FANOUT SERVICE :

1. Receive messages from producer and process 1 by 1 (below is 1 message) :

```
[
{
  messageId: '81bad750-c6c4-c50f-a75b-2f87d2279f24',
  receiptHandle:
'mltrnmfjisvzlibrnyufrklubhbwqszswoblmjctopmjtpxfwkhcbrebxtvfvhjhnrdzlpltay
yrxayttcqomazfljiabgsxfgmgdggoncnbrqaawpzduhmlqseykqlvfopafwqwcowcm
gplwaaxfderomkjtoqbtxgozqbpfpltgpypbqaswzpal',
  md5OfBody: 'f3c697fcc0cea54fd9e8db060576da18',
  body: {
    timestampFromProducer: '2021-07-30T18:02:34.910Z',
    triggerType: 'data-contract',
    triggerExecutionId: 'NOTUSED',
    businessEventId: '36bff171ba9f40bd8d06a9123dc01796',
    businessEventInstanceId: 'd0043448-9f2e-4ddb-8abe-0b3bb75abb66',
    businessEventArrivalTimestamp: 1627668154862,
    'x-trace-id': '4ba95fdd-a653-4a2f-a06b-f05d124a721c',
    'x-from-span-id': '128f9631-922a-4cb4-9e71-16c53a16c35c',
    explicitContext: 'global',
    associateId: 'ea64cf42-bda2-4a35-a721-d68815e47f74',
    clientId: '002',
    mask: '53331331531fa2ff56782738c5dee1bdfb884ab7',
    userContext: 'US'
  },
  attributes: {
    senderId: 'AIDAIT2UOQQY3AUEKVGXU',
    sentTimestamp: '1627668154937',
    approximateReceiveCount: '1',
    approximateFirstReceiveTimestamp: '1627668154942',
    messageDeduplicationId: '080e07e6-4ad1-46f3-950d-41774b0d8e90',
    messageGroupId: '36bff171ba9f40bd8d06a9123dc01796.002'
  }
}
]
```

2. Then checks messageGroupId and takes clientid and then sees if any getDisabledConsumers r there for that using generateRuntimeSuppressionGetQuery
3. We will differentiate using key and return disabledMdos, disabledMiniappConsumers from getDisabledConsumers.
4. We process message : params to be passes : messageLogger, message,

disabledConsumers

5. Extract businesseventid from above process message function and then search if it has any consumers ??? Using dependents endpoint - getMetadataForConsumers function :

'http://router/api/ohcm-search-service/v0/${mask}/${businessEventId}/dependants';

Using Lou cache mech. (Storing consumer ids in Lru)

6. For each metadata consumer we will format payload and based on consumer type we will send to that consumer service.
7.