Event Hub Testing with Service Bus Explorer and CluedIn

Purpose

Let's use service bus explorer from the very beginning to understand what a freshly minted event hub looks like and how to configure it for use with CluedIn.

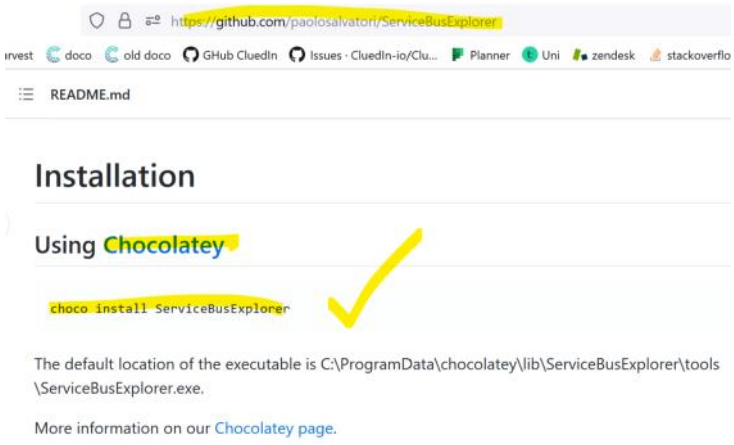
Results

Events seen in the event hub as sent from CluedIn.

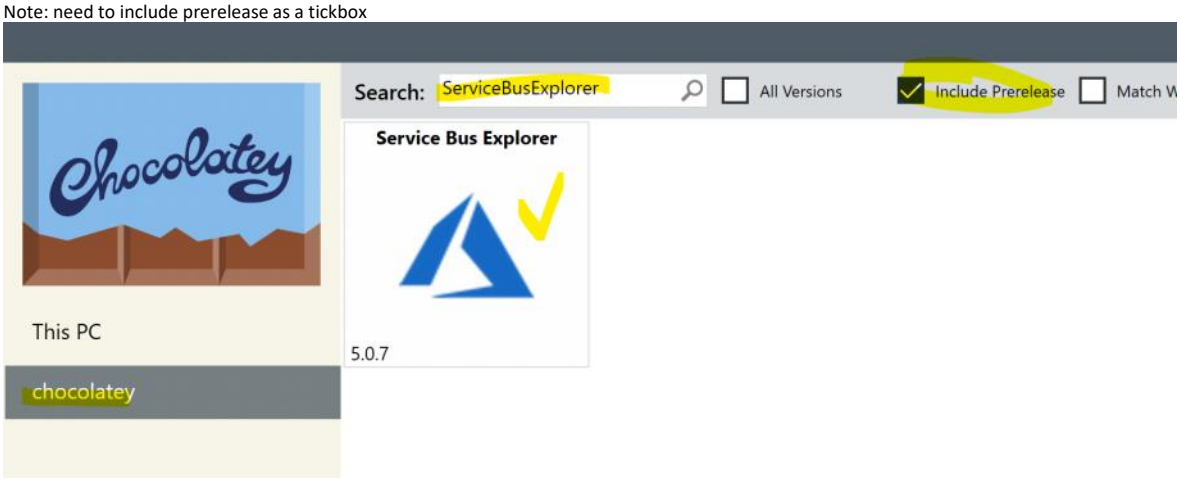
Method

Setup

Install Service Bus Explorer



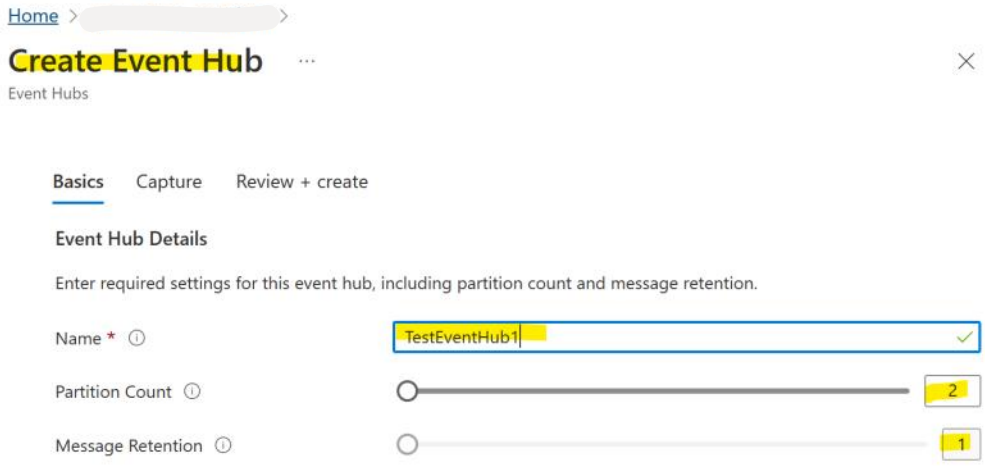
Note: need to include prerelease as a tickbox

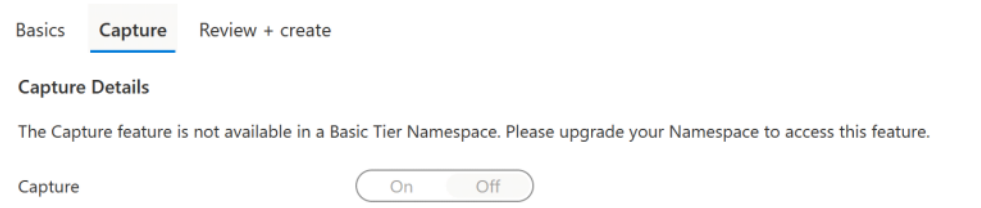


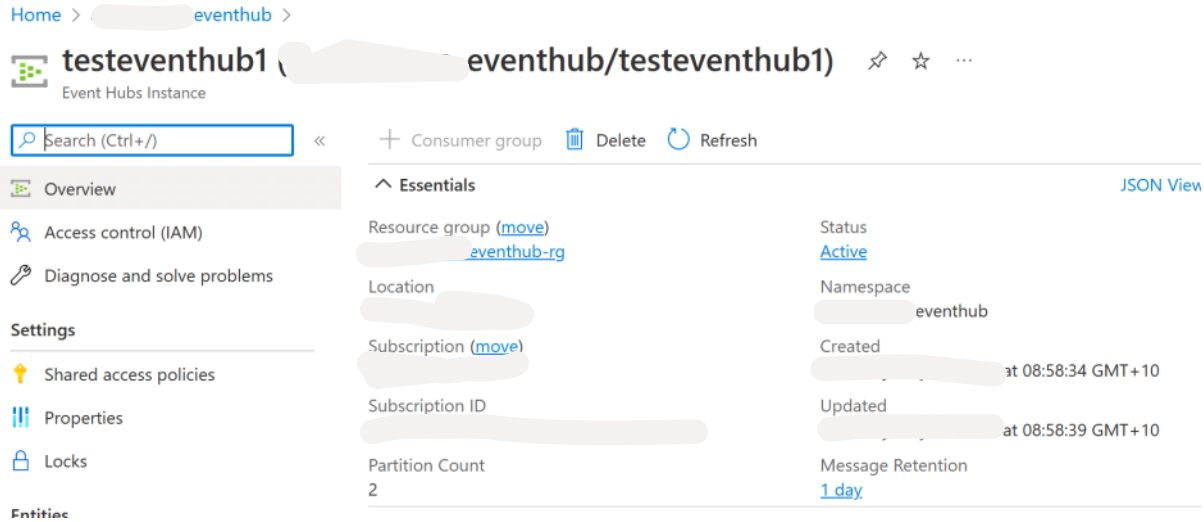
run as admin
C:\ProgramData\chocolatey\lib\ServiceBusExplorer\tools\ServiceBusExplorer.exe

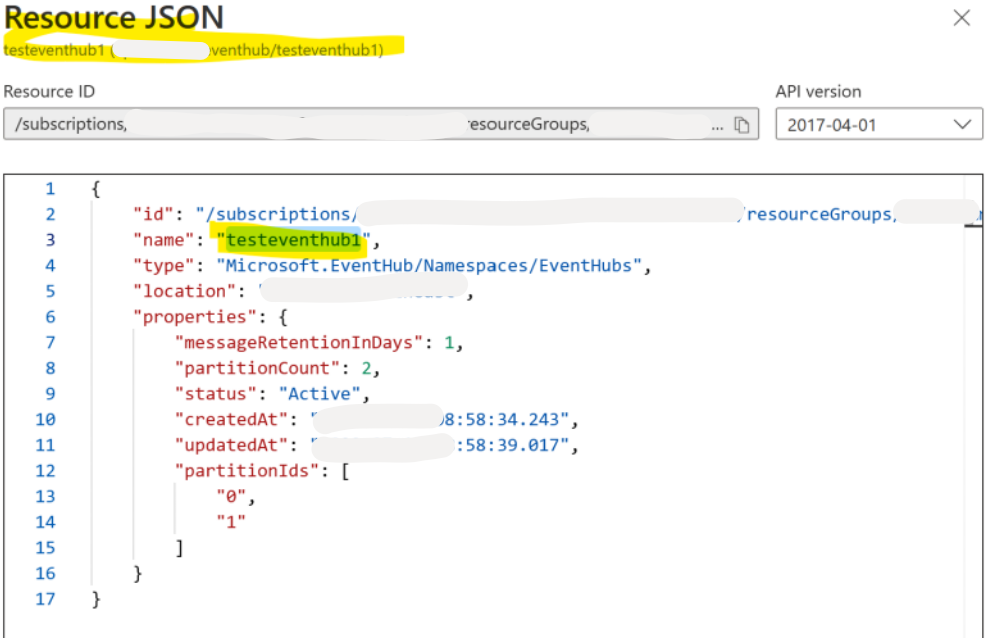
Create Event Hub

Let's create an event hub in the event hub namespace









Microsoft Documentation

read

https://docs.microsoft.com/en-us/azure/event-hubs/
https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-about

Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Data sent to an event hub can be transformed and stored by using any real-time analytics provider or batching/storage adapters.

The following scenarios are some of the scenarios where you can use Event Hubs:

- Anomaly detection (fraud/outliers)
- Application logging
- Analytics pipelines, such as clickstreams
- Live dashboards
- Archiving data
- Transaction processing
- User telemetry processing
- Device telemetry streaming

Capture event data

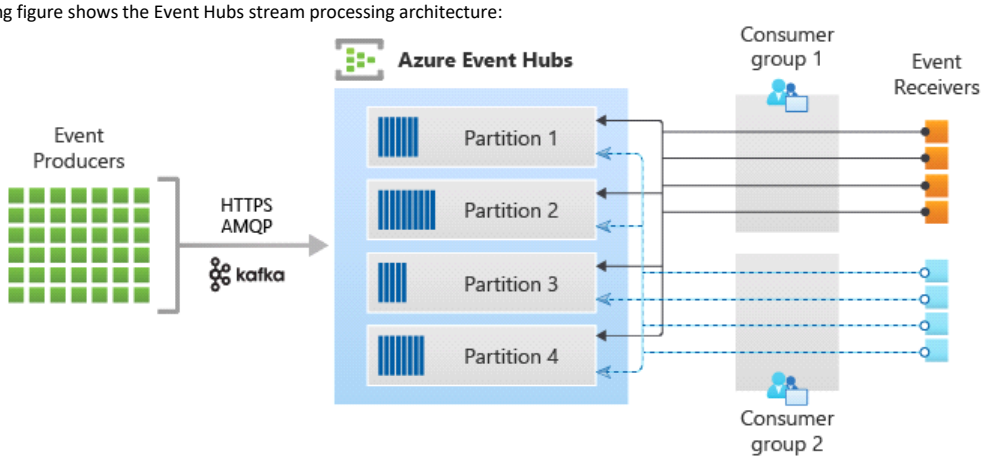
Capture your data in near-real time in an Azure Blob storage or Azure Data Lake Storagefor long-term retention or micro-batch processing. You can achieve this behavior on the same stream you use for deriving real-time analytics. Setting up capture of event data is fast. There are no administrative costs to run it, and it scales automatically with Event Hubs throughput units or processing units. Event Hubs enables you to focus on data processing rather than on data capture.

Key architecture components

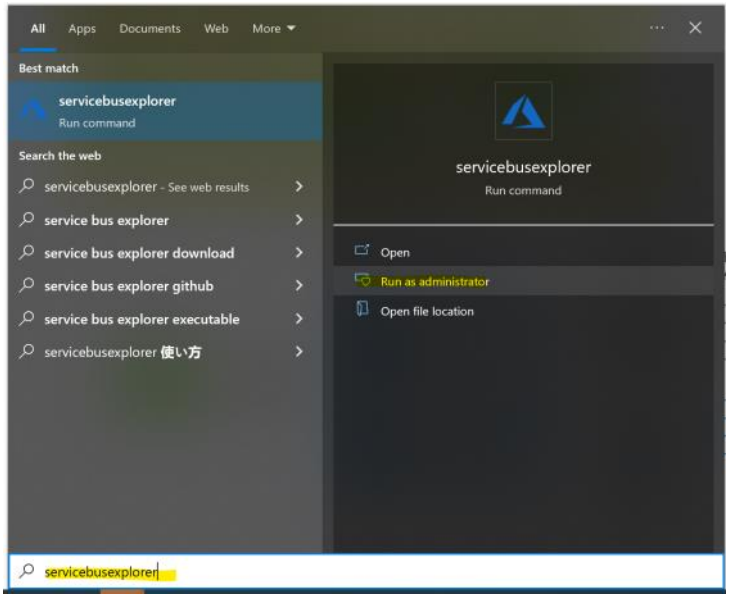
Event Hubs contains the following key components:

- **Event producers**: Any entity that sends data to an event hub. Event publishers can publish events using HTTPS or AMQP 1.0 or Apache Kafka (1.0 and above)
- **Partitions**: Each consumer only reads a specific subset, or partition, of the message stream.
- **Consumer groups**: A view (state, position, or offset) of an entire event hub. Consumer groups enable consuming applications to each have a separate view of the event stream. They read the stream independently at their own pace and with their own offsets.
- Throughput units (standard tier) or processing units (premium tier) or capacity units (dedicated) : Pre-purchased units of capacity that control the throughput capacity of Event Hubs.
- **Event receivers**: Any entity that reads event data from an event hub. All Event Hubs consumers connect via the AMQP 1.0 session. The Event Hubs service delivers events through a session as they become available. All Kafka consumers connect via the Kafka protocol 1.0 and later.

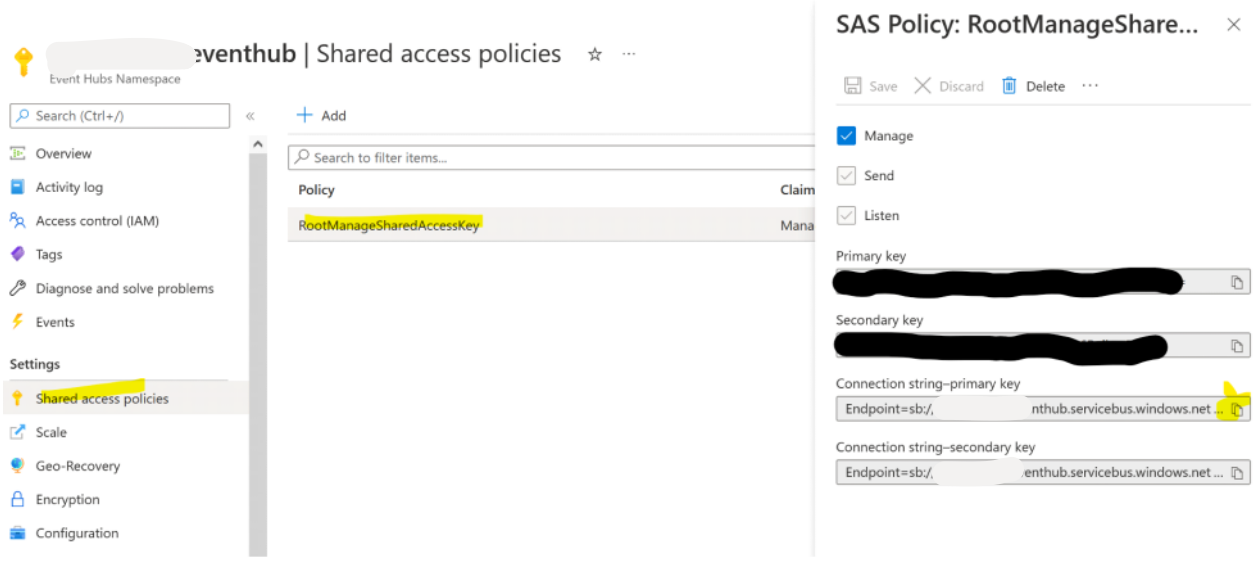
The following figure shows the Event Hubs stream processing architecture:



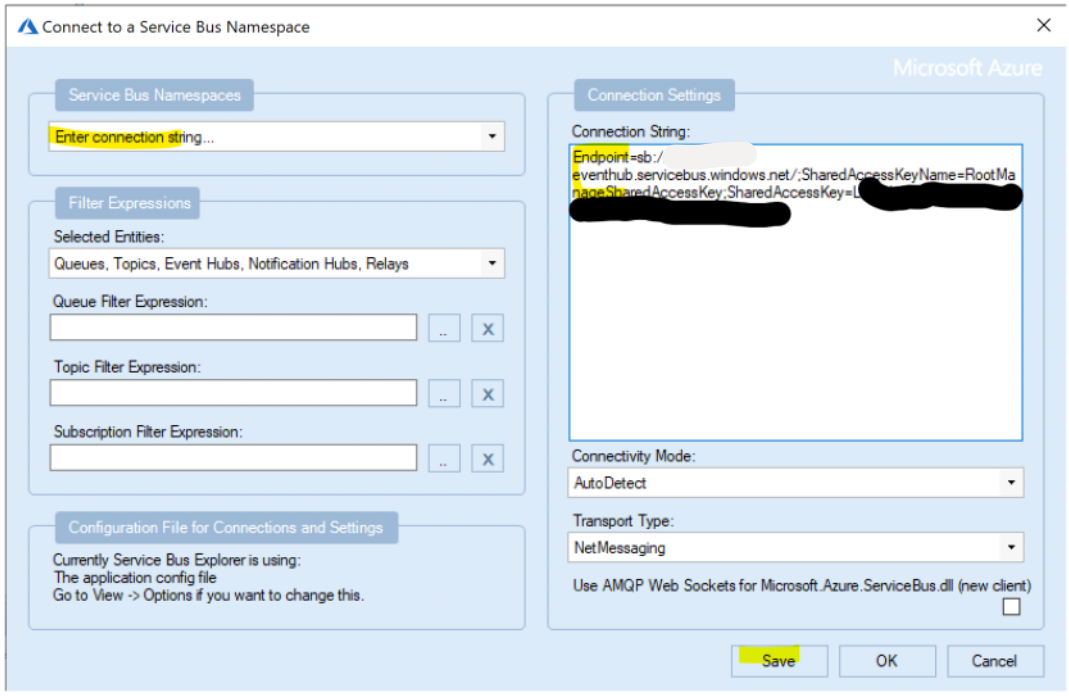
Connect to the blank event hub with ServiceBusExplorer



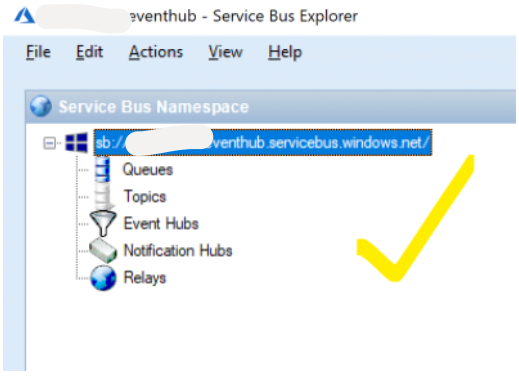
let's try the default connection string



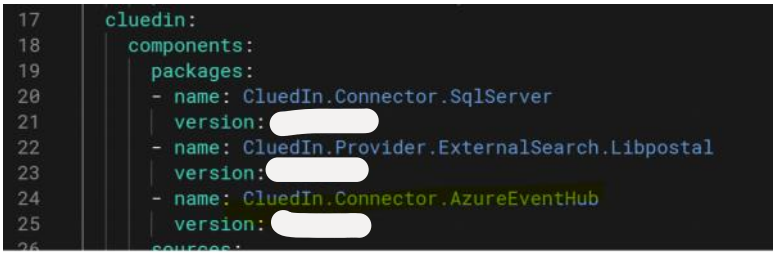
Endpoint=sb://-eventhub.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=LcWsbG<redacted>Gx3r9VFUc=



Success! - it is blank



Ensure The CluedIb.Connector.AzureEventHub is installed



```
cluedin:
  components:
    packages:
    - name: CluedIn.Connector.SqlServer
      version:
    - name: CluedIn.Provider.ExternalSearch.Libpostal
      version:
    - name: CluedIn.Connector.AzureEventHub
      version:
```

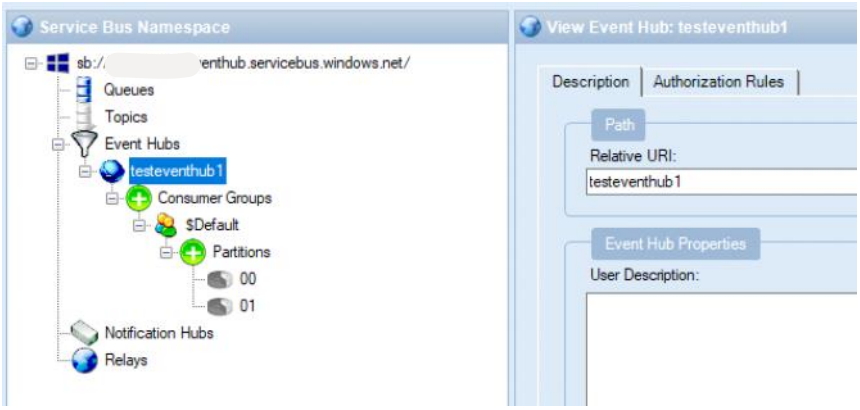
Connection String

Endpoint=sb://-eventhub.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=L<redacted>Gx3r9VFUc=

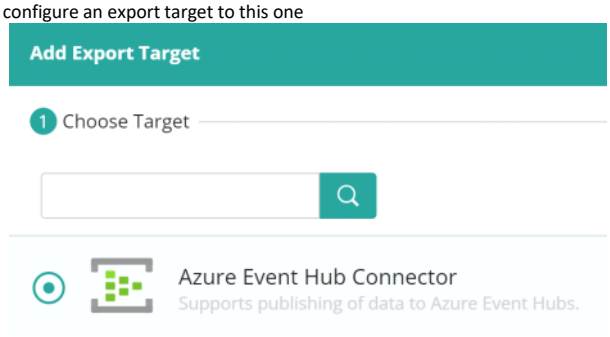
Event Hub Name

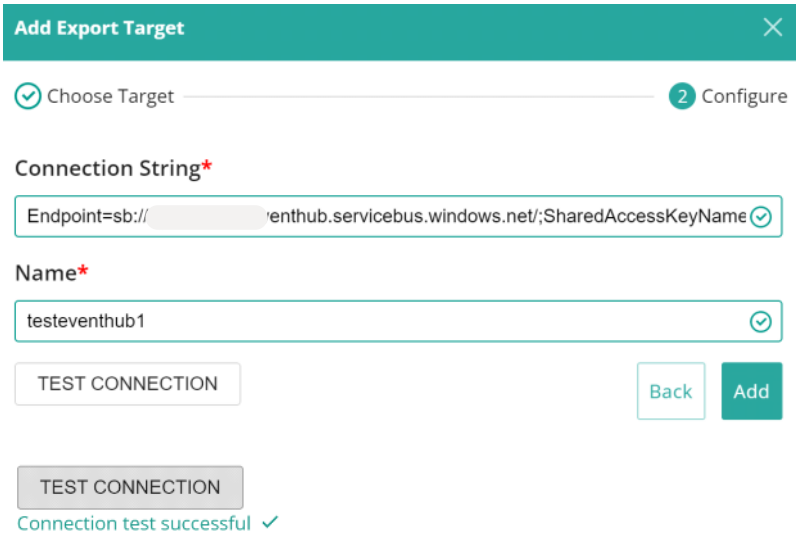
`testeventhub1`

we can see this in the sbexplorer

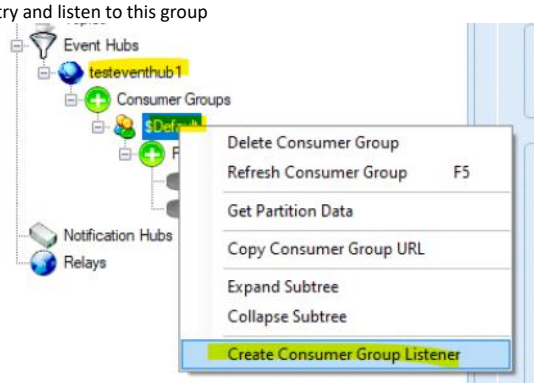


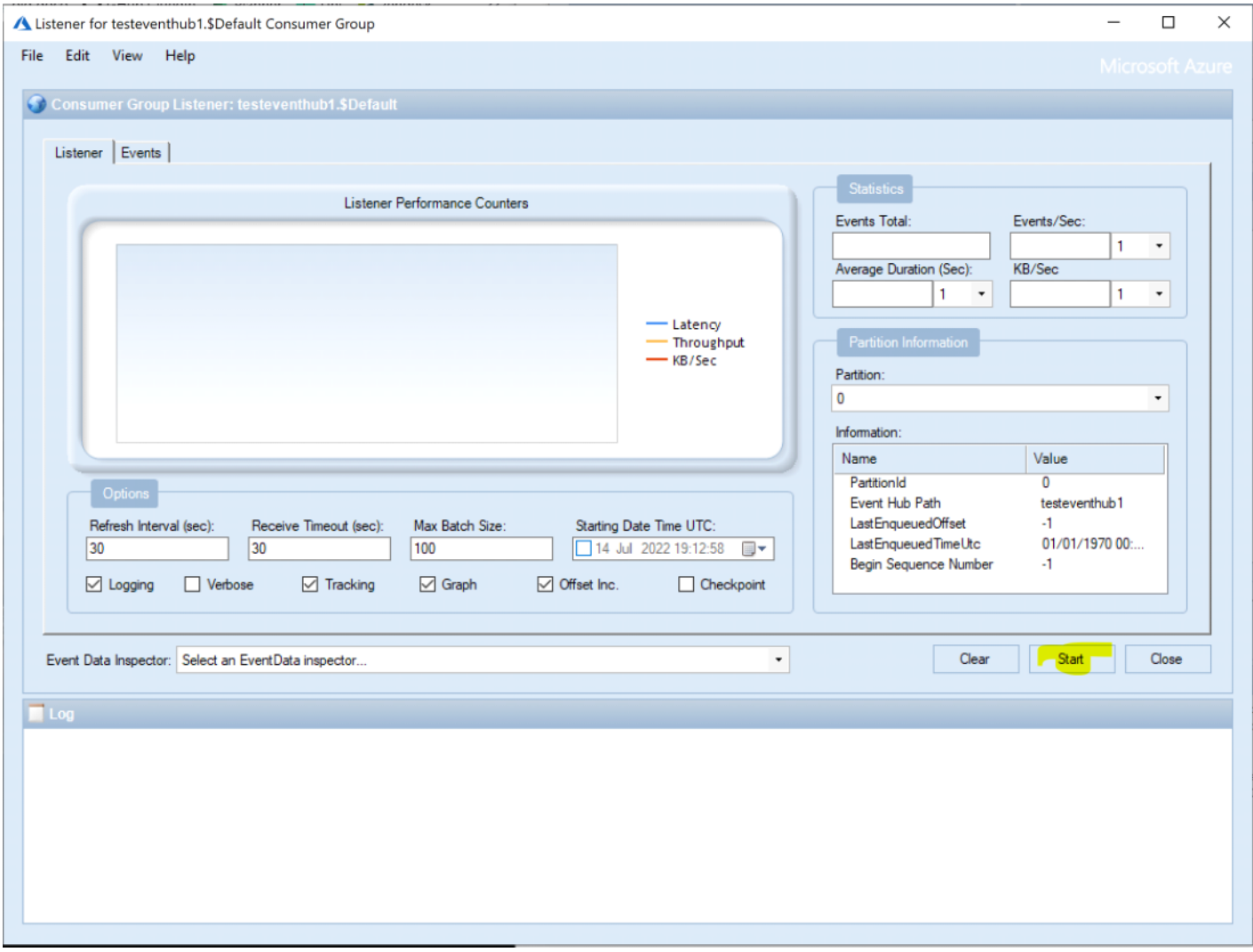
Let's configure an export target to this one



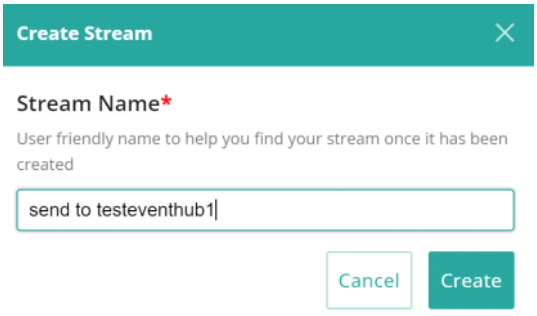


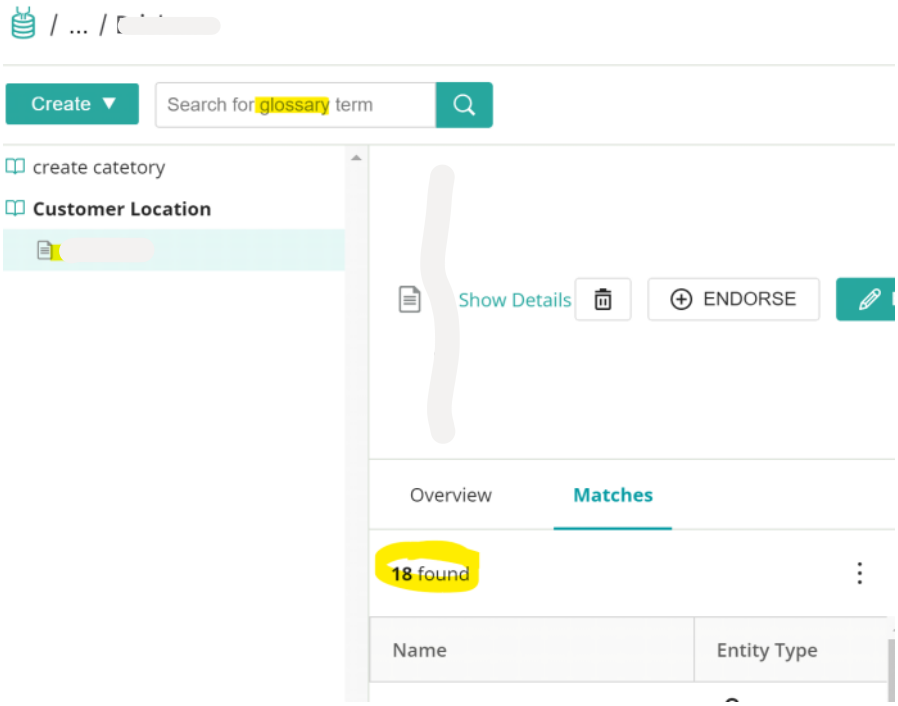
let's try and listen to this group

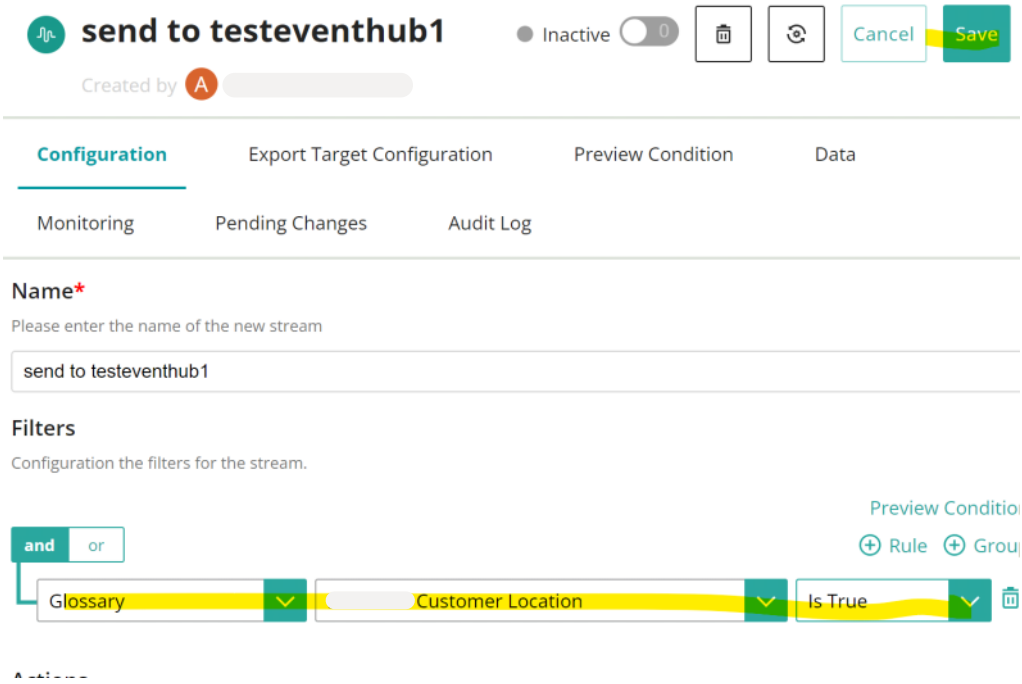




Create test stream - using glossary term as the filter




Create
Search for glossary term
create catetory
Customer Location
Show Details
ENDORSE
Overview
Matches
18 found
Name
Entity Type


send to testeventhub1
Inactive
Cancel
Save
Created by
Configuration
Export Target Configuration
Preview Condition
Data
Monitoring
Pending Changes
Audit Log
Name*
Please enter the name of the new stream
send to testeventhub1
Filters
Configuration the filters for the stream.
Preview Conditio
and
or
Rule
Grou
Glossary
Customer Location
Is True

Configuration | Export Target Configuration | **Preview Condition** | Data | Pending Changes | Audit Log

Attribute Type	Property Person Address Line 1	Property Person Addre
/Metadata/KeyValue	8 smallwood pl	NULL
/Metadata/KeyValue	8 smallwood pl	NULL
/Metadata/KeyValue	8 SMALLWOOD PL	NULL
/Metadata/KeyValue	8 Smallwood place	NULL
/Metadata/KeyValue	smallwood place	NULL
/Metadata/KeyValue	8 smallwood pl	NULL
/Metadata/KeyValue	8 smallwood pl	NULL
/Metadata/KeyValue	8 smallwood Place	NULL
/Metadata/KeyValue	smallwood place	NULL
/Metadata/KeyValue	8 smallwood place	NULL

Target name*

The name of the target container for your exported data (eg: for SQL, this would be the r

customer_location

Streaming Mode*

Export targets has the option to support multiple streaming modes. The streaming mod
the data should be exported into the target. A synchronized streaming mode, if support
between the data stored in CluedIn and the target. An event log streaming mode, on the
events (delete, update, create) each time an action is happening in CluedIn.

This export target only supports Sync mode

Export Edges*

Export edges gives you the necessary information to link correctly your data once it has l

0 Outgoing

0 Incoming

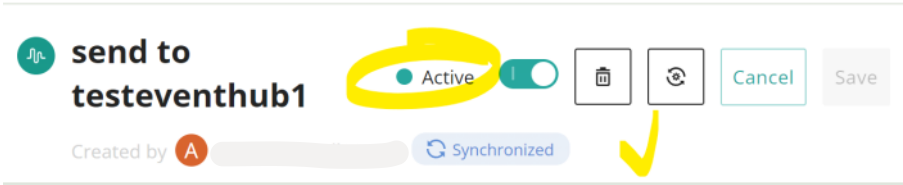
Properties to export

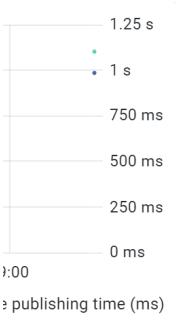
Default identify information for an entity is always exported. Add properties if you want

Add Property

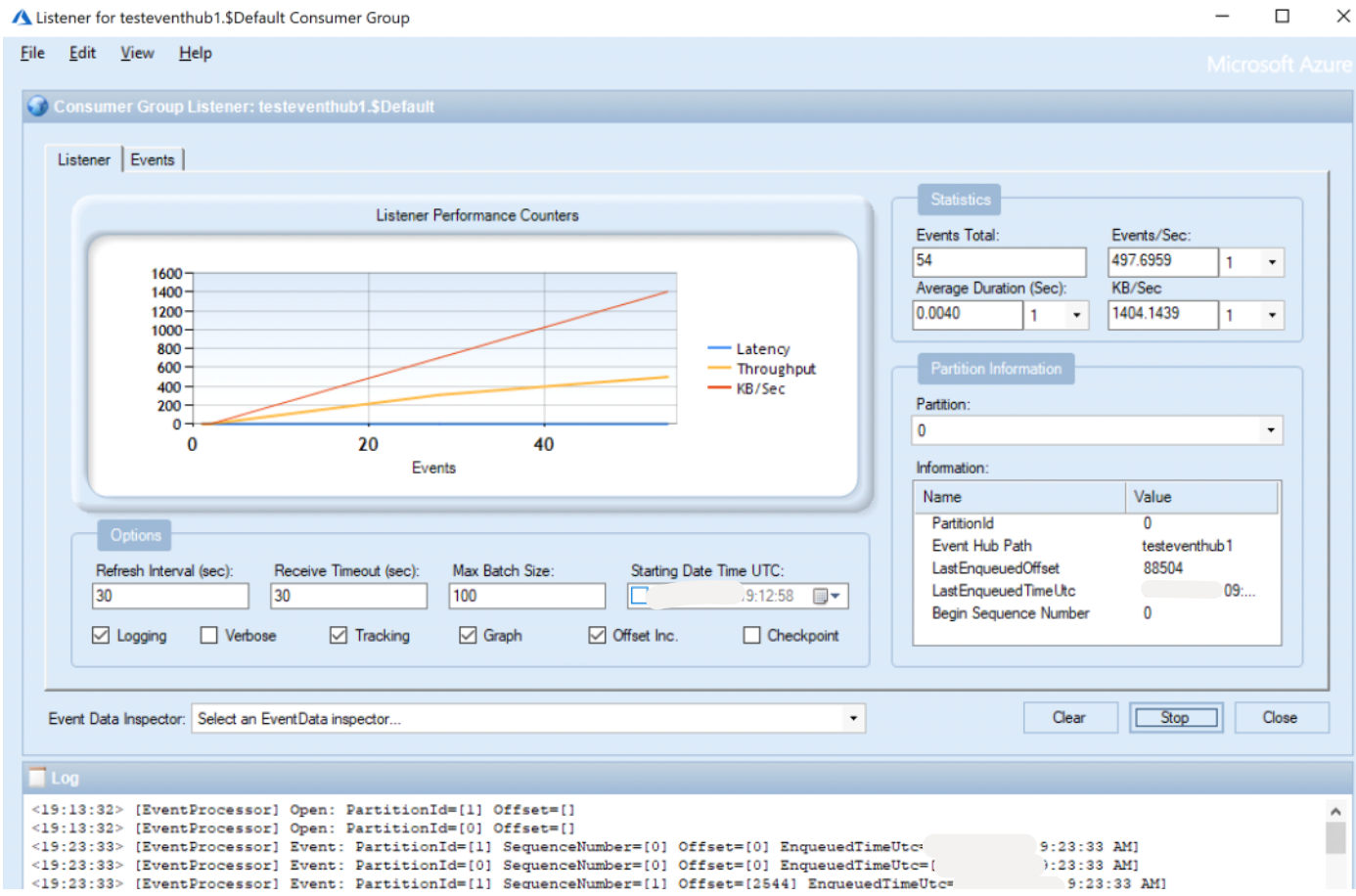
Property	Type
CreatedDate	entity
DisplayName	entity
person.sourceId	vocabulary
person.addressLine1	vocabulary

make the stream active such that events will start to stream

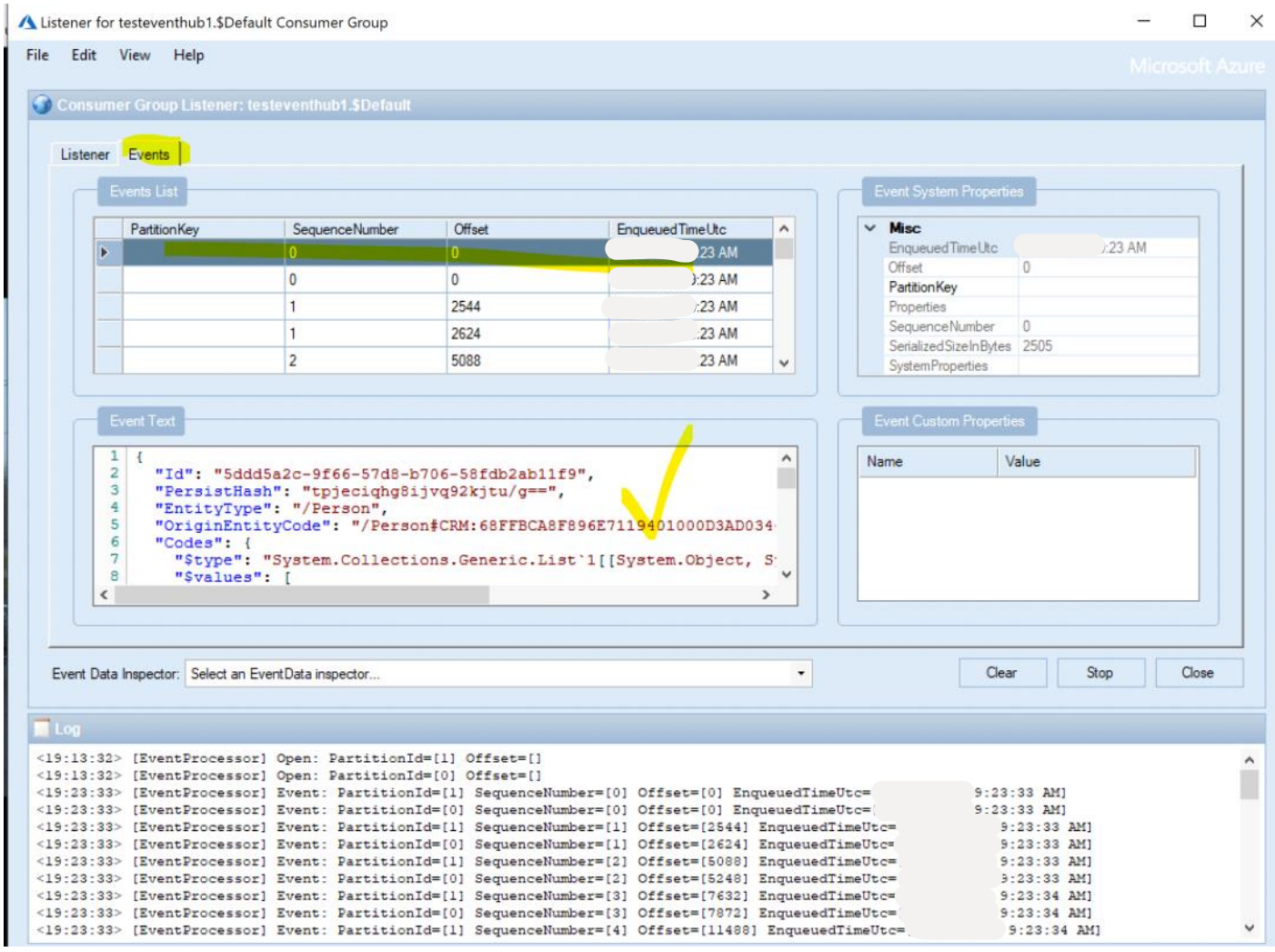




back to sbexplorer



```
<19:13:32> [EventProcessor] Open: PartitionId=[1] Offset=[]
<19:13:32> [EventProcessor] Open: PartitionId=[0] Offset=[]
<19:23:33> [EventProcessor] Event: PartitionId=[1] SequenceNumber=[0] Offset=[0] EnqueuedTimeUtc=                9:23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[0] SequenceNumber=[0] Offset=[0] EnqueuedTimeUtc=[               ):23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[1] SequenceNumber=[1] Offset=[2544] EnqueuedTimeUtc=              9:23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[0] SequenceNumber=[1] Offset=[2624] EnqueuedTimeUtc=[             9:23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[1] SequenceNumber=[2] Offset=[5088] EnqueuedTimeUtc=              9:23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[0] SequenceNumber=[2] Offset=[5248] EnqueuedTimeUtc=[             9:23:33 AM]
<19:23:33> [EventProcessor] Event: PartitionId=[1] SequenceNumber=[3] Offset=[7632] EnqueuedTimeUtc=              9:23:34 AM]
```

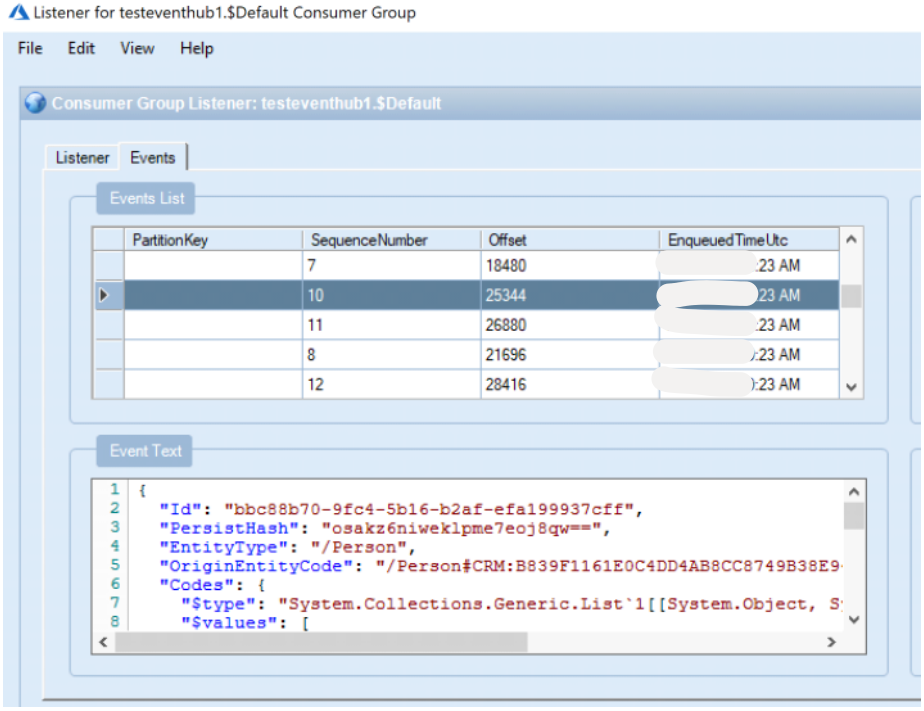


Example of the event text


```
{
  "Id": "5ddd5a2c-9f66-57d8-b706-58fdb2ab11f9",
  "PersistHash": "tpjeciqhg8ijvq92kjtu/g==",
  "EntityType": "/Person
",
  "OriginEntityCode": "/Person#CRM:68FFBCA8F896E7119401000D3AD0344E",
  "Codes": {
    "$type": "System.Collections.Generic.List`1[[System.Object,
System.Private.CoreLib]], System.Private.CoreLib",
    "$values": [
      "/Person#CluedIn(email):<redacted>",
      "/Person#CluedIn(hash-
sha1):321e83a280d5d684b1ec8a326a347c68512b636f",
      "/Person#CluedIn(hash-
sha1):58f93a833b6ebe59a70a42cc4be1632ca125e439",
      "/Person#CluedIn(hash-
sha1):808b401a17a8797e8bcfbfb329c44c490317b73a",
      "/Person#CluedIn(hash-
sha1):f40efacf0a1973282552b6be15a88f4565ae333b",
      "/Person#CluedIn(hash-
sha1):fc15ceaf65af9302039dff035f4b9acdaa9351f5",
      "/Person#CluedIn(mobilenumber):+<redacted>",

"/Person#CluedInImporter(dataset-8803377A86AE4801809D9DFEA6C4D391):110006
705",

"/Person#CluedInImporter(dataset-8803377A86AE4801809D9DFEA6C4D391):110213
613",

"/Person#CluedInImporter(dataset-8803377A86AE4801809D9DFEA6C4D391):65244"
,

"/Person#CluedInImporter(dataset-8803377A86AE4801809D9DFEA6C4D391):68FFBC
A8F896E7119401000D3AD0344E",
      "/Person#CluedInImporter(dataset-
```

```
C38C41C00B514A5AA339D3E31E13C313):68FFBCA8F896E7119401000D3AD0344E",
      "/Person#CluedInImporter(dataset-
DB5A90D46C77479991D1BBD122C7F5A3):68FFBCA8F896E7119401000D3AD0344E",

"/Person#CluedInImporter(datasource-3):68FFBCA8F896E7119401000D3AD0344E",
      "/Person#CluedInImporter(datasource-5):110006705",
      "/Person#CluedInImporter(datasource-5):110213613",
      "/Person#CluedInImporter(datasource-5):65244",

"/Person#CluedInImporter(datasource-5):68FFBCA8F896E7119401000D3AD0344E",

"/Person#CluedInImporter(datasource-7):68FFBCA8F896E7119401000D3AD0344E",
      "/Person#CluedInImporter(datasourcegroup-3):110006705",
      "/Person#CluedInImporter(datasourcegroup-3):110213613",
      "/Person#CluedInImporter(datasourcegroup-3):65244",

"/Person#CluedInImporter(datasourcegroup-3):68FFBCA8F896E7119401000D3AD03
44E",
      "/Person#CRM:68FFBCA8F896E7119401000D3AD0344E",
      "/Person#File Data Source:110006705",
      "/Person#File Data Source:110213613",
      "/Person#File Data Source:65244",
      "/Person#File Data Source:68FFBCA8F896E7119401000D3AD0344E",
      "/Person#Global:68FFBCA8F896E7119401000D3AD0344E",
      "/Person#Jasper:110006705",
      "/Person#Jasper:110213613",
      "/Person#Jasper:65244",
      "/Person#Jasper:68FFBCA8F896E7119401000D3AD0344E",
      "/Person#Loyalty:68FFBCA8F896E7119401000D3AD0344E",
      "/Person#Loyalty:B000738417"
    ]
  },
  "CreatedDate": "<redacted>T10:13:38+00:00",
  "DisplayName": null,
  "personsourceId": "110213613",
  "personaddressLine1": "8 smallwood pl"
}
```


spot audit



num of records looks good