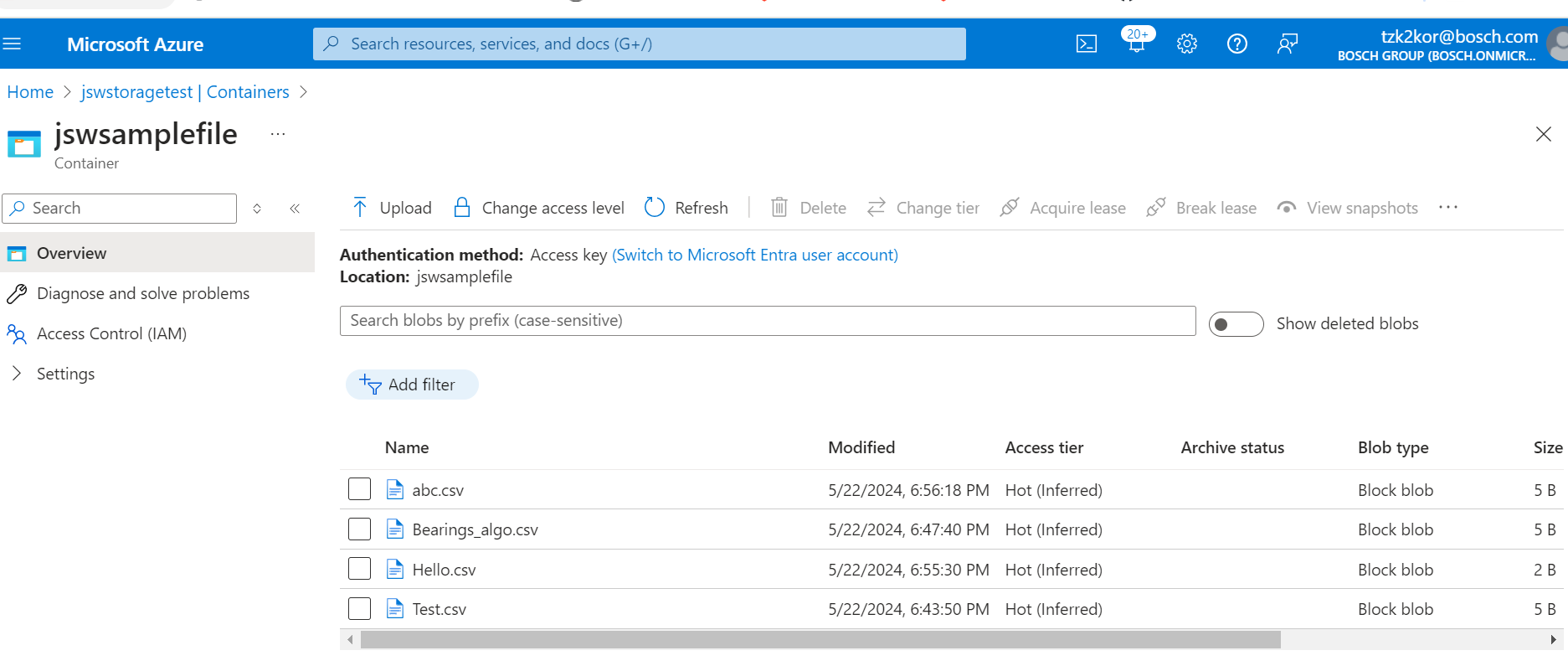
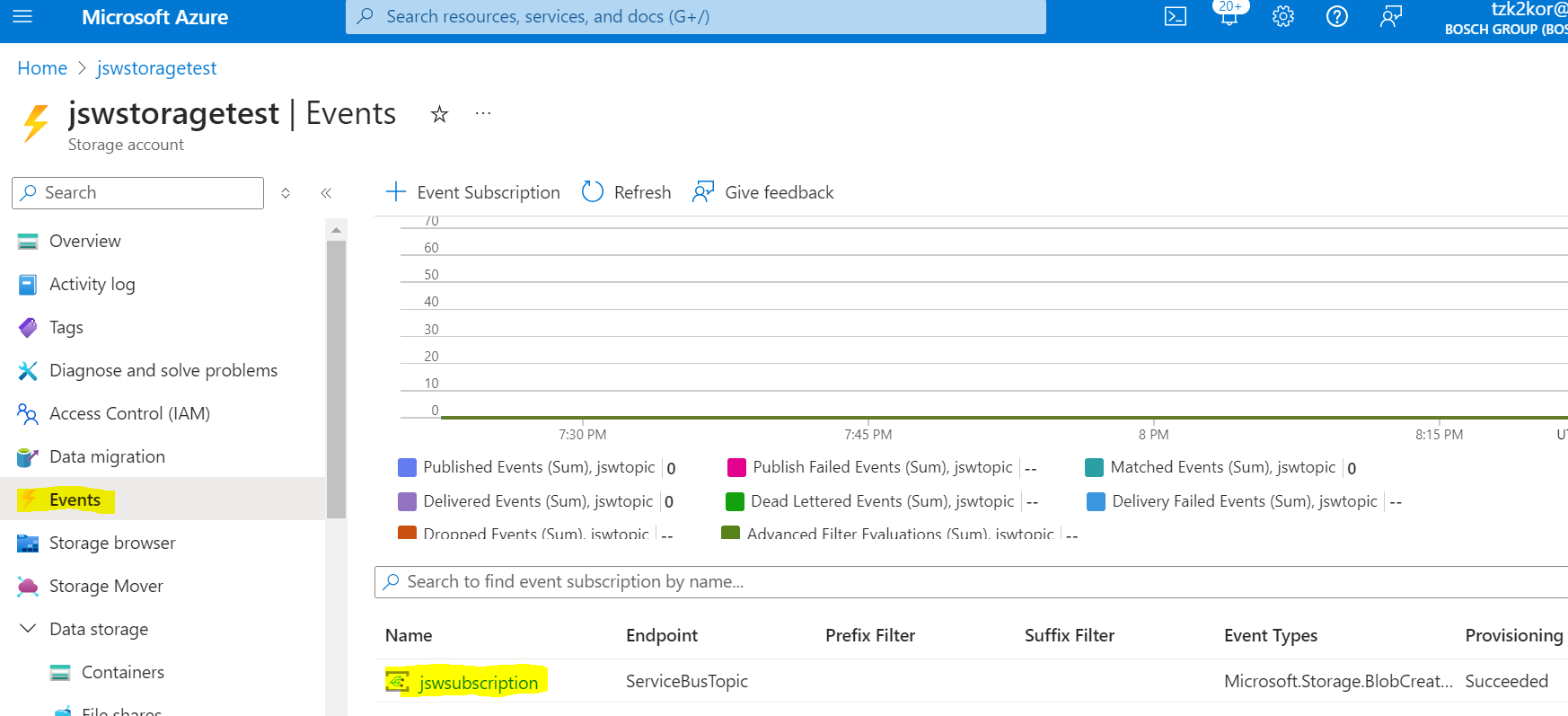
High Frequency Setup:

1. Create a Blob storage in Azure, to place the files as below :

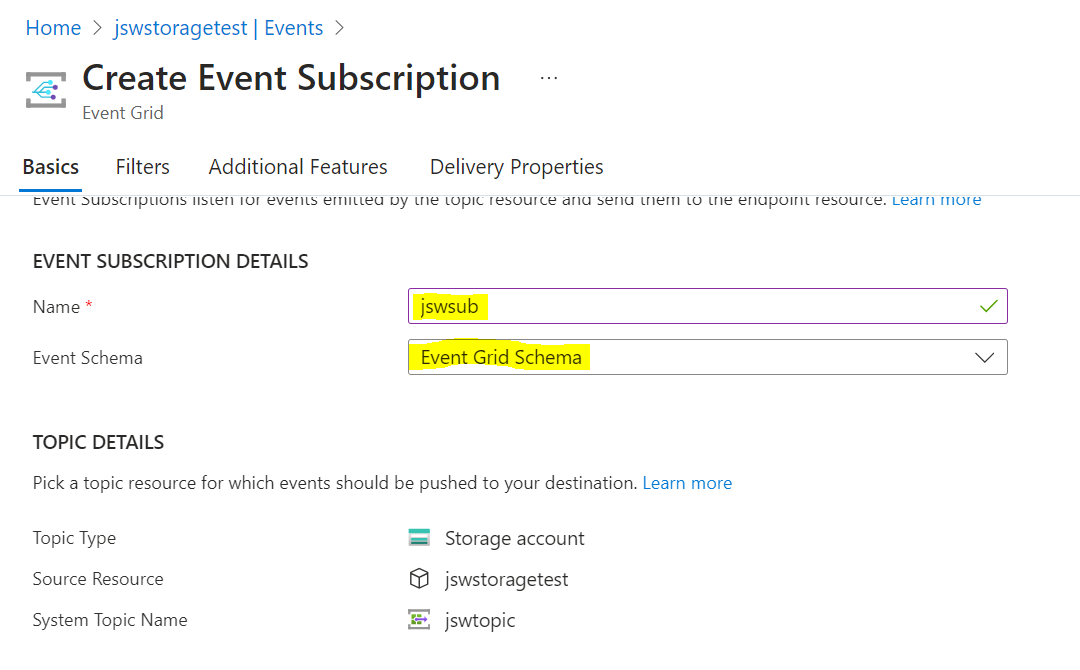


1. Create an Event Grid to the Blob Storage to Trigger an event whenever a file is placed in Blob as below.

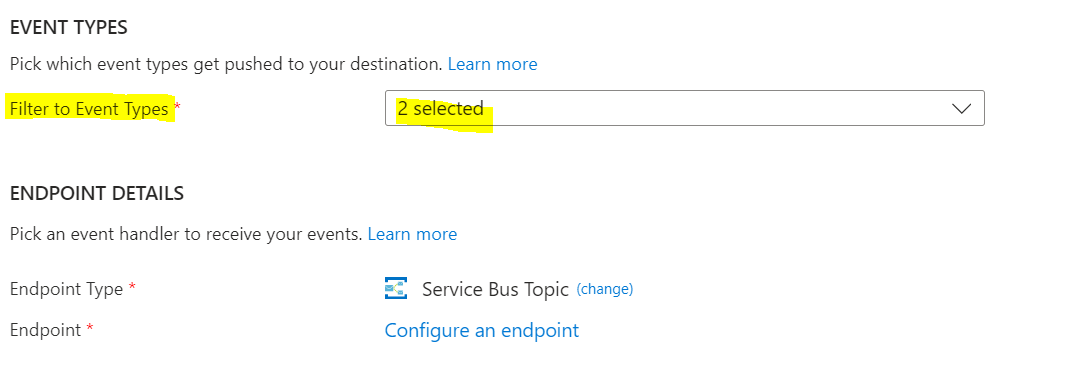


Below are the configuration of event Grid:

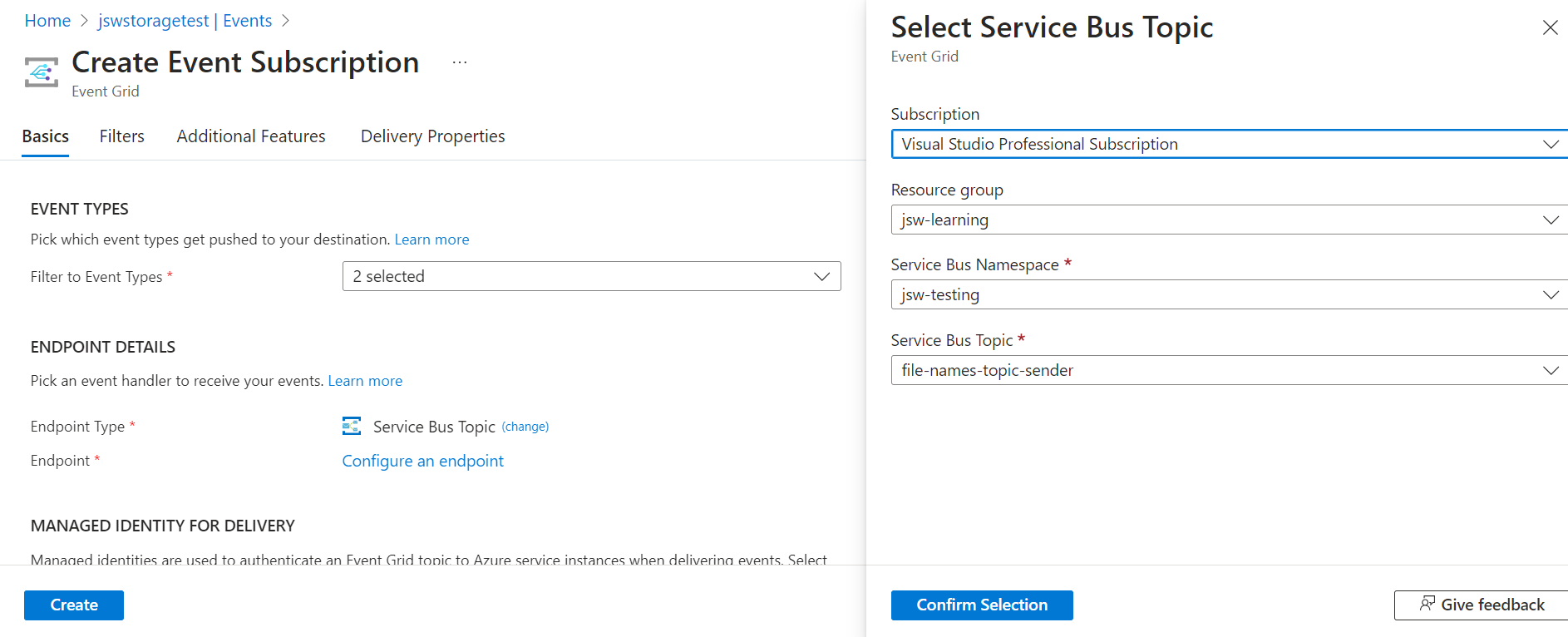
* **Event Schema: Event Grid**



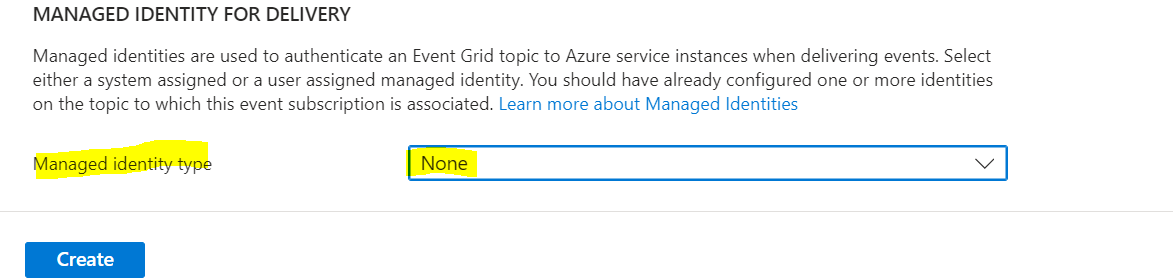
* **Event Type: Blob Created, Blob Deleted**

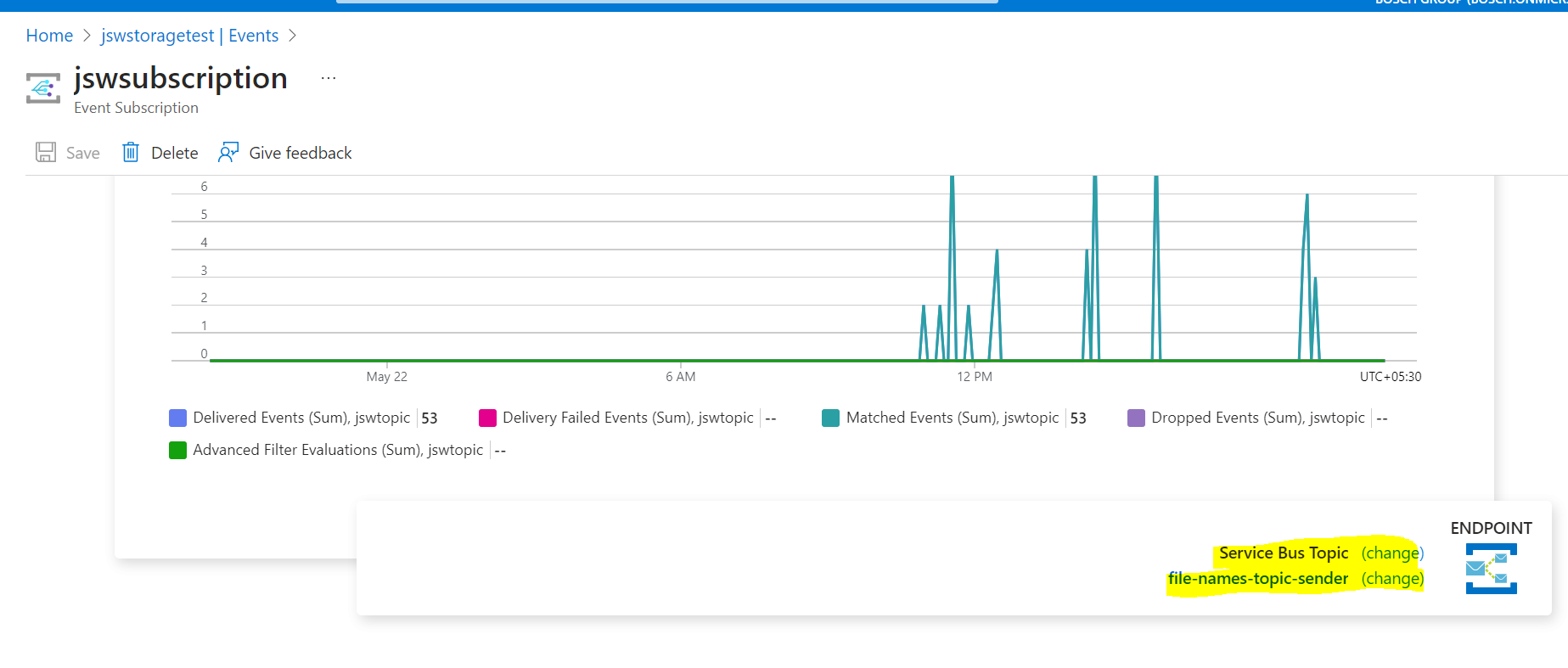


* **Endpoint Type: Service Bus Topic**

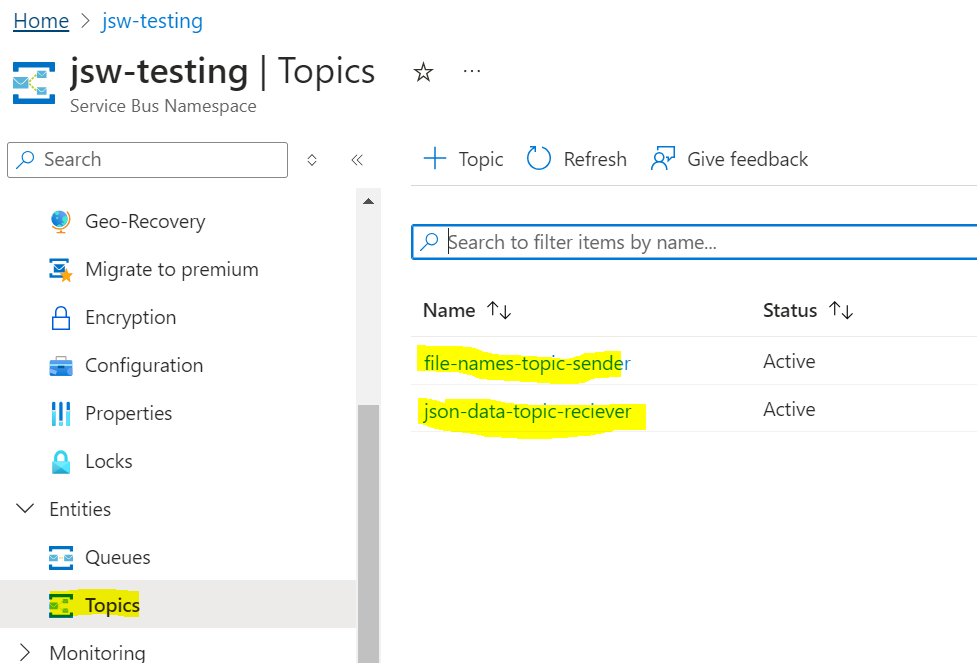


* **Managed Identity Type: None**



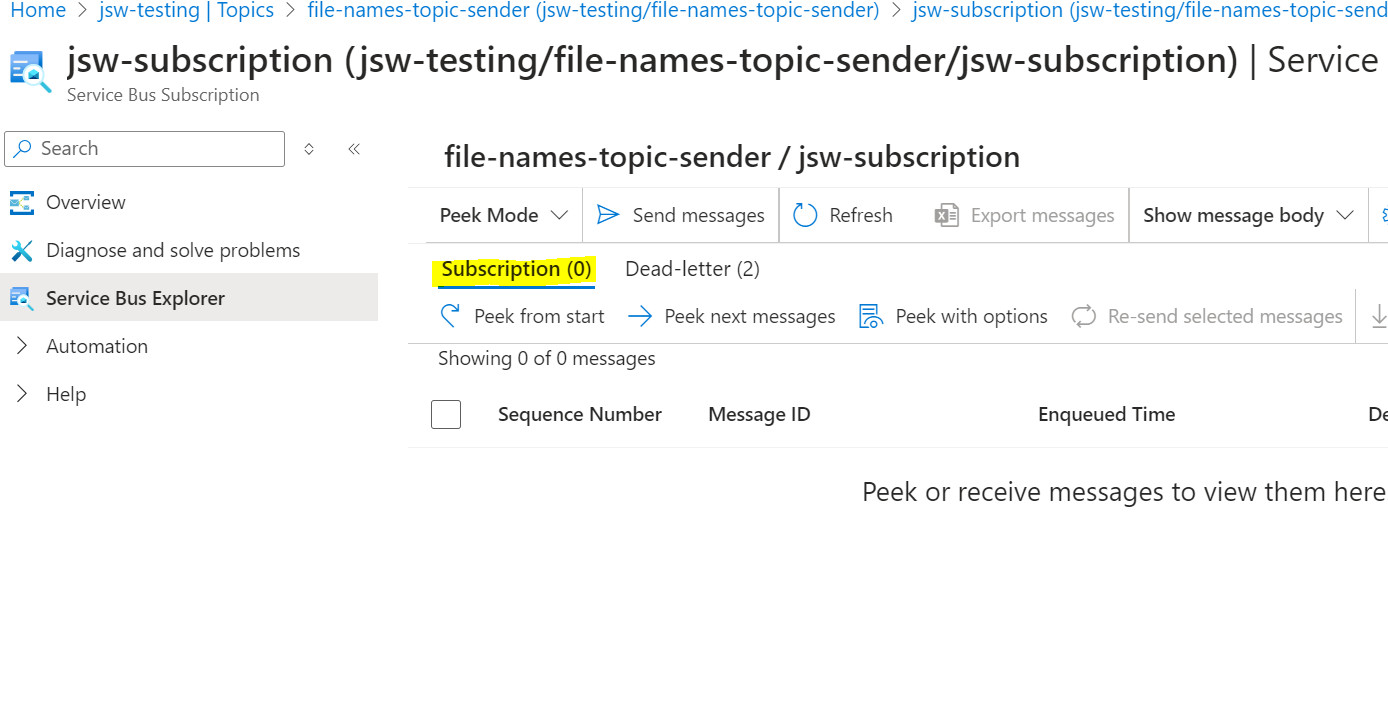


1. Create Two service Bus Topic one for sender and another for receiver as below :

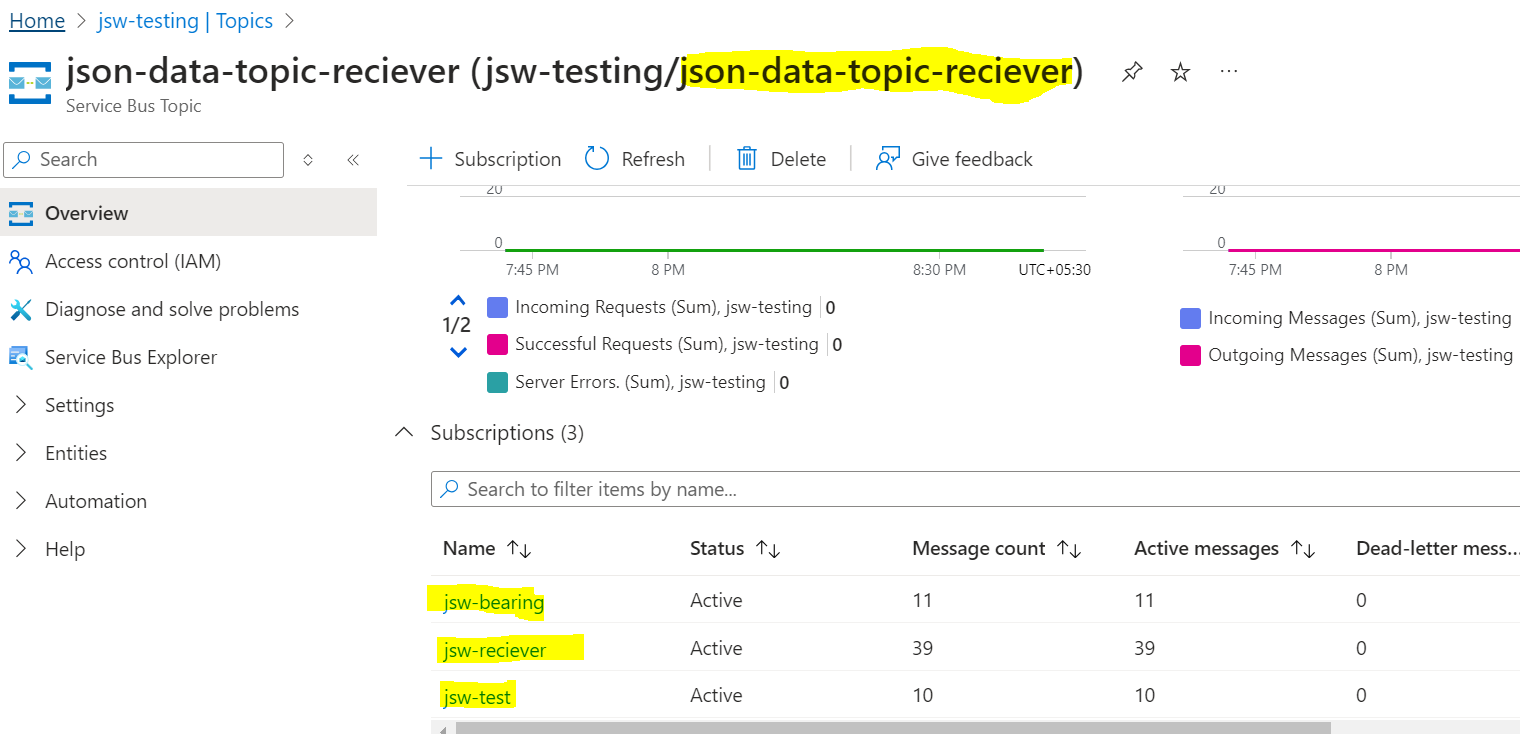


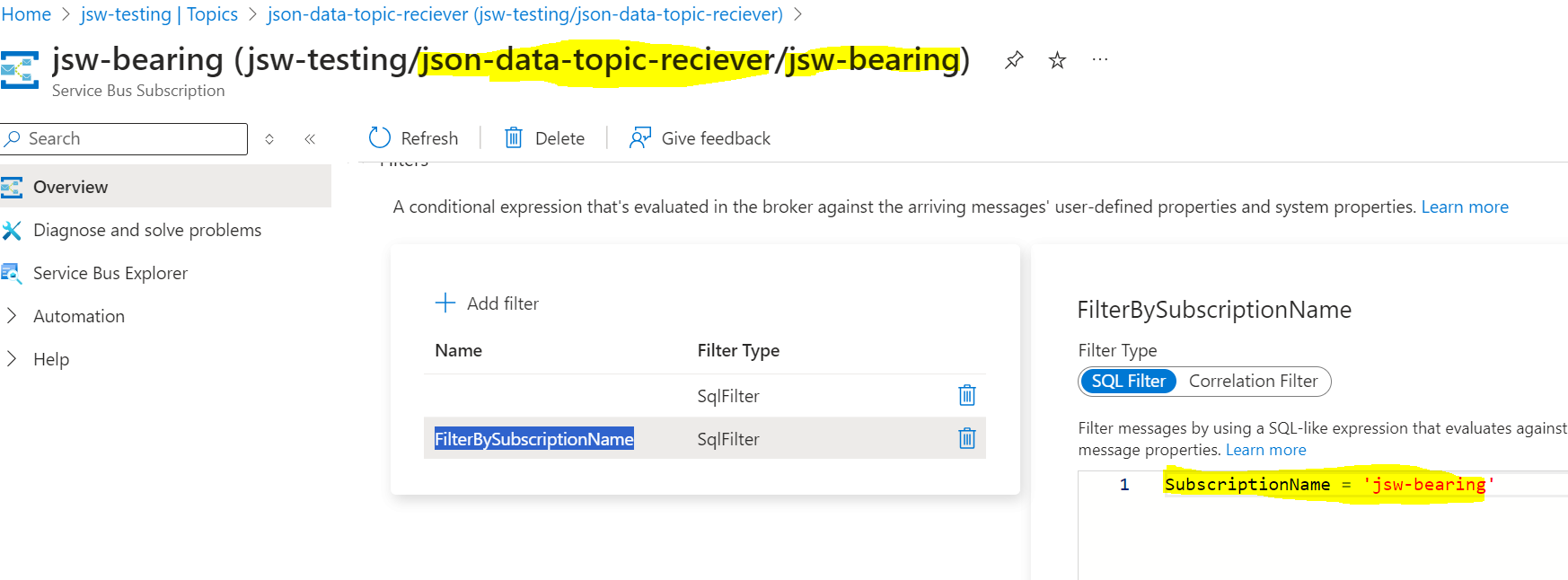
1. One Subscription can be created at sender side which will help us to view if the message is being read or it has passed into dead letter (Optional)

If the message is read correctly and picked by function app and send to receiver service bus topic the count of messages at the sender side will be 0 as below



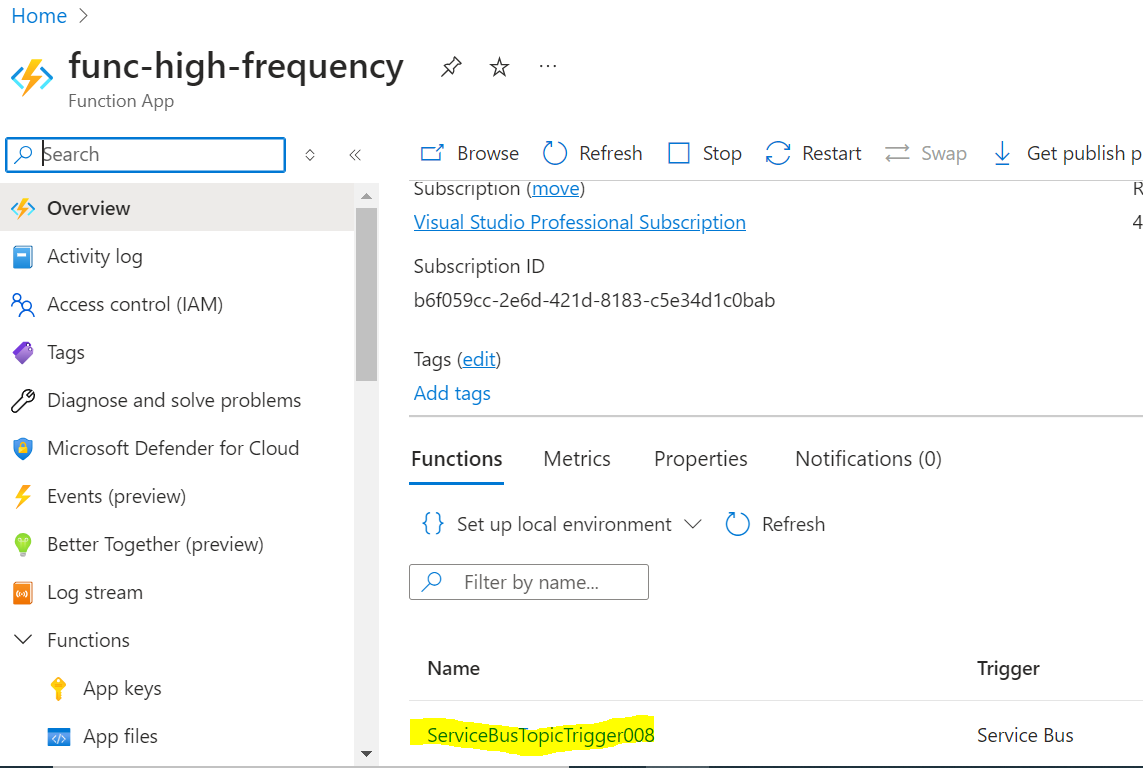
1. Service Bus Topic at the receiver should have different subscription, to redirect the messages based on the file name as below:



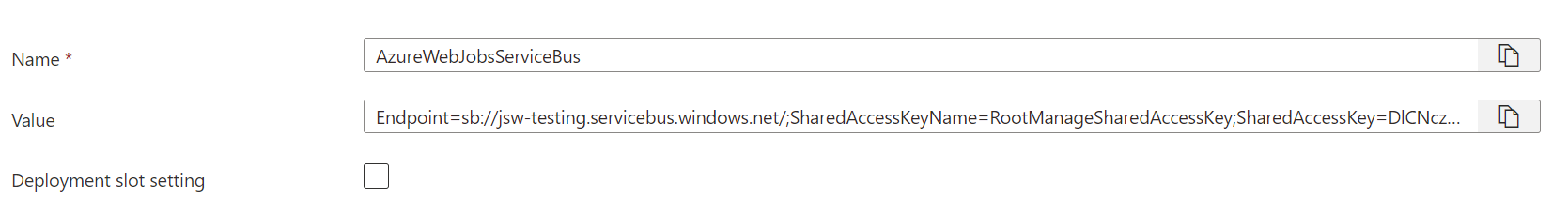
Filter should be applied to the each of the subscriptions , which help to place the data only in those particular subscriptions 

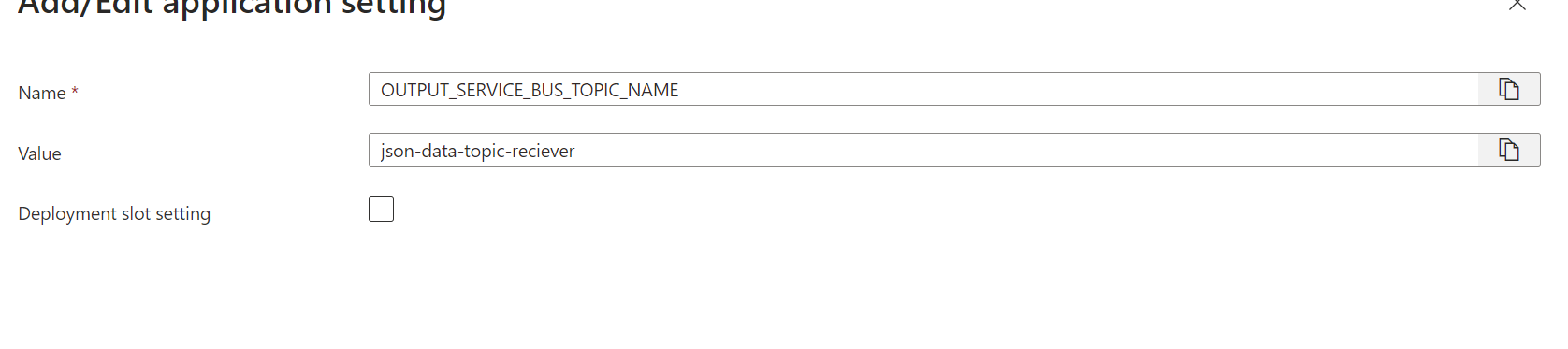
1. Create an Azure Function app , in order to pick the service bus topic message that gets places in sender when there is an event grid trigger and place it in service bus topic receiver subscription.

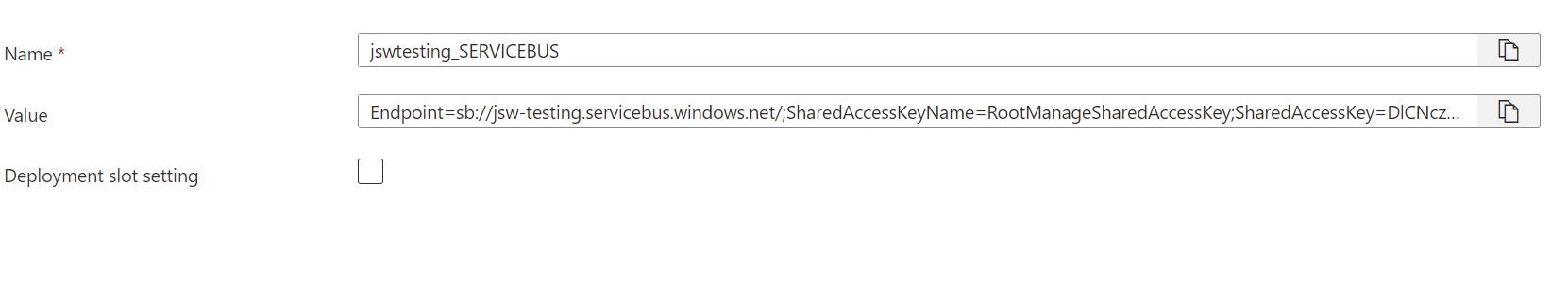
**Note**: Trigger type while creating the Azure Function Should be: Service Bus Topic



Configure the environment variable to have Connection string values and Receiver Service Bus topic name as below:

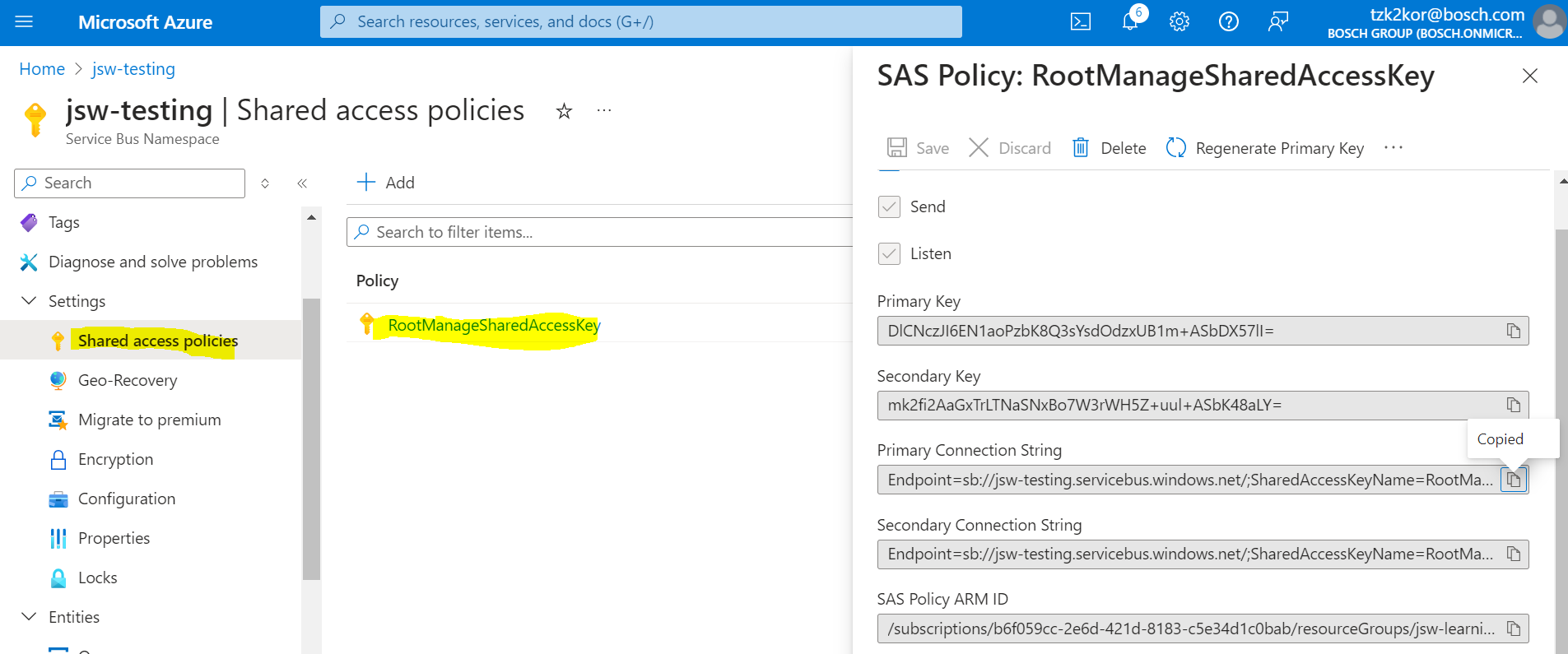






**Note:**

Connection String value will be picked from Shared Access Token of the service Bus Namespace created ( it is not specific to any service bus topic or service bus queue)



* Below is the code that is being used in the function app:

import os

import logging

import json

import azure.functions as func

from azure.servicebus import ServiceBusClient, ServiceBusMessage

app = func.FunctionApp()

@app.function\_name(name="ServiceBusTopicTrigger008")

@app.service\_bus\_topic\_trigger(arg\_name="message",

                               topic\_name="file-names-topic-sender",

                               connection="jswtesting\_SERVICEBUS",

                               subscription\_name="jsw-subscription")

def test\_function(message: func.ServiceBusMessage):

    try:

        # Read the message payload

        message\_body = message.get\_body().decode("utf-8")

        logging.info("Python ServiceBus topic trigger processed message.")

        logging.info("Message Body: %s", message\_body)

        # Parse the message body as JSON

        event = json.loads(message\_body)

        # Function to extract filename from subject

        def extract\_filename\_from\_subject(event):

            try:

                subject = event.get('subject', '')

                filename = subject.split('/')[-1]

                return filename

            except Exception as e:

                logging.error(f"Error extracting filename: {e}")

                return None

        # Extract the filename

        filename = extract\_filename\_from\_subject(event)

        if filename:

            logging.info("Extracted filename: %s", filename)

            # Determine the subscription based on the filename

            subscription\_name = determine\_subscription\_name(filename)

            if subscription\_name:

                # Initialize a Service Bus client for the receiver topic

                receiver\_connection\_string = os.environ["jswtesting\_SERVICEBUS"]

                receiver\_topic\_name = os.environ["RECIEVER\_TOPIC"]

                # Send the message to the determined subscription

                with ServiceBusClient.from\_connection\_string(receiver\_connection\_string) as receiver\_client:

                    with receiver\_client.get\_topic\_sender(receiver\_topic\_name) as sender:

                        # Create a new message with the same payload and add a custom property

                        service\_bus\_message = ServiceBusMessage(message\_body)

                        service\_bus\_message.application\_properties = {"SubscriptionName": subscription\_name}

                        logging.info("Message created for sending to subscription: %s", subscription\_name)

                        sender.send\_messages(service\_bus\_message)

                        logging.info("Message sent to topic: %s, subscription: %s", receiver\_topic\_name, subscription\_name)

                logging.info(f"Forwarded message to {receiver\_topic\_name}, subscription: {subscription\_name}: {message\_body}")

            else:

                logging.warning("No subscription determined based on filename.")

        else:

            logging.warning("No filename extracted from the subject.")

    except Exception as e:

        logging.error(f"Error processing message: {e}")

def determine\_subscription\_name(filename):

    # Logic to determine subscription based on filename

    if filename == "Bearings\_algo.csv":

        return "jsw-bearing"

    elif filename == "Test.csv":

        return "jsw-test"

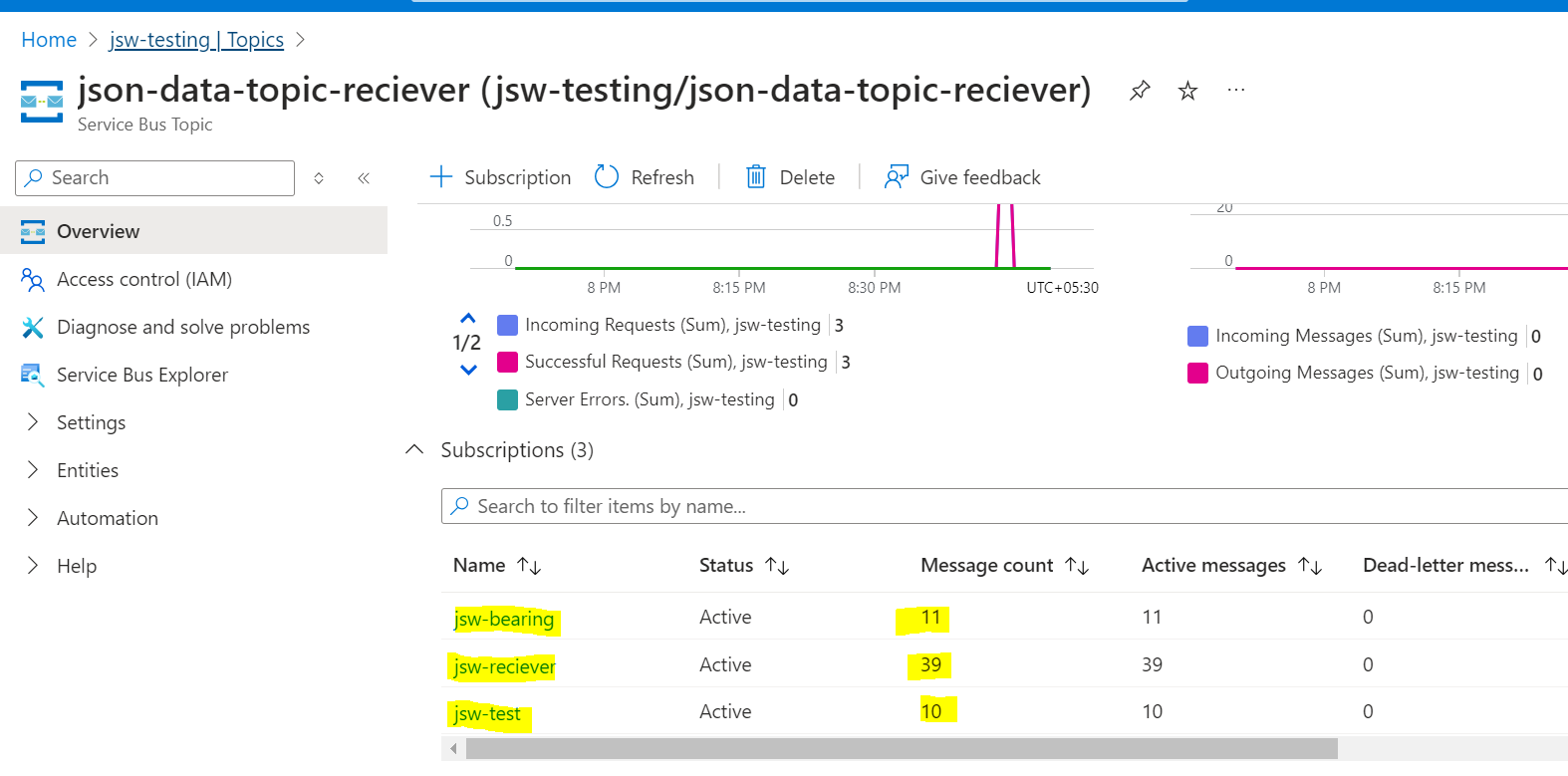
    elif filename == "abc.csv":

        return "jsw-receiver"

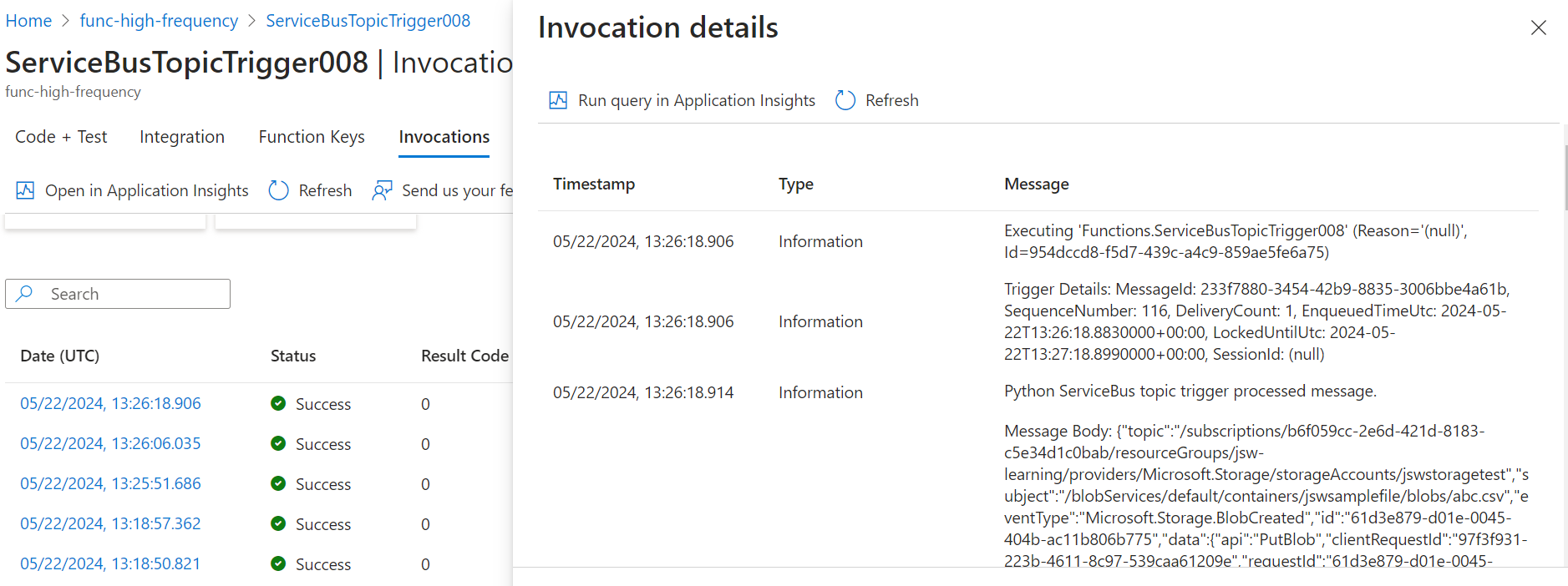
    else:

        return None

1. After the message are successfully redirect to the correct subscription, they can be viewed thorough the service bus explorer and see the count increasing in their respective service bus topic subscription



**Note**: To Monitor the Azure Function App Logs as below:



**Summary**:

Blob (Place a file) -🡪 Event Grid Trigger (Blob ) -🡪 Service Bus Topic Sender (1 Subscription) 🡪 Function App (Service Bus Topic Trigger) 🡪 Service Bus Topic Receiver (3 Subscription)