

## Zadanie 2 – opis

W zadaniu 2 moim celem było dokonanie implementacji notyfikacji e-mail o tym, że dane znajdują się w utworzonym przy poprzednim zadaniu „bucketcie”. Do tego zadania użyłem serwisu SNS.

Na samym początku utworzyłem temat, ustawiłem typ jako „Standard” i nazwałem go „data\_uploaded”:

Amazon SNS > Topics > Create topic

### Create topic

**Details**

Type [Info](#)  
Topic type cannot be modified after topic is created

☐ FIFO (first-in, first-out)

- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS

☒ Standard

- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

**Name**

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).

**Display name - optional**  
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message. [Info](#)

Maximum 100 characters, including hyphens (-) and underscores (\_).

Następnie utworzyłem subskrypcję, ustawiłem „protokół” na „Email” i w zakładce „Endpoint” podałem maila, na który miałem wysłać powiadomienie.

W kolejnym kroku przeszedłem do mojego „bucketu” i w zakładce „Properties” przeszedłem do „Event notifications”, gdzie utworzyłem event. Typ ustawiłem na „Put”.

**Event notifications (1)** [Edit](#) [Delete](#) [Create event notification](#)

Send a notification when specific events occur in your bucket. [Learn more](#)

<input type="checkbox"/>	Name	Event types	Filters	Destination type	Destination
<input type="checkbox"/>	data_uploaded	Put	-	SNS topic	<a href="#">data_uploaded</a>

**Amazon EventBridge** [Learn more](#) or [see EventBridge pricing](#) [Edit](#)



For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications.


Send notifications to Amazon EventBridge for all events in this bucket

Off

Następnie w „Destination” wybrałem „SNS topic” i „Choose from your SNS topics”, gdzie wybrałem nazwę tematu:

**Destination**

 Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. [Learn more](#) 

**Destination**  
Choose a destination to publish the event. [Learn more](#) 

☐ **Lambda function**  
Run a Lambda function script based on S3 events.

☒ **SNS topic**  
Send notifications to email, SMS, or an HTTP endpoint.


☐ **SQS queue**  
Send notifications to an SQS queue to be read by a server.

**Specify SNS topic**  
☒ Choose from your SNS topics  
☐ Enter SNS topic ARN

**SNS topic**  

Choose SNS topic ▼

Niestety pojawił się błąd:

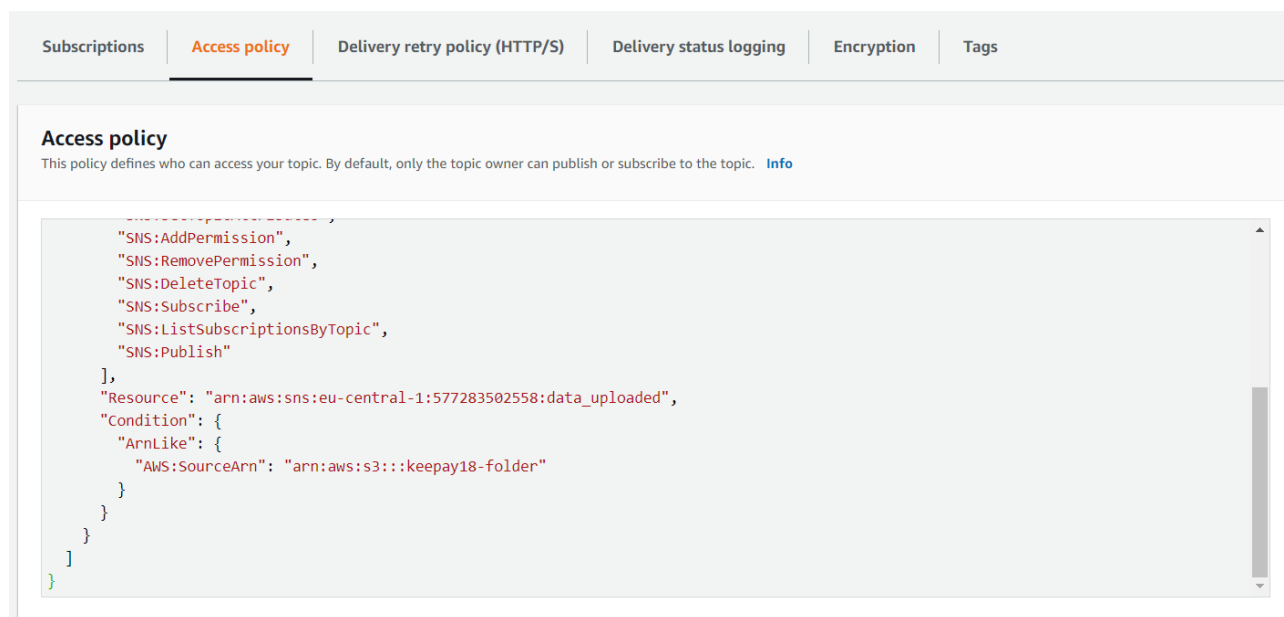
 **Unknown Error**  
An unexpected error occurred.

▼ API response

Unable to validate the following destination configurations

Wystąpił dlatego, ponieważ mój „bucket” nie jest publiczny.

Aby naprawić ten błąd, wróciłem do SNS i w zakładce „Access policy” zmieniłem kod na podany poniżej:



Musiałem podać „bucket ARN (Amazon Resource Name)”

Natomiast domyślny kod wygląda tak:



Po przedstawionych wyżej operacjach, notyfikacje będą przychodzić na podany wcześniej adres skrzynki mailowej.