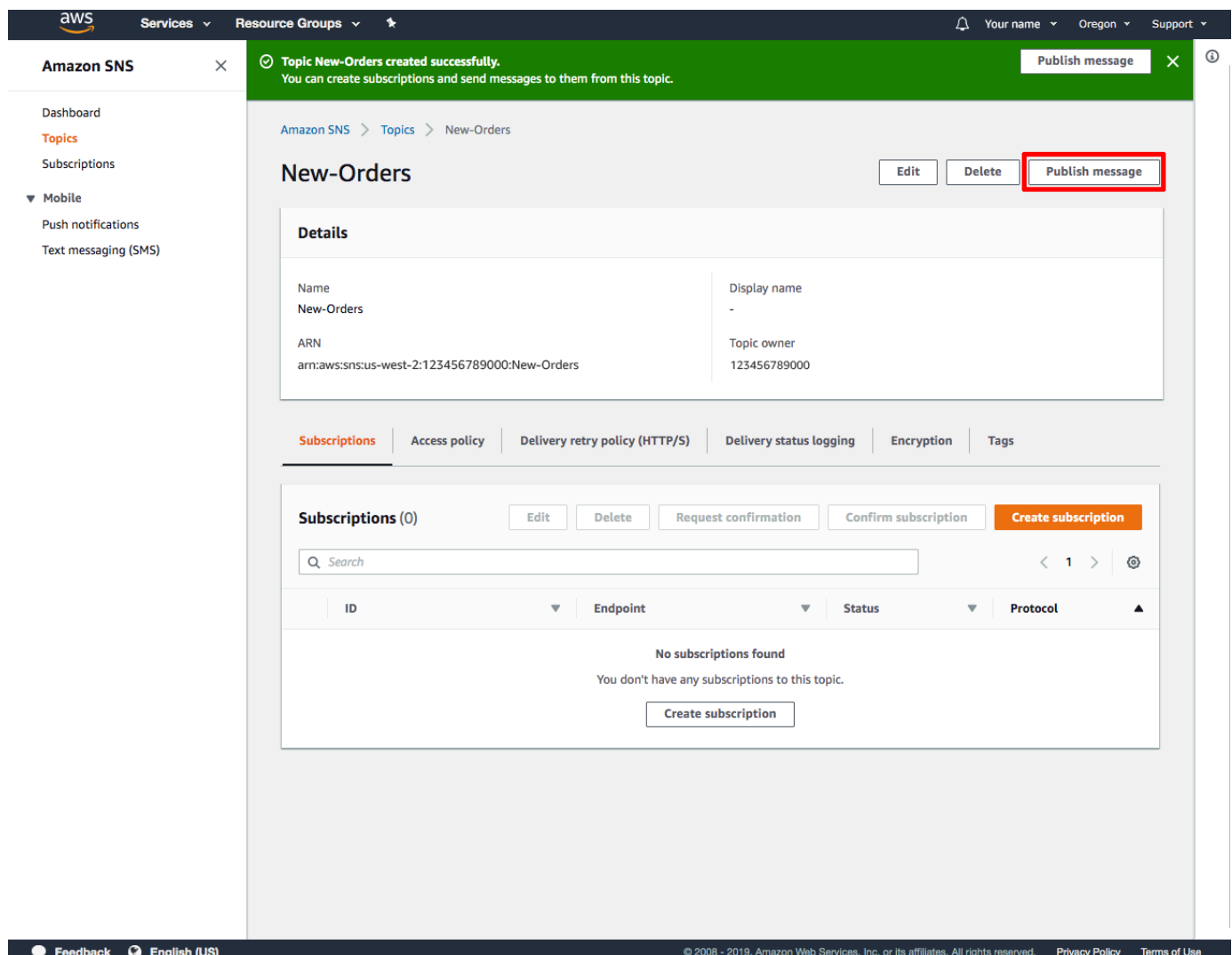


Step 5: Publish a Message to the Topic

Your queues are now subscribed to the topic. In this step, you will simulate a new order by having the fictional ecommerce application push a message to the topic with the order details.

1. Open the publish page

In the Amazon SNS console **New Orders** topic details page, click **Publish message**.



The screenshot shows the Amazon SNS console interface. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information. A green notification banner at the top states 'Topic New-Orders created successfully. You can create subscriptions and send messages to them from this topic.' Below this, the 'New-Orders' topic details are displayed. The 'Publish message' button is highlighted with a red box. The 'Details' section shows the topic name 'New-Orders', ARN 'arn:aws:sns:us-west-2:123456789000:New-Orders', and topic owner '123456789000'. Below the details, there are tabs for 'Subscriptions', 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. The 'Subscriptions (0)' section shows a search bar and a table with columns for ID, Endpoint, Status, and Protocol. A message indicates 'No subscriptions found' and provides a 'Create subscription' button.

2. Send the order message

Publish Message to topic page appears. In the Subject box, type Order 123-4567890-1234567. In the **Message** field, enter the following text to represent a sample order:

1 x Widget @ \$29.99 USD 2 x Widget Cables @ \$4.99

Click **Publish Message**. A confirmation dialog box will appear.

The screenshot shows the AWS console interface for publishing a message to an SNS topic. The breadcrumb trail is Amazon SNS > Topics > New-Orders > Publish message. The main heading is "Publish message to topic".

Message details

Topic ARN
arn:aws:sns:us-west-2:123456789000:New-Orders

Subject - *optional*
123-4567890-1234567
Maximum 100 printable ASCII characters

Time to Live (TTL) - *optional*
This setting applies only to mobile application endpoints. The number of seconds that the push notification service has to deliver the message to the endpoint. [Info](#)

Message body

Message structure

- Identical payload for all delivery protocols.
The same payload is sent to endpoints subscribed to the topic, regardless of their delivery protocol.
- Custom payload for each delivery protocol.
Different payloads are sent to endpoints subscribed to the topic, based on their delivery protocol.

Message body to send to the endpoint

```
1 1 x Widget @ $29.99 USD
2 2 x Widget Cables @ $4.99
```

Message attributes

Message attributes let you provide structured metadata items (such as timestamps, geospatial data, signatures, and identifiers) for the message. [Info](#)

Type	Name	Value	
Select attribute type	Enter attribute name	value or ["value1", "value2"]	Remove

[Add another attribute](#)

Cancel **Publish message**

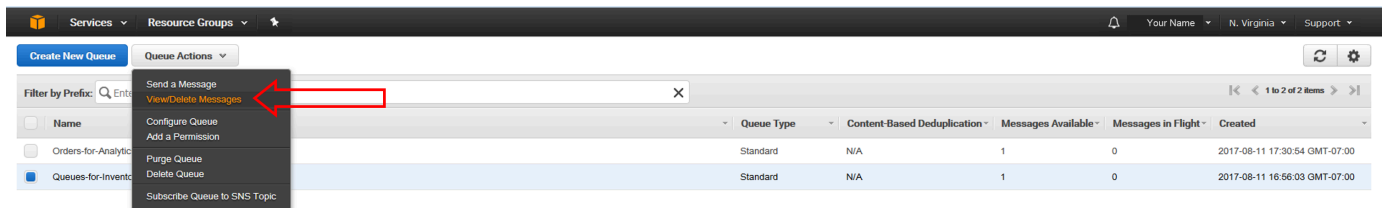
Step 6: Verify the Subscription

Once a new message is published, Amazon SNS will deliver that message to every endpoint that is subscribed to the topic. In a fanout scenario like this one, the Amazon SQS queues are the endpoints.

In this step, you will confirm that the queues received the new order notification by viewing the message that the topic sent to the queues.

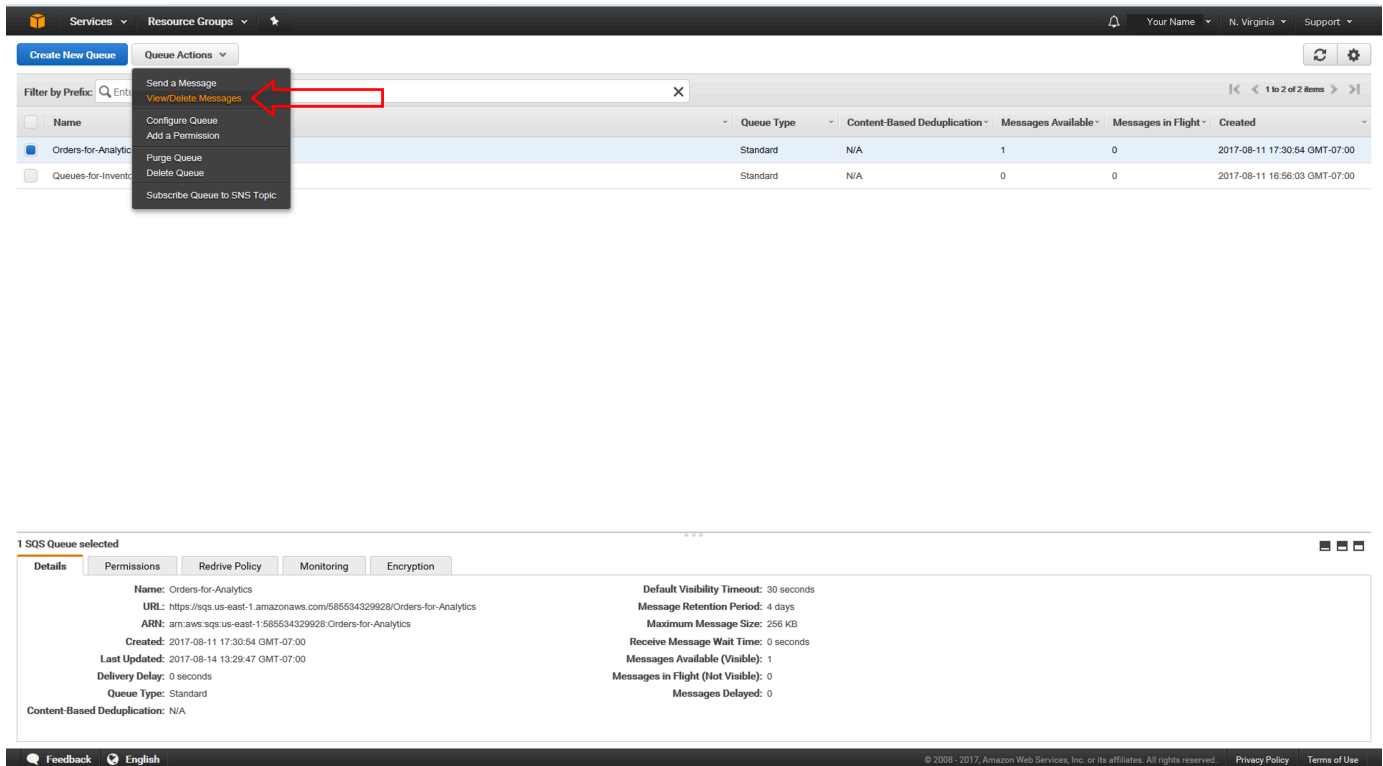
1. Open queue messages

In the Amazon SQS console, check the box for the **Orders-for-Inventory** queue from the queue list. From the **Queue Action** drop-down, select **View/Delete Messages**.



2. Start message polling

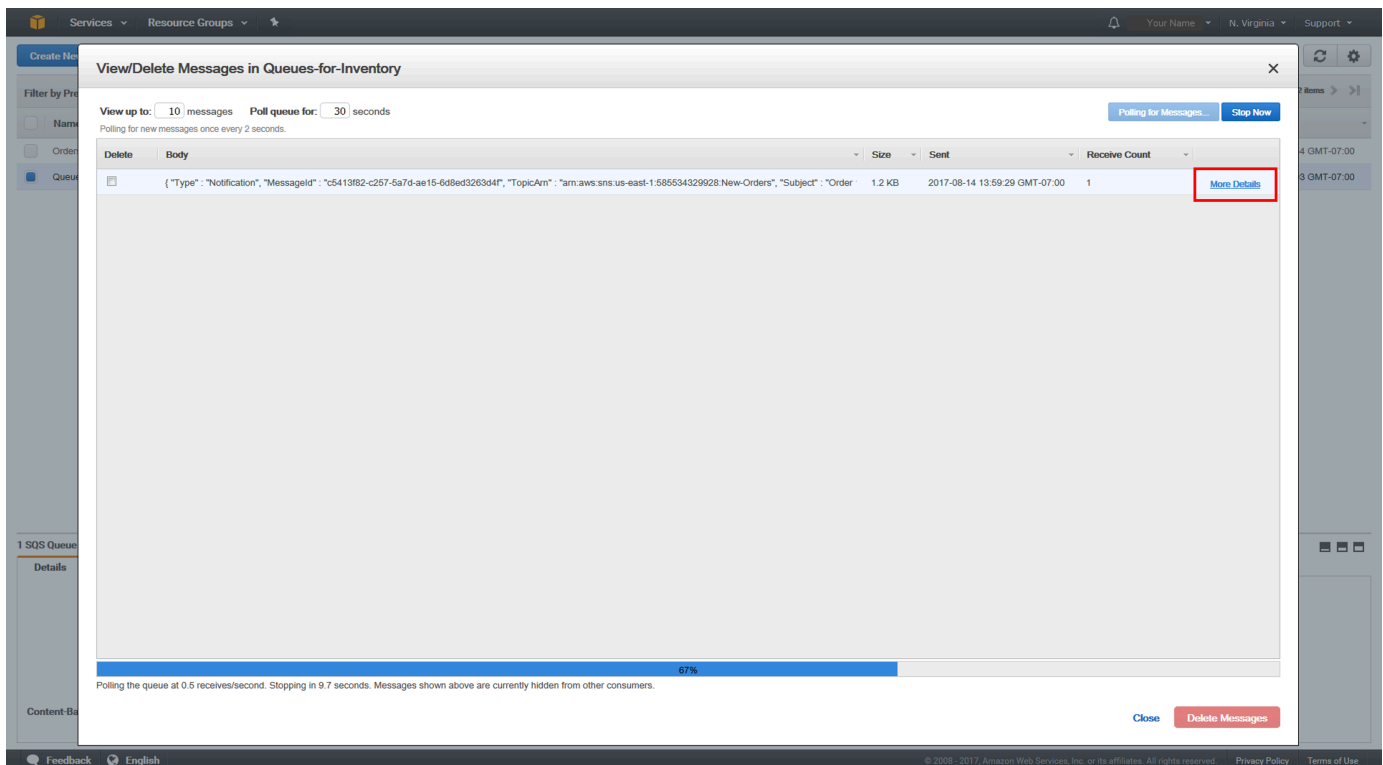
Click **Start Polling for Messages**.



The screenshot shows the AWS IAM console interface. At the top, there are navigation tabs for 'Services', 'Resource Groups', and 'Support'. Below this, there's a 'Queue Actions' dropdown menu. The menu items are: 'Send a Message', 'View/Delete Messages' (highlighted with a red arrow), 'Configure Queue', 'Add a Permission', 'Purge Queue', 'Delete Queue', and 'Subscribe Queue to SNS Topic'. In the background, a table lists SQS queues with columns for Name, Queue Type, Content-Based Deduplication, Messages Available, Messages in Flight, and Created. Two queues are visible: 'Orders-for-Analytics' and 'Queues-for-Inventory'.

3. View message panel

The **View/Delete Messages in Orders-for-Inventory** dialog box appears.



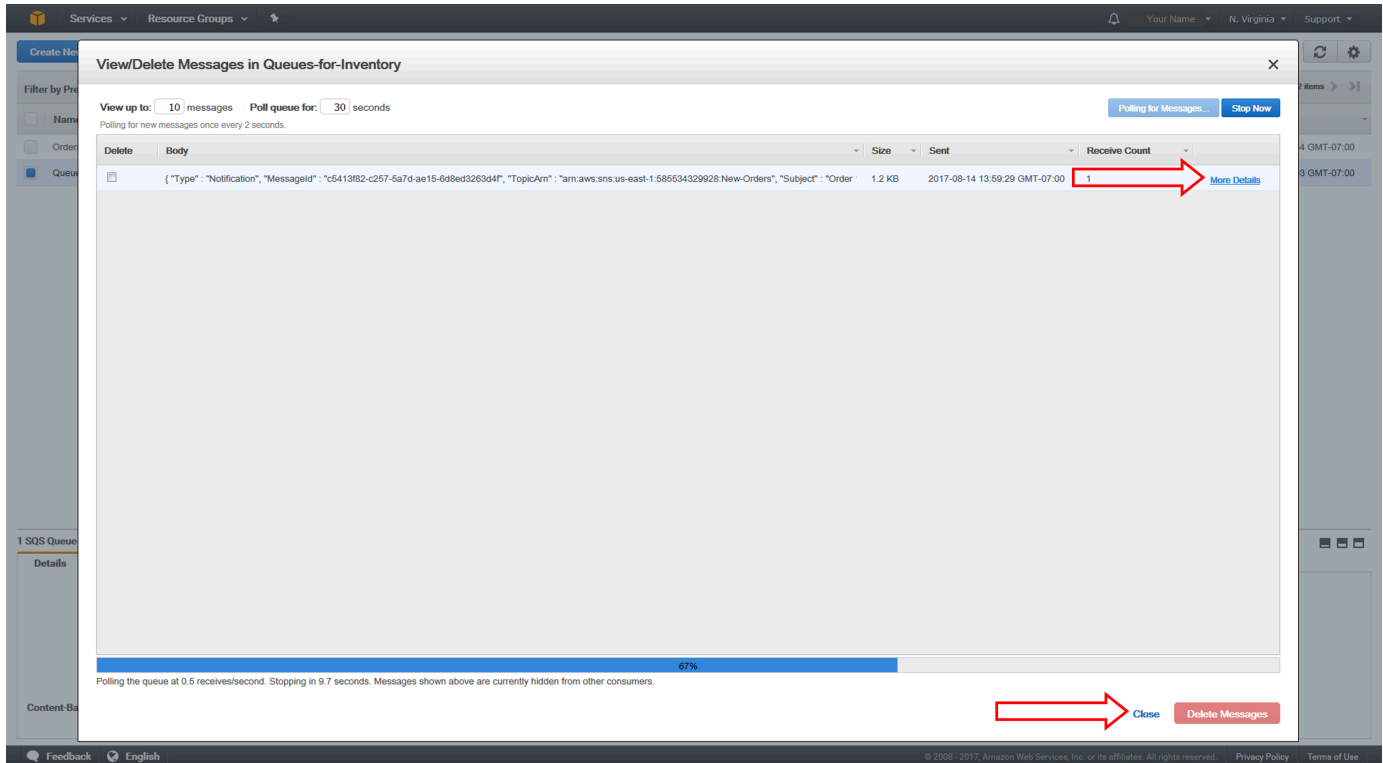
The screenshot shows the 'View/Delete Messages in Queues-for-Inventory' dialog box. At the top, it says 'View up to: 10 messages' and 'Poll queue for: 30 seconds'. Below this, there's a table with columns: 'Delete', 'Body', 'Size', 'Sent', and 'Receive Count'. One message is listed with a body containing JSON: `{ "Type": "Notification", "MessageId": "c541382c-257-5a7d-ae15-6d8ed3263d4f", "TopicArn": "arn:aws:sns:us-east-1:586534329928:New-Orders", "Subject": "Order" }`. The 'More Details' link is highlighted with a red box. At the bottom, there's a progress bar showing '67%' and a 'Delete Messages' button.

4. Check message content

In the **Body** column, click **More Details**. The **Message Details** box contains a JSON document that contains the subject and message that you published to the topic.

You have confirmed that the **Orders-for-Inventory** queue received the notification of the new order from the **New-Orders** topic.

Click **Close**.



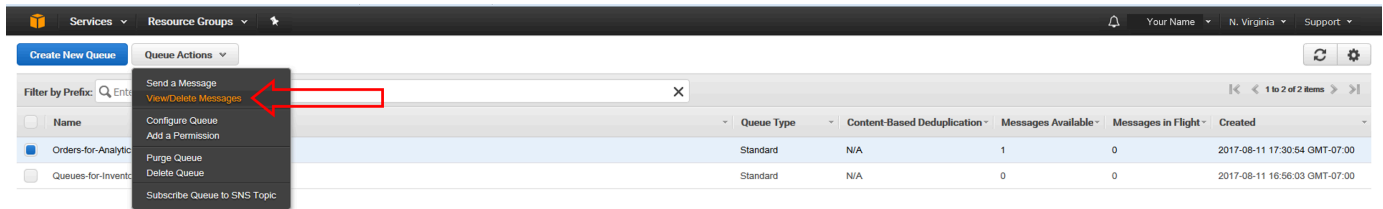
5. Delete message

We'll assume that our fictional Inventory Service has finished processing this message, and that we can now safely delete the message from the queue.

Click **Delete 1 Message**. To confirm, click **Yes, Delete Checked Messages**. Then click **Close**.


6. Verify Orders-for-Analytics queue received notification

Repeat steps 1 through 5 to confirm that the **Orders-for-Analytics** queue also received the notification of the new order.



The screenshot shows the Amazon SNS console interface. At the top, there are navigation tabs for 'Services' and 'Resource Groups'. Below this is a 'Queue Actions' dropdown menu with options: 'Send a Message', 'View/Delete Messages' (highlighted with a red arrow), 'Configure Queue', 'Add a Permission', 'Purge Queue', 'Delete Queue', and 'Subscribe Queue to SNS Topic'. Below the menu is a table of queues:

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
Orders-for-Analytic	Standard	N/A	1	0	2017-08-11 17:30:54 GMT-07:00
Queues-for-Invent	Standard	N/A	0	0	2017-08-11 16:56:03 GMT-07:00



The screenshot shows the details of a selected SQS queue. The queue name is 'Orders-for-Analytics'. The details are organized into two columns:

Details	Permissions	Redrive Policy	Monitoring	Encryption
Name: Orders-for-Analytics URL: https://sqs.us-east-1.amazonaws.com/585534329928/Orders-for-Analytics ARN: arn:aws:sqs:us-east-1:585534329928:Orders-for-Analytics Created: 2017-08-11 17:30:54 GMT-07:00 Last Updated: 2017-08-14 13:29:47 GMT-07:00 Delivery Delay: 0 seconds Queue Type: Standard Content-Based Deduplication: N/A	Default Visibility Timeout: 30 seconds Message Retention Period: 4 days Maximum Message Size: 256 KB Receive Message Wait Time: 0 seconds Messages Available (Visible): 1 Messages in Flight (Not Visible): 0 Messages Delayed: 0			

Clean Up Resources

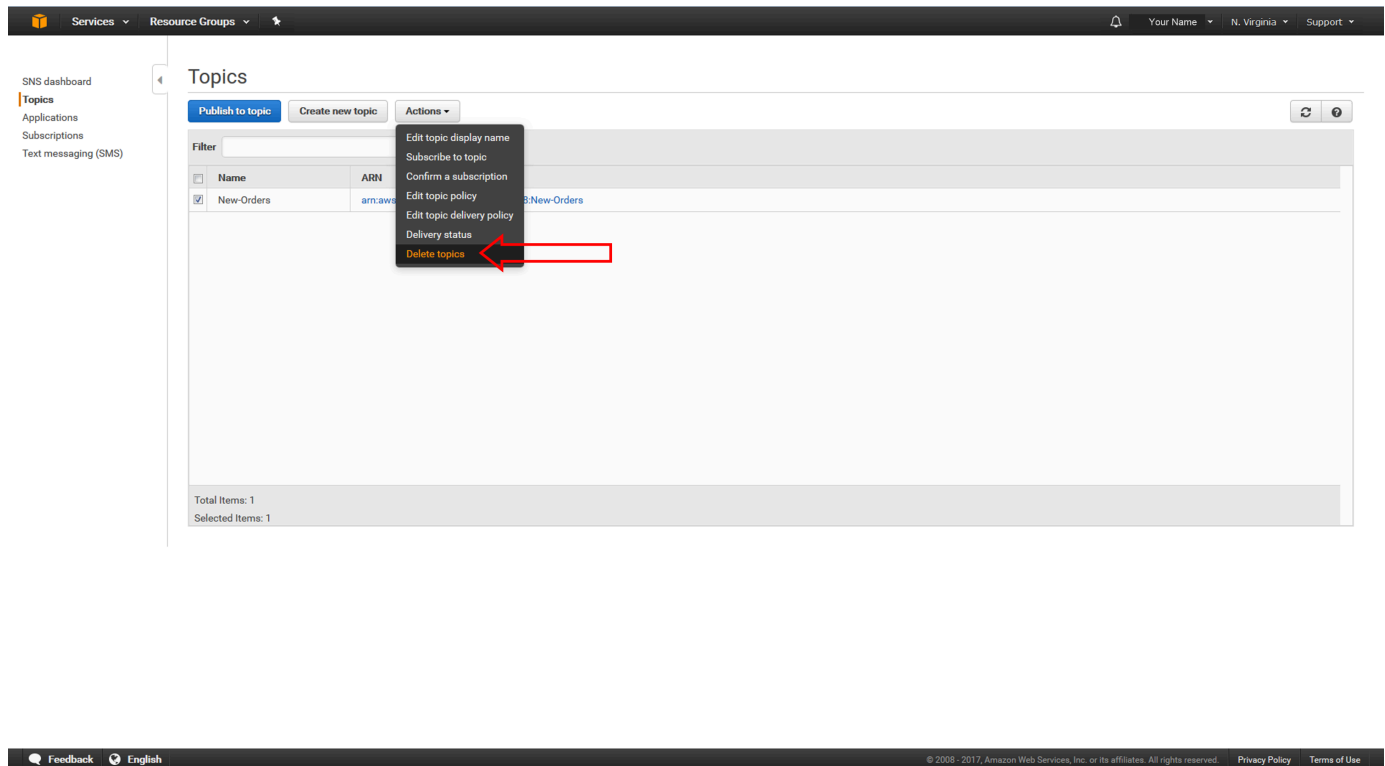
In this step, you will delete the resources you have created for this tutorial, which include the Topic Subscriptions, Topics, and Queues. It is a best practice to delete resources you are no longer using so you don't incur charges.

1. Delete the topics

Open the Amazon SNS console and click **Topics** in the left navigation pane.

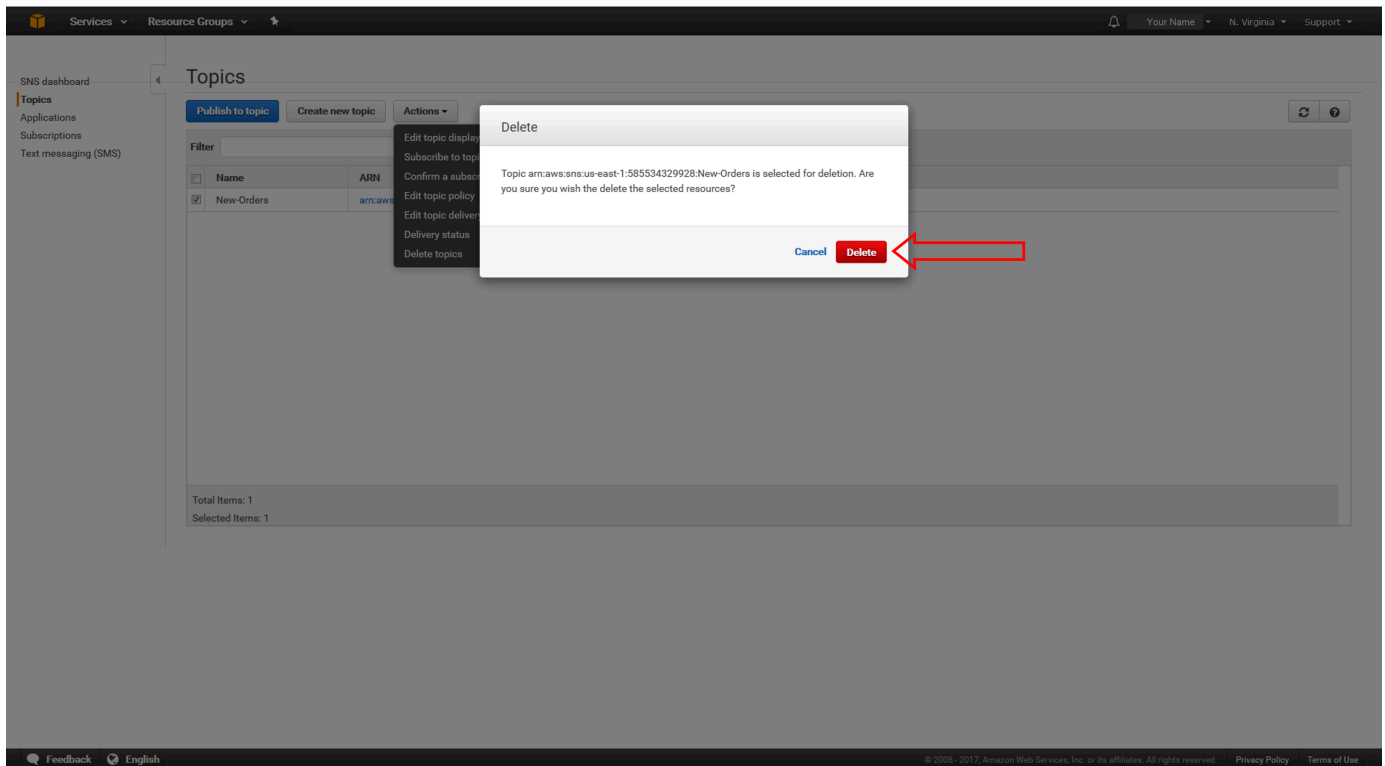
Select the **New-Orders** topic.

Click **Delete** to delete topics.



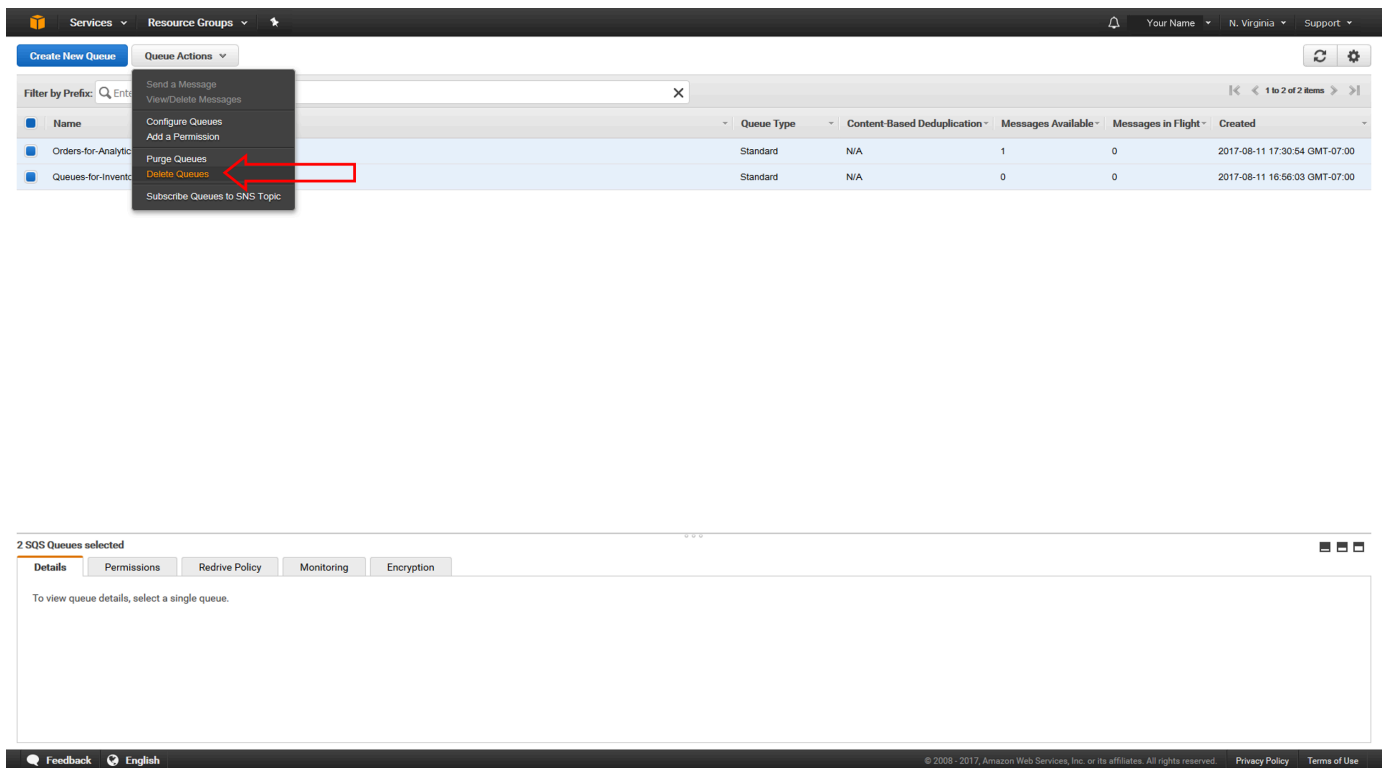
2. Confirm deletion

The **Delete** confirmation dialog box appears. Type **delete me** in the dialog box and click **Delete**. The topic, and its subscriptions, are deleted. You can now close the SNS browser window (but don't sign out, as you still need to delete the queues in the SQS console).



3. Delete the queues

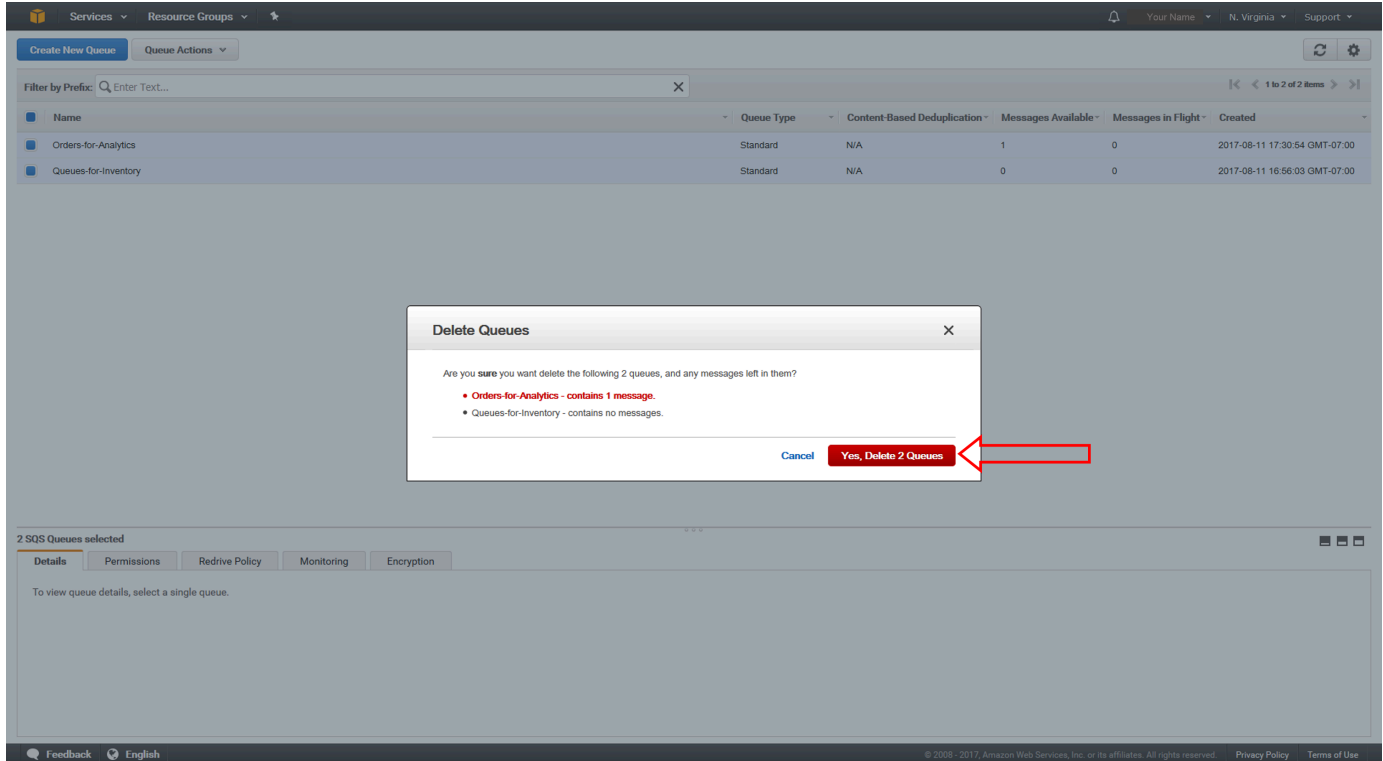
In the Amazon SQS console, select the **Orders-for-Inventory** and **Orders-for-Analytics** queues. From **Queue Actions**, select **Delete Queues**.



4. Confirm deletion

The **Delete Queues** dialog box is displayed. Click **Yes, Delete 2 Queues**. The queues are deleted.

You can now sign out of the Amazon SQS console.



Congratulations!

You have implemented a fanout scenario using Amazon SNS and Amazon SQS. You are now ready to use Amazon SNS and Amazon SQS together to deliver messages to applications that require immediate notification of an event, or to buffer messages in Amazon SQS queues for other applications to process later.