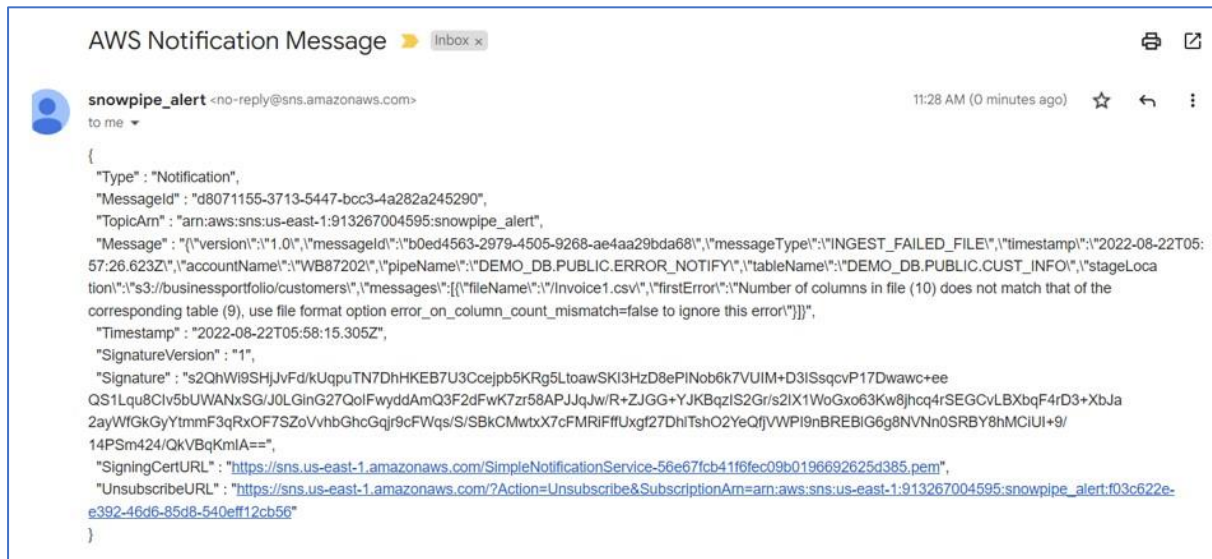


Snowpipe: Send Email notification on Error



During this post we will see how we can configure the ALERT or EMAIL mechanism from Snowpipe. Say you have configured the Snowpipe in your account. In case of any file processing via Snowpipe, if error comes we don't have any alert mechanism in place. Moreover, We need to monitor the status of pipe manually to confirm the status of load. To ease our monitoring process we will configure the Email notification which would trigger the mail in case of any failure.

Steps to be included:

1. SNS Service: Firstly, Configure Simple Notification Service (Amazon SNS) is a managed service that provides message delivery from publishers to subscribers. Publishers communicate asynchronously with subscribers by sending messages to a Clients can subscribe to the SNS topic and receive published messages using a supported endpoint type, such as Amazon Kinesis Data Firehose, Amazon SQS

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.

SNS

- Select Everyone for this use case in Access Policy-optional section.
- Create Topic and noted down the generated ARN.

2. Secondly, Create subscription to whom mail will be sent.

Create subscription

Details

Topic ARN

Protocol
The type of endpoint to subscribe

Endpoint
An email address that can receive notifications from Amazon SNS.

After your subscription is created, you must confirm it. [Info](#)

☐ f03c622e-e392-46d6-85d8-540eff12cb56
sachin.mittal04@gmail.com
☒ Confirmed
EMAIL-JSON
snowpipe_alert

- An email will be sent to mail id specify in Endpoint.
- Confirm the subscription and your subscription will be confirmed like below

3. Thirdly, and most importantly, Go to AWS Management Console and create Policy : **snowflake_sns_error_notification**

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the console.

Visual editor **JSON**

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sns:Publish"
8       ],
9       "Resource": "arn:aws:sns:us-east-1:913267004595:snowpipe_alert"
10    }
11  ]
12 }
```

Policy

4. Moreover, Create Role and Associate Policy.

☐ **AWS service**
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

☒ **AWS account**
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ **Web identity**
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

☐ **SAML 2.0 federation**
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

☐ **Custom trust policy**
Create a custom trust policy to enable others to perform actions in this account.

An AWS account

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ This account (913267004595)

☒ **Another AWS account**

Account ID
Identifier of the account that can use this role

913267004595

Account ID is a 12-digit number.

Options

☒ **Require external ID (Best practice when a third party will assume this role)**
You can increase the security of your role by requiring an optional external identifier, which prevents "confused deputy" attacks. This is recommended if you do not own or have administrative access to the account that can assume this role. The external ID can include any characters that you choose. To assume this role, users must be in the trusted account and provide this exact external ID. [Learn more](#)

External ID

1234567

Important: The console does not support using an external ID with the Switch Role feature. If you select this option, entities in the trusted account must use the API, CLI, or a custom federation proxy to make cross-account iam:AssumeRole calls. [Learn more](#)

Snowpipe Role

- Associate your previous created Policy
- Name the Role : snowpipe_SNS_Error_Role

5. Creating the Notification Integration.

- In addition, Create a notification integration using CREATE NOTIFICATION INTEGRATION. An integration is a Snowflake object that references the SNS topic you created

```
CREATE NOTIFICATION INTEGRATION Snowpipe_Error_Notify
ENABLED = true
TYPE = QUEUE
NOTIFICATION_PROVIDER = AWS_SNS
DIRECTION = OUTBOUND
AWS_SNS_TOPIC_ARN = 'arn:aws:sns:us-east-1:913267004595:snowpipe_alert'
AWS_SNS_ROLE_ARN = 'arn:aws:iam::913267004595:role/snowpipe_SNS_Error_Role'
```

- DESC notification integration Snowpipe_Error_Notify;
- Note down: SF_AWS_IAM_USER_ARN, SF_AWS_EXTERNAL_ID

Snowpipe Role

- Associate your previous created Policy
- Name the Role : snowpipe_SNS_Error_Role

5. Creating the Notification Integration.

- In addition, Create a notification integration using CREATE NOTIFICATION INTEGRATION. An integration is a Snowflake object that references the SNS topic you created

```
CREATE NOTIFICATION INTEGRATION Snowpipe_Error_Notify
ENABLED = true
TYPE = QUEUE
NOTIFICATION_PROVIDER = AWS_SNS
DIRECTION = OUTBOUND
AWS_SNS_TOPIC_ARN = 'arn:aws:sns:us-east-1:913267004595:snowpipe_alert'
AWS_SNS_ROLE_ARN = 'arn:aws:iam::913267004595:role/snowpipe_SNS_Error_Role'
```

- DESC notification integration Snowpipe_Error_Notify;
- Note down: SF_AWS_IAM_USER_ARN, SF_AWS_EXTERNAL_ID

property	property_type	property_value
ENABLED	Boolean	true
NOTIFICATION_PROVIDER	String	AWS_SNS
DIRECTION	String	OUTBOUND
AWS_SNS_TOPIC_ARN	String	arn:aws:sns:us-east-1:913267004595:snowpipe_alert
AWS_SNS_ROLE_ARN	String	arn:aws:iam::913267004595:role/snowpipe_SNS_Error_Role
SF_AWS_IAM_USER_ARN	String	arn:aws:iam::397374258482:user/g263-s-ohsx4170
SF_AWS_EXTERNAL_ID	String	WB87202_SF_CRole=2_rMGx7JMefUSYI8/ByOTtcyxU8eQ=
COMMENT	String	

Notification Info

6. After that, Modify the Trust Relationship in the IAM Role with above noted ARN and External ID.

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Principal": {
7         "AWS": "arn:aws:iam::397374258482:user/g263-s-ohsx4170"
8       },
9       "Action": "sts:AssumeRole",
10      "Condition": {
11        "StringEquals": {
12          "sts:ExternalId": "WB87202_SFCRole=2_rMGx7JMeFUSYI8/ByOTtcyxU8eQ="
13        }
14      }
15    ]
16  }
17 }
```

Trust Policy

7. Above all, Create PIPE and Enabling Error Notifications.

```
CREATE or replace PIPE demo_db.public.error_notify auto_ingest=true
ERROR_INTEGRATION = Snowpipe_Error_Notify
AS copy into CUST_INFO from @demo_db.public.ext_csv_stage/customers
```

8. Finally, Configure the PIPE ARN with S3 Bucket and upload the file to bucket. This will trigger the PIPE and in case of any error an email will be triggered to our mail ID.

snowpipe_alert <no-reply@sns.amazonaws.com>
to me

11:28 AM (0 minutes ago)

```
{
  "Type": "Notification",
  "MessageId": "d8071155-3713-5447-bcc3-4a282a245290",
  "TopicArn": "arn:aws:sns:us-east-1:913267004595:snowpipe_alert",
  "Message": "(\"version\":\"1.0\", \"messageId\":\"b0ed4563-2979-4505-9268-ae4aa29bda68\", \"messageType\":\"INGEST_FAILED_FILE\", \"timestamp\":\"2022-08-22T05:57:26.623Z\", \"accountName\":\"WB87202\", \"pipeName\":\"DEMO_DB.PUBLIC.ERROR_NOTIFY\", \"tableName\":\"DEMO_DB.PUBLIC.CUST_INFO\", \"stageLocation\":\"s3://businessportfolio/customers\", \"messages\": [{ \"fileName\":\"/Invoice1.csv\", \"firstError\": \"Number of columns in file (10) does not match that of the corresponding table (9), use file format option error_on_column_count_mismatch=false to ignore this error\" } ] )\",
  "Timestamp": "2022-08-22T05:58:15.305Z",
  "SignatureVersion": "1",
  "Signature": "s2QhW9SHjvFdKUpuTN7DhHKEB7U3Ccejpb5KRg5LtoawSKi3HzD8ePINob6k7VUIM+D3ISsqvP17Dwawc+eeQS1Lq8Civ5bUWANxSGJ0LGING27QoIFwyddAmQ3F2dFwK7zr58APJJqJw/R+ZJGG+YJKBqzIS2Gr/s2IX1WoGxo63Kw8jhcq4rSEGCVLBxqF4rD3+XbJa2ayWfGkGyYtmMF3qRxOF7SzoVvhbGhcGqjr9cFWqs/S/BKcMwtX7cFMRIFfUxgf27DhTshO2YeQfjVWPi9nBREBiG6g8NVNn0SRBY8hMCiUI+9/14PSm424/QkVBqKmlA==",
  "SigningCertURL": "https://sns.us-east-1.amazonaws.com/SimpleNotificationService-56e67fcb41f6fec09b0196692625d385.pem",
  "UnsubscribeURL": "https://sns.us-east-1.amazonaws.com/?Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-east-1:913267004595:snowpipe_alert:f03c622e-e392-46d6-85d8-540eff12cb56"
}
```

Notification Msg

```
select FILE_NAME, ROW_COUNT, ROW_PARSED, ERROR_COUNT, STATUS, PIPE_NAME from
table(information_schema.copy_history(table_name=>'CUST_INFO', start_time=> dateadd(day, -1, current_timestamp()))) where row_count <> row_parsed;
```

Query ID: 1.41s 1 rows

Row	FILE_NAME	ROW_COUNT	ROW_PARSED	ERROR_COUNT	STATUS	PIPE_NAME
1	/Invoice1.csv	0	7	7	Load failed	ERROR_NOTIFY

Snowpipe Error