

**AWS Group and Policies Setup**

August 31, 2016

**Document Control Information**

**Document Edit History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | Date | Description | Author |
| 1.0 | 08/30/2016 | Initial Version | Ankit Prakash |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Support / SME Resources**

|  |  |  |  |
| --- | --- | --- | --- |
| **Team** | Name | Role | Contact (Email / Phone) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1. Overview 4](#_Toc460338948)

[2. Purpose 4](#_Toc460338949)

[3. Groups and Policies 4](#_Toc460338950)

[a. DevFullAccess 4](#_Toc460338951)

[b. TestReadAccess 5](#_Toc460338952)

[c. TestReadWriteAccess 5](#_Toc460338953)

[d. TestAdminAccess 5](#_Toc460338954)

[e. ProdReadAccess 5](#_Toc460338955)

[f. ProdAdminAccess 5](#_Toc460338956)

[4. Amazon S3 Bucket Creation Process 6](#_Toc460338957)

[5. Amazon SNS Topic Creation Process 8](#_Toc460338958)

[6. Amazon SQS Queue Creation Process 11](#_Toc460338959)

1. Overview

This document provides the AWS Group and Policies Setup required for S3/SNS/SQS creation

which are needed by several project. The is coherent with TCCC practice.

1. Purpose

This document talks about various groups and policies we are going to setup to address access to

the Amazon Console. Right now there is a Power Developer group which allows to do everything.

Other groups like DevSQSFullAccess, DevSNSFullAccess and DevS3FullAccess are there. But they

are not working the way they are intended for. Their policies are primitive and if we add users to

these groups users are not able to click on SQS, SNS and S3, as they are getting permission denied

error. We need to expand these policies to address the issue.

1. Groups and Policies

There are several groups and their policies.

1. DevFullAccess

As Dev environment meant for full access to the developers. This group policy looks like as

follows. We are putting the proposed policy here to give general idea on how we are

progressing to do this.

{

"Version": "2012-10-17",

"Statement":

[ { "Effect": "Allow",

"Action": [ "sqs:ListQueues" ],

"Resource": "\*"

},

{

"Action": [ "sqs:\*" ],

"Effect": "Allow",

"Resource": "arn:aws:sqs:\*:390037866200:\*-DEV"

},

{

"Effect": "Allow",

"Action": [ "s3:\*" ],

"Resource": [ "arn:aws:s3:::\*-Dev" ]

},

{

"Action": [ "sns:ListTopics" ],

"Effect": "Allow",

"Resource": "arn:aws:sns:\*:390037866200:\*"

},

{

"Action": [ "sns:\*" ],

"Effect": "Allow",

"Resource": "arn:aws:sns:\*:390037866200:\*-DEV"

},

{

"Effect": "Allow",

"Action": [ "iam:ChangePassword", "iam:GetAccountPasswordPolicy" ],

"Resource": "\*"

}

]

}

DevFullAccess group will allow everything on Dev queues, Dev SNS and Dev S3 buckets. We are

going to add all the developers to this group. They should be creating new SQS queues, S3

Buckets and SNS Topics.

1. TestReadAccess

**I**n this group we will set up the policy so that developers can read messages in the queues in

all test queues. They can’t write to the queues. They can’t create new queues as well.

1. TestReadWriteAccess

In this group we will set up the policy so that developers can write messages to the queues. But

they can’t create new queues. For new queues developers need to approach the user who

belongs to **PowerDeveloper** group.

1. TestAdminAccess

This group is like DevFullAccess group for test environment. Users belong to this group will be

able to do everything in test environment.

1. ProdReadAccess

In this group we will set up the policy so that developer can read messages in the queues in all

prod queues. They can’t write to the queues. They can’t create new queues as well.

1. ProdAdminAccess

This group is like DevFullAccess for prod environment. Users belong to this group will be able

to do everything in prod environment.

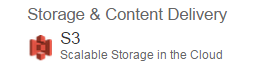
1. Amazon S3 Bucket Creation Process

1. Login to the AWS with your credentials (Your credentials should have necessary access so that you create, read and delete the buckets when needed).

url :- <https://esb-integration-resources.signin.aws.amazon.com/console>

[https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2#](https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2)

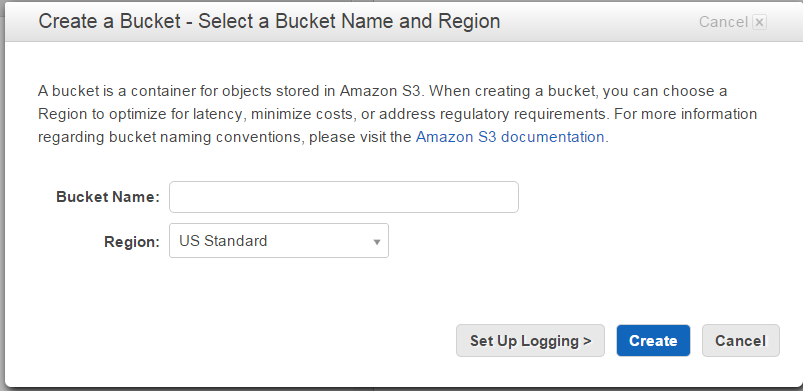
1. Go to the services and select S3 inside the Storage & Content Delivery.



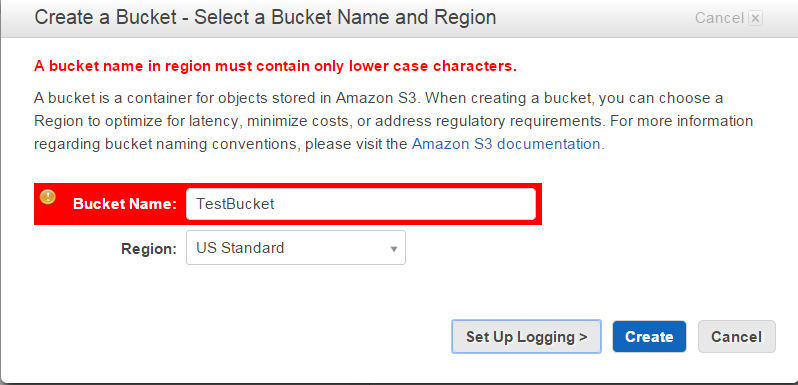
1. For creating the buckets select the create bucket option on top left of the web page.



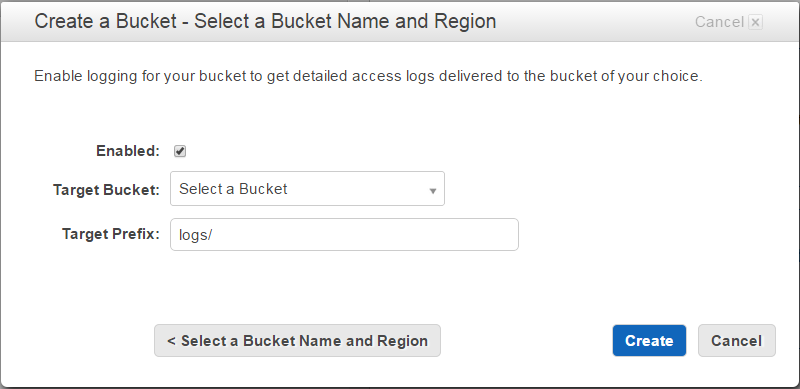
1. A popup window will appear then fill necessary details like bucket name and region.



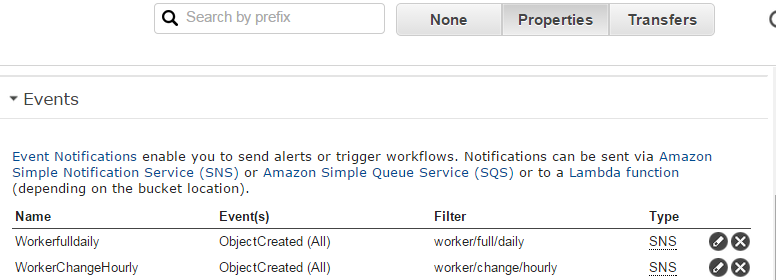
Make sure bucket name should be in lower case otherwise it will throw error.



1. If we want to set up the logging details for the bucket, then we need to select the Set Up Logging> option appear on the popup window. Just enable it and provide the bucket name where it should drop the logging files.

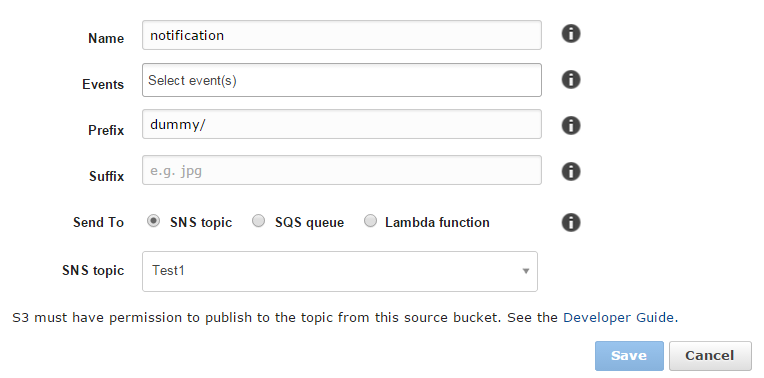


1. Inside the bucket if we want to create the multiple folders and we want to read the files which are received by the bucket then we need to configure the SNS topic with folders. We can do it by adding the events inside the properties.



1. After selecting the properties, we need to select the add notification option.



1. A set of input boxes will open where we need to add the properties.

Before adding any TOPIC name, we need to make sure that topic is already created in AWS. We can add only one topic for the same folder location.

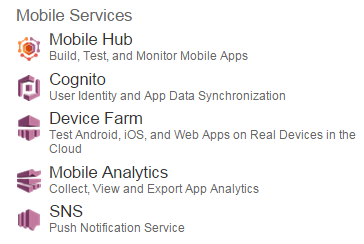
1. Amazon SNS Topic Creation Process

1. Login to the AWS with your credentials (Your credentials should have necessary access so that you create, read and delete the buckets when needed).

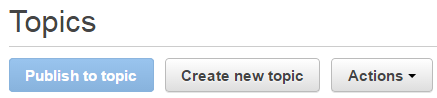
url :- <https://esb-integration-resources.signin.aws.amazon.com/console>

[https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2#](https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2)

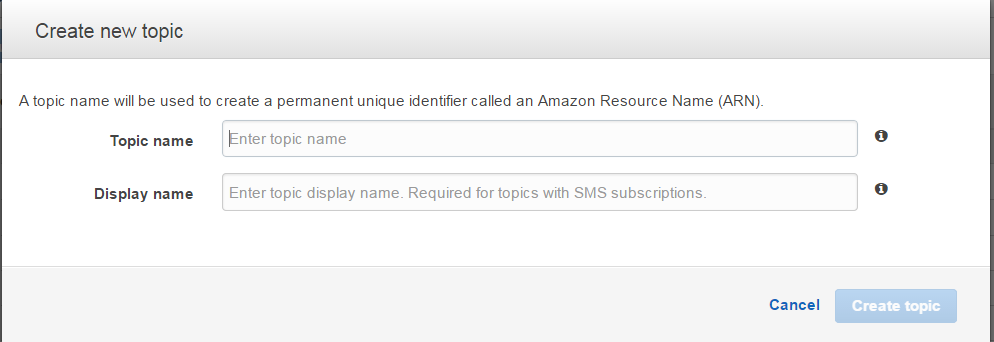
1. Go to the services and select SNS under the Mobile Service.



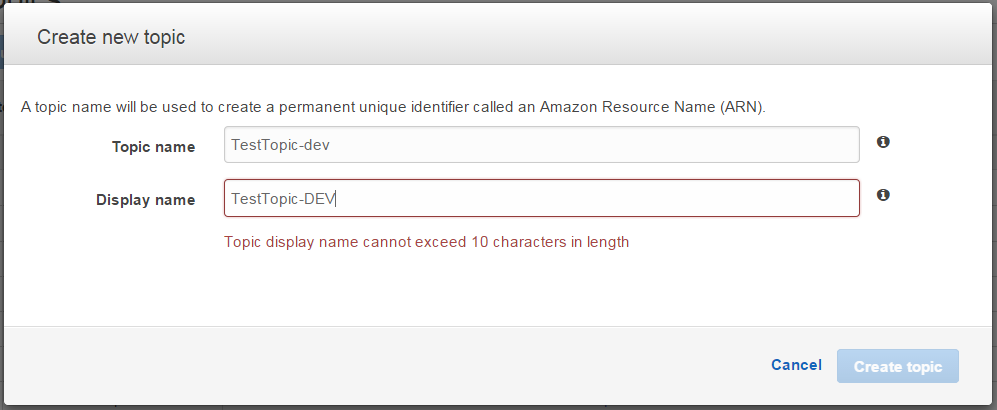
1. Inside TOPIC many options are there like Create Topic, Create Platform Application, Create Subscription, Publish Message and Publish Text Message. But out all options we need to select the Create topic option.



1. A popup window will open then we need to provide the essential information for that.



Make sure the Display name should not exceed the 10 characters.

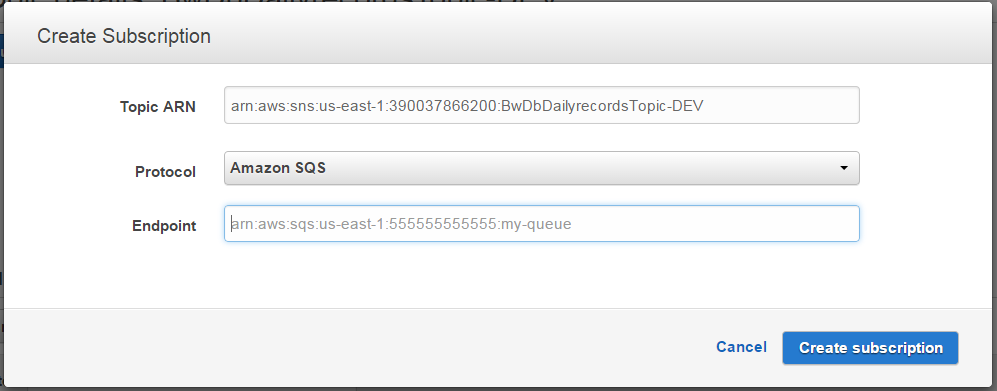


1. For creating the subscription select the topic which you want to subscribe then create subscription.

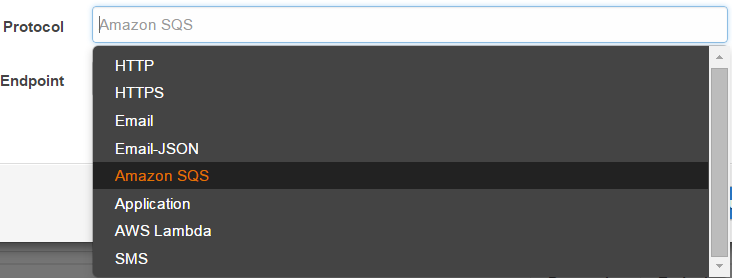


We can create as many numbers of subscribers for the same topic.

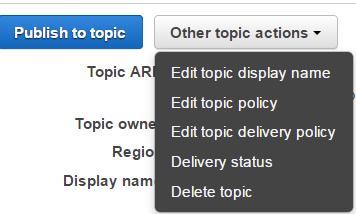
1. A popup window will open then we need to provide the necessary details in the window.



Topic ARN will come default, we need to select the protocol as we are creating subscription for Amazon SQS so selected it.



The Endpoint will be the ARN value of the Amazon SQS which is going to subscribe it. Before providing we need to make sure that the Amazon SQS is created in AWS.

1. If we want to change or update the policy of the TOPIC, then we need to select the Other topic actions and select edit topic policy. We can do this activity based on our requirement. 
2. Policies contains the set of configuration which we need to provide if we want to receive any notification from the S3 bucket. We need to add the bucket name from which topic will receive notification.

e.g. "ArnLike": {

"AWS:SourceOwner": "arn:aws:s3:\*:\*:<bucketName>"

}

We need to add two more configurations for providing the authentication to the S3 buckets to send the notification to topic, so that topic will publish and subscribe it.

e.g.

{

"Sid": "\_\_console\_pub\_0",

"Effect": "Allow",

"Principal": {

"AWS": "\*"

},

"Action": "SNS:Publish",

"Resource": "<TopicArn\_Value>"

},

{

"Sid": "\_\_console\_sub\_0",

"Effect": "Allow",

"Principal": {

"AWS": "\*"

},

"Action": ["SNS:Subscribe", "SNS:Receive"],

"Resource": "<TopicArn\_Value>"

}

1. After adding these policies, we need to update the policy and then try to create the notification event to configure the SNS with the S3 bucket. In the event we can create multiple events if we have the multiple folders inside the bucket.

Note:- If we have not configured the policy properly then the creation of the event for

buckets with topic is impossible.

1. We can publish the messages through publish to topic option.



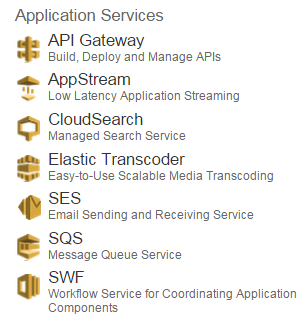
We can publish raw message as well as JSON message with subject names.

1. Amazon SQS Queue Creation Process

1. Login to the AWS with your credentials (Your credentials should have necessary access so that you create, read and delete the buckets when needed).

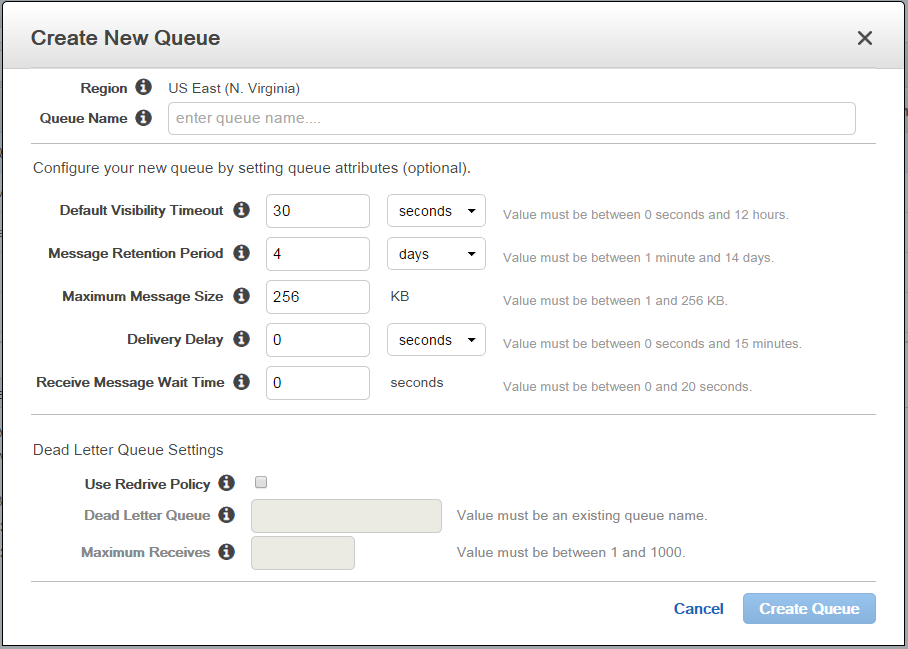
url :- <https://esb-integration-resources.signin.aws.amazon.com/console>

[https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2#](https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2)

1. Go to the Application Services and select SQS.
2. For creating the queues we need to select the create new queue option.

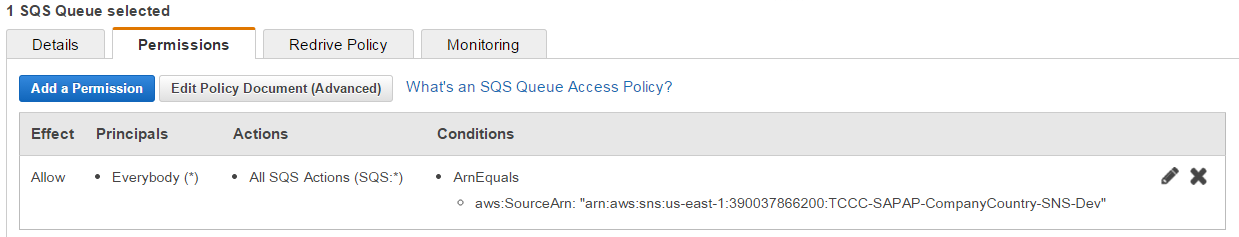


1. After selecting the creating new queue option, a popup window will open.

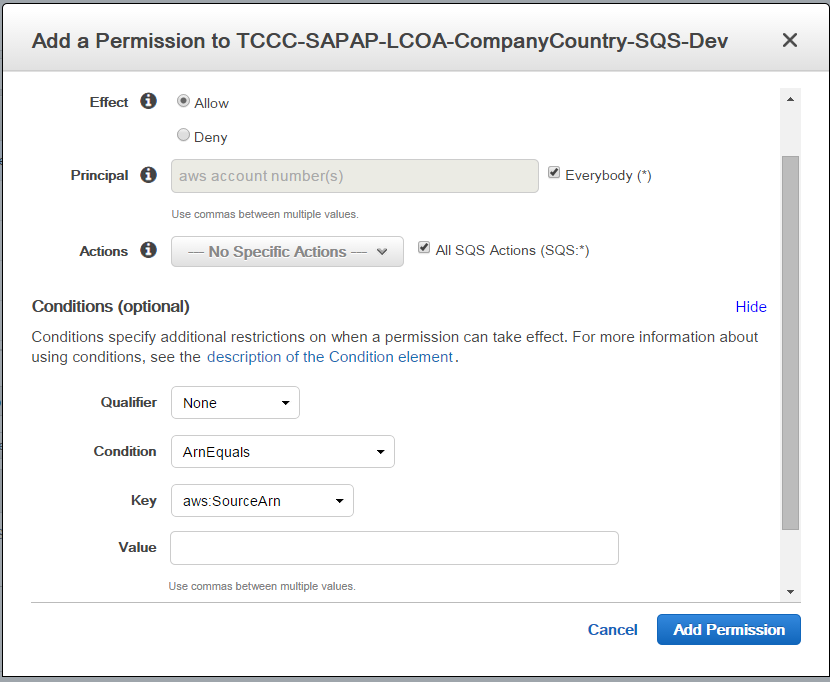


In this window we need to provide the queue name along with necessary properties which want to use for queue.

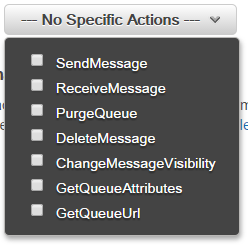
1. After creating the SQS if we want to subscribe the notification from topic then we need to provide the permissions. For providing the permission we need to select the queue, a property window will appear below the web page.



In this window we need to select Add a Permission option then we need to provide the following properties based on our requirement.

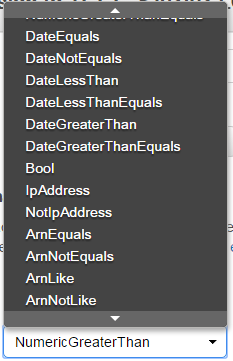


We can allow multiple user to use the same queue by adding them in principle. In Actions we can add multiple actions. For this we need to select the action, a drop down check box will where we can multiple actions or actions depends on our requirement.



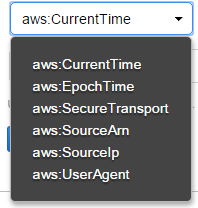
we can also select all SQS actions if we need all actions support instead of selecting the one by one from the drop down check box.

After this we need to select the condition we want to use for SQS.



Normally we are using ArnEquals condition.

Next we need to provide the key from the provided options.



Normally we are using the aws:SourceArn.

And then we need to provide the ARN value of the topic, this can contain multiple topic ARN values. We can add multiple conditions. After this we need to select the add permission.