EventGrid

User Guide

Issue 01

Date 2025-08-18

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Getting Started with EG

EventGrid (EG) is a serverless event bus service for standard and centralized access of Huawei Cloud services and custom or SaaS applications. You can build a loosely coupled, distributed event-driven architecture to flexibly route events via CloudEvents.

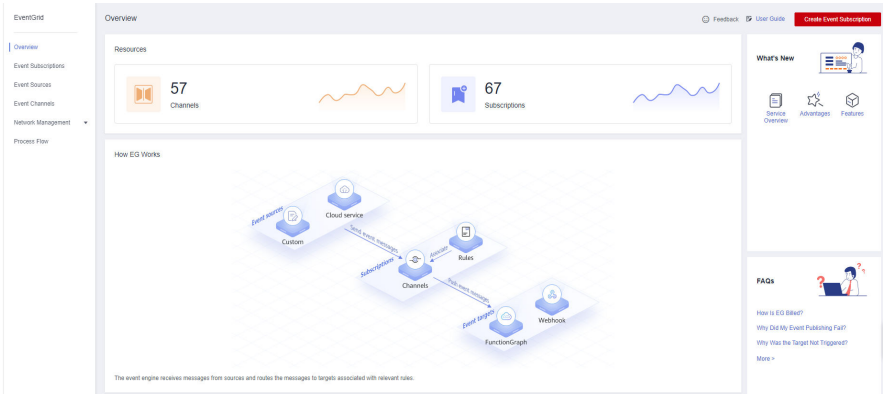
Prerequisites

1. You have **registered a HUAWEI ID and enabled Huawei Cloud services**.
2. Your account has permission to use EG. For details about how to authorize an account and bind permissions to it, see **Creating a User and Granting EG Permissions**.

Logging In to the EG Console

Step 1 Log in to the **EG console**.

Figure 1-1 EG console



----End

2 Permissions Management

2.1 Creating a User and Granting EG Permissions

This section describes how to use **Identity and Access Management (IAM)** to implement fine-grained permissions control for your EG resources. With IAM, you can:

- Create IAM users for employees based on the organizational structure of your enterprise. Each IAM user has their own security credentials to access EG resources.
- Grant only the permissions required for users to perform a specific task.
- Entrust an account or cloud service to perform professional and efficient O&M on your EG resources.

If your account does not require individual IAM users, skip this section.

This section describes the procedure for granting permissions (see **Figure 2-1**).

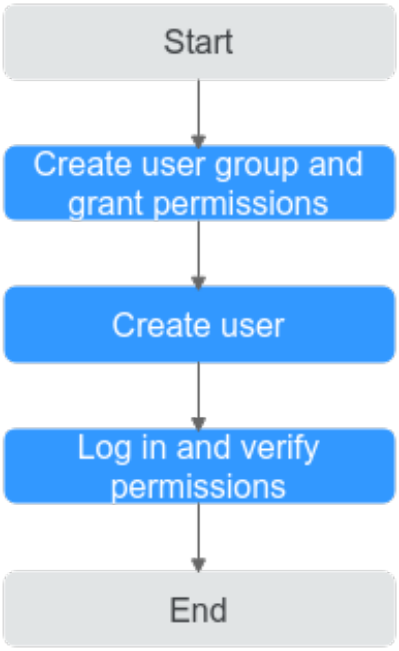
Prerequisites

Learn about the permissions (see **System-defined roles and policies supported by EG**) supported by EG and choose policies according to your requirements.

For the permissions of other services, see **System Permissions**.

Process Flow

Figure 2-1 Process for granting EG permissions



1. **Create a user group and assign permissions**.

   Create a user group on the IAM console, and assign it the read-only permissions for EG.

2. **Create an IAM user and add them to the user group**.

   Create a user on the IAM console and add the user to the group created in **Step 1**.

3. **Log in** and verify permissions.

   Log in to the **EG console** as the created user and verify the read-only permissions for EG.

2.2 Custom Policies

Custom policies can be created to supplement the system-defined policies of EG.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Edit policies from scratch or based on an existing policy in JSON format.

For details, see **Creating a Custom Policy**. The following section contains examples of common EG custom policies.

Example Custom Policies

- Example 1: Allow user to delete event sources

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "
                eg:sources:delete
                eg:sources:list
             "
         ]
      }
   ]
}
```

- Example 2: Deny event source deletion

  A policy with only "Deny" permissions must be used in conjunction with other policies to take effect. If the permissions assigned to a user include both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

  The following method can be used if you need to assign permissions of the **EG FullAccess** policy to a user but also forbid the user from deleting event sources. Create a custom policy to disallow event source deletion and assign both policies to the group the user belongs to. Then the user can perform all operations on EG except deleting event sources. The following is an example of a deny policy:

```
{
      "Version": "1.1",
      "Statement": [
            {
                  "Effect": "Deny",
                  "Action": [
                        "eg:sources:delete"
                  ]
            }
      ]
}
```


3 Event Sources

3.1 Introduction

Event sources include Huawei Cloud services, custom applications, and SaaS applications. They produce events and publish them to EG.

EG supports the following event sources:

- Cloud service: Huawei Cloud services publish specific types of events to EG through predefined channels. The events are filtered with rules and then routed to targets. For details about the supported cloud service event sources, see **Cloud Service Event Sources**.
- Custom
  - Custom applications publish events to EG through custom channels. The events are filtered with rules and then routed to targets.
  - Custom event sources include DMS for RabbitMQ and DMS for RocketMQ.

CAUTION

EG does not encrypt the information in event sources. If your events contain sensitive information, encrypt it for security.

3.2 Cloud Service Event Sources

This section describes the cloud service event sources supported by EG, and depicts how to view their predefined event types.

Constraints

Currently, only write events are supported. Read events are not supported.

Cloud Service Event Source List

The following table lists the cloud service event sources supported by EG.

Table 3-1 Cloud service event sources

Cloud Application Engine (CAE)	Database and Application Migration (UGO)	Classroom	Content Moderation
Virtual Private Cloud (VPC)	CodeCheck	GaussDB NoSQL	API Gateway (APIG)
Data Warehouse Service (DWS)	CloudDeploy	Identity and Access Management (IAM)	EventGrid (EG)
Ubiquitous Cloud Native Service (UCS)	Scalable File Service (SFS)	CloudIDE	Face Recognition Service (FRS)
Cloud Service Engine (CSE)	Direct Connect	Data Lake Visualization (DLV)	NAT Gateway
Workspace	IoT Device Access (IoTDA)	Distributed Message Service (DMS)	Knowledge Graph (KG)
IoT Edge	Log Tank Service (LTS)	CloudBuild	Object Storage Migration Service (OMS)
Cloud Backup and Recovery (CBR)	Message & SMS (MSGSMS)	Elastic IP (EIP)	Cloud Trace Service (CTS)
Cloud Search Service (CSS)	Video Analysis Service (VAS)	Data Admin Service (DAS)	Bare Metal Server (BMS)
CloudTest	VPC Endpoint (VPCEP)	Cloud Storage Gateway (CSG)	Virtual Private Network (VPN)
Enterprise Router (ER)	Recommender System (RES)	Cloud Server Backup Service (CSBS)	Content Delivery Network (CDN)
Container Guard Service (CGS)	Situation Awareness (SA)	CodeHub	CloudTable
Volume Backup Service (VBS)	CloudSite	Cloud Phone (CPH)	Cloud Performance Test Service (CPTS)
Intelligent EdgeCloud (IEC)	FunctionGraph	Server Migration Service (SMS)	Tag Management Service (TMS)

Conversational Bot Service (CBS)	Relational Database Service (RDS)	Domain Name Service (DNS, Region)	Storage Disaster Recovery Service (SDRS)
Voice Call	Application Performance Management (APM)	Application Orchestration Service (AOS)	Data Ingestion Service (DIS)
Database Security Service (DBSS)	HiLens	Cloud Data Migration (CDM)	Multi-Site High Availability Service (MAS)
CloudPipeline	Image Recognition	OBS Application Service	Object Storage Service (OBS)
Intelligent EdgeFabric (IEF)	SoftWare Repository for Container (SWR)	Distributed Cache Service (DCS)	Auto Scaling (AS)
Vulnerability Scan Service (VSS)	Graph Engine Service (GES)	Data Lake Insight (DLI)	Cloud Container Instance (CCI)
CodeArts Req	Document Database Service (DDS)	Data Replication Service (DRS)	ModelArts
Distributed Database Middleware (DDM)	Simple Message Notification (SMN)	ServiceStage	CodeArts
Blockchain Service (BCS)	Application Operations Management (AOM)	MapReduce Service (MRS)	Cloud Bastion Host (CBH)
Host Security Service (HSS)	Web Application Firewall (WAF)	Elastic Load Balance (ELB)	Elastic Volume Service (EVS)
ROMA Connect	Cloud Container Engine (CCE)	Image Management Service (IMS)	Elastic Cloud Server (ECS)

Viewing Event Types

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Sources**.

Step 3 On the **Cloud Service** tab, click the desired event source.

Step 4 View the event types and description in the **Event Types** area.

----End

3.3 Creating an Event Source

3.3.1 Custom Application

Create a custom application event source.

Prerequisites

(Optional) You have **created an event channel**.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Sources**.

Step 3 Click **Create Event Source**.

Step 4 Set event source information by referring to **Table 3-2**.

Table 3-2 Custom application event source parameters

Parameter	Description
Provider	The default value is **Custom**.
Type	Select **Custom application**.
Name	Event source name. The name cannot be modified once the event source is created.
Description	Describe the event source.

Step 5 Click **OK**.

View this event source on the **Custom** tab.

NOTE

- Only the event source description can be modified. To modify it, click **Edit** in the row that contains the desired event source.
- To view details about a custom event source, click its name in the custom event source list.
- If the event source is new (unavailable in the event source list), the monitoring information cannot be queried on the Cloud Eye console after the event delivery.

----End

Follow-Up Procedure

(Optional) **Creating an Event Subscription**

3.3.2 DMS for RabbitMQ

Create a DMS for RabbitMQ event source.

DMS for RabbitMQ is supported in these regions: CN East-Shanghai1, CN East-Shanghai2, CN North-Beijing4, CN North-Ulanqab1, and CN South-Guangzhou.

Prerequisites

- (Optional) You have **created an event channel**.
- You have purchased a DMS for RabbitMQ instance. The instance contains queues and is in the **Running** state. For details, see **Buying an Instance**.
- You have **created a private endpoint** with the same VPC and subnet as the RabbitMQ instance.
- You have configured the **default** security group with rules for the RabbitMQ instance. For details, see **How Do I Configure a Security Group When Creating an Event Source?**

Creating a RabbitMQ Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Bus** > **Event Sources**.

Step 3 Click **Create Event Source**.

Step 4 Set event source information by referring to **Table 3-3**.

Table 3-3 RabbitMQ event source parameters

Parameter	Description
Provider	The default value is **Custom**.
Type	Select **DMS for RabbitMQ**. **NOTE** You will be prompted to create an agency when creating your first DMS for RabbitMQ event source. For details, see **Authorization**.
Name	Event source name. The name cannot be modified once the event source is created.
Description	Describe the event source.
Channel	Select an existing custom event channel or click **Create Event Channel** to create one. The channel cannot be modified once the event source is created.
Instance	Select a RabbitMQ instance.
Username	Username of the RabbitMQ instance.
Password	Password of the RabbitMQ instance.

Parameter	Description
Vhost	Virtual host of the RabbitMQ instance.
Queue	Queue in the RabbitMQ instance.

Step 5 Click **OK**.

View this event source on the **Custom** tab.

NOTE

- Only the event source description can be modified. To modify it, click **Edit** in the row that contains the desired event source.
- To view details about a custom event source, click its name in the custom event source list.

----End

Viewing the Event Format

Prerequisites

1. A RabbitMQ instance has been created.
2. You have created an endpoint in the same VPC and subnet as the RabbitMQ instance.

Step 1 Create an event channel.

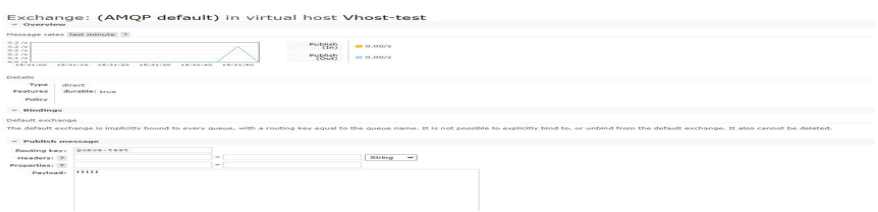
Step 2 Create a RabbitMQ event source.

Step 3 Create an event subscription whose source is RabbitMQ and target is FunctionGraph.

Step 4 Send an event.

Obtain RabbitMQ information, allow port **15671** in the security group, and access the RabbitMQ web UI by entering the public access URL in your browser.

Figure 3-1 Logging in to RabbitMQ and sending events



Step 5 View the event message format.

1. View event trace details.
   a. In the navigation pane on the left, choose **Event Bus** > **Event Channels**. Click **View Events** or choose **More** > **View Events** on the right of the channel name.
   b. On the displayed page, click an event name in the **Event ID** column to view the event details.

      The event format within the message body is as follows:

      ```
      {
      "datacontenttype": "application/json",
      "data": {
      "context": "11111"
      },
      "subject": "RABBITMQ:cn-north-4:f003dc69-2fc3-4c44-9062-0b9a2c6cb8cc/
      0ef1e7a03280f3ed2f69c00c652a5744:RABBITMQ:source-rabbitmq",
      "specversion": "1.0",
      "id": "cd845ec7-0314-400d-9293-d39d7b258d9b",
      "source": "source-rabbitmq",
      "time": "2024-02-05T15:31:51Z",
      "type": "RABBITMQ:CloudTrace:RabbitmqCall"
      }
      ```

2. View the logs of the target function.
   a. Log in to the FunctionGraph console, choose **Functions** > **Function List** in the navigation pane, and click the name of the target function to go to the function details page.
   b. Choose **Monitoring** > **Logs** > **Request List** and click a request ID to view the log details.

----End

3.3.3 DMS for RocketMQ

Create a DMS for RocketMQ event source.

Prerequisites

- (Optional) You have **created an event channel**.
- You have purchased a DMS for RocketMQ instance. The instance contains topics and is in the **Running** state. For details, see **Buying a RocketMQ Instance**.
- You have **created a private endpoint** with the same VPC and subnet as the RocketMQ instance.
- You have configured the **default** security group with rules for the RocketMQ instance. For details, see **How Do I Configure a Security Group When Creating an Event Source?**

Creating a RocketMQ Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Sources**.

Step 3 Click **Create Event Source**.

Step 4 Set event source information by referring to **Table 3-4**.

Table 3-4 RocketMQ event source parameters

Parameter	Description
Provider	The default value is **Custom**.
Type	Select **DMS for RocketMQ**. **NOTE** You will be prompted to create an agency when creating your first DMS for RocketMQ event source. For details, see **Authorization**.
Name	Event source name. The name cannot be modified once the event source is created.
Description	Describe the event source.
Channel	Select an existing custom event channel or click **Create Event Channel** to create one. The channel cannot be modified once the event source is created.
Instance	Select a RocketMQ instance. **Self-hosted RocketMQ** indicates your own RocketMQ.
Topic	Topic of the RocketMQ instance.
Consumer Group	Consumer group of the RocketMQ instance.
Username	Required if ACL has been enabled for the RocketMQ instance.
Secret Key	Required if ACL has been enabled for the RocketMQ instance.
VPC	Available only when you selected **Self-hosted RocketMQ** for **Instance**.
Subnet	Available only when you selected **Self-hosted RocketMQ** for **Instance**.
Connection Address	Available only when you selected **Self-hosted RocketMQ** for **Instance**. Enter the connection address of your own RocketMQ.
SSL	Available only when you selected **Self-hosted RocketMQ** for **Instance**. Specify whether to enable SSL. **SSL cannot be modified if your RocketMQ is running. If SSL is changed, delete the event source and create a new one.**
ACL	Available only when you selected **Self-hosted RocketMQ** for **Instance**. Specify whether to enable ACL.

Step 5 Click **OK**.

View this event source on the **Custom** tab.

NOTE

- Only the event source description can be modified. To modify it, click **Edit** in the row that contains the desired event source.
- To view details about a custom event source, click its name in the custom event source list.

----End

Viewing the Event Format

Prerequisites

1. You have created a RocketMQ instance.
2. You have created an endpoint in the same VPC and subnet as the RocketMQ instance.

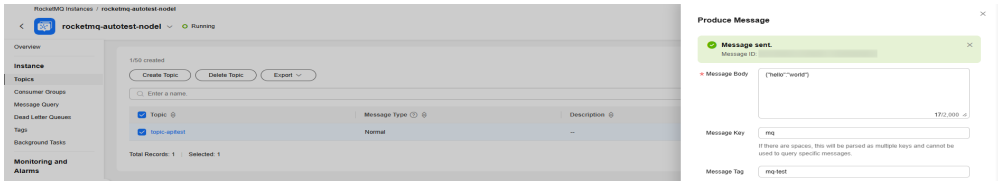
Step 1 Create an event channel.

Step 2 Creating a RocketMQ Event Source

Step 3 Create an event subscription whose source is RocketMQ and target is FunctionGraph.

Step 4 Send an event.

Figure 3-2 Sending an event



Step 5 View the event message format.

1. View event trace details:

   The event format within the message body is as follows:

```
{
  "datacontenttype": "application/json",
  "data": {
    "context": "{\"hello\":\"world\"}",
    "topic": "topic-test"
  },
  "subject": "ROCKETMQ:cn-north-4:f003dc69-2fc3-4c44-9062-0b9a2c6cb8cc/
0ef1e7a03280f3ed2f69c00c652a5744:ROCKETMQ:source-rocketmq",
  "specversion": "1.0",
  "id": "e6cc599b-0664-4078-87dd-5630087d5f7e",
  "source": "source-rocketmq",
  "time": "2024-02-05T14:20:31Z",
  "type": "ROCKETMQ:CloudTrace:RocketmqCall"
}
```

2. View the logs of the target function.

   a. Log in to the FunctionGraph console, choose **Functions** > **Function List** in the navigation pane, and click the name of the target function to go to the function details page.

b. Choose **Monitoring** > **Logs** > **Request List** and click a request ID to view the log details.

----End

3.4 Deleting a Custom Event Source

Delete a custom event source that will no longer be used.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Sources**.

Step 3 On the **Custom** tab, click **Delete** in the row that contains the desired event source.

Step 4 Click **Yes**.

----End

4 Event Channels

4.1 Introduction

Event channels receive events from event sources.

EG supports the following event channels:

- Cloud service: A channel automatically created by EG to receive events from cloud services. This channel cannot be modified. **Events generated by cloud service event sources can only be published to this channel.**
- Custom: Channels you create to receive events from custom sources.

4.2 Creating an Event Channel

Create a custom event channel.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Channels**.

Step 3 Click **Create Event Channel**.

Step 4 Enter a channel name and description. Click **OK**. The following table describes the parameters.

Table 4-1 Parameters for creating a custom event channel

Parameter	Description
Channel	Enter a channel name.
Description	Describe the channel.

View this channel in the **Custom** area.

NOTE

- Only the event channel description can be modified. To modify it, click **Edit** in the row that contains the desired event channel.
- To view details about a custom event channel, click its name in the custom event channel list.

----End

4.3 Deleting a Custom Event Channel

Delete an event channel that will no longer be used.

Constraints

If the event channel to delete is associated with sources and subscriptions, disassociate it first.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Channels**.

Step 3 Click **Delete** in the row that contains the desired event channel.

Step 4 Click **Yes**.

----End

4.4 Publishing Events

Publish events to a channel.

By publishing events, check whether an event source, channel, and target have been connected, whether the configured rules are valid, and whether events can be sent to the target.

Prerequisites

- You have **created an event channel**.
- You have created an **application event source**.
- You have configured an event target and **created an event subscription** with the preceding resources.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Channels**.

Step 3 Click **Publish Event**.

Step 4 Configure the parameters described in the following table.

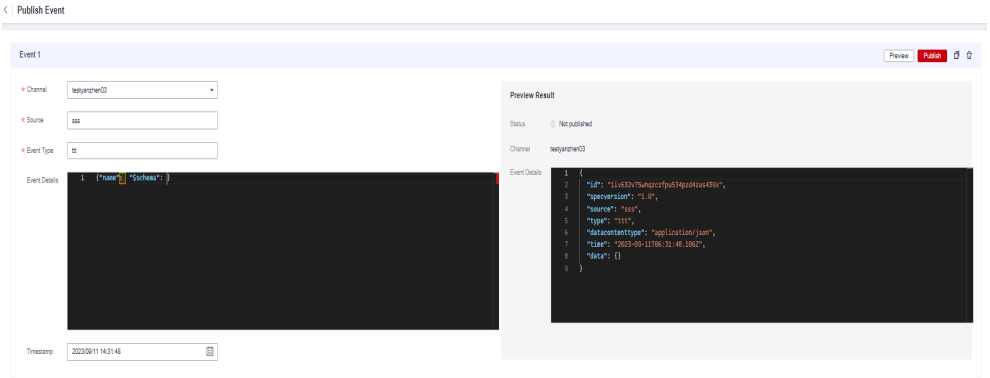
Table 4-2 Parameters for publishing events

Parameter	Description
Channel	Select a channel.
Source	Enter a custom application event source.
Event Type	Enter an event type.
Event Details	Enter event content in JSON format.
Timestamp	Select a timestamp.

Step 5 Click **Preview** to preview the event.

Step 6 Click **Publish**. If the event is successfully published, a result similar to that in **Figure 4-1** is displayed.

Figure 4-1 Publishing an event



NOTE

- To publish more events, click **Add Event**.
- You can publish one or more events at a time.
- To clone an event, click .
- To delete an event, click .
- Each event cannot exceed 64 KB.

----End

4.5 Viewing Event Traces

View traces of an event channel.

You can query sources, details, delivery targets, and delivery status of events in 72 hours.

Event traces are supported in these regions: CN East-Shanghai1, CN East-Shanghai2, CN North-Beijing4, CN North-Ulanqab1, and CN South-Guangzhou.

Constraints

Details about events that failed to be delivered can be queried in 72 hours, but details about successfully delivered events may be available in a longer period.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Channels**.

Step 3 Click **View Events**.

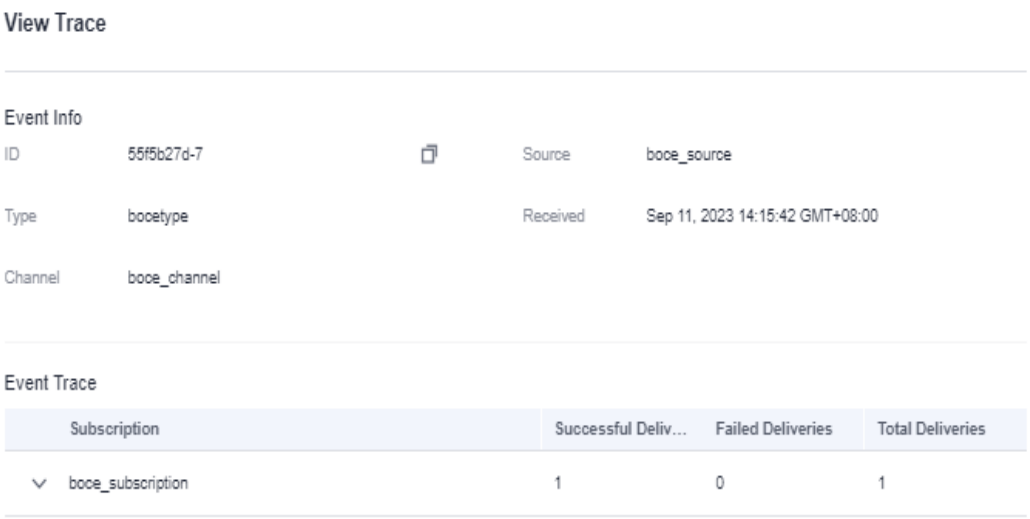
Step 4 Click the filter icon on the right to query the event.

Table 4-3 Filter parameters

Parameter	Description
Time Range	Select an event publishing period.
Filter By	You can select **Event source and type**, and **Delivery status and subscription name**.
Event ID	Enter an event ID.

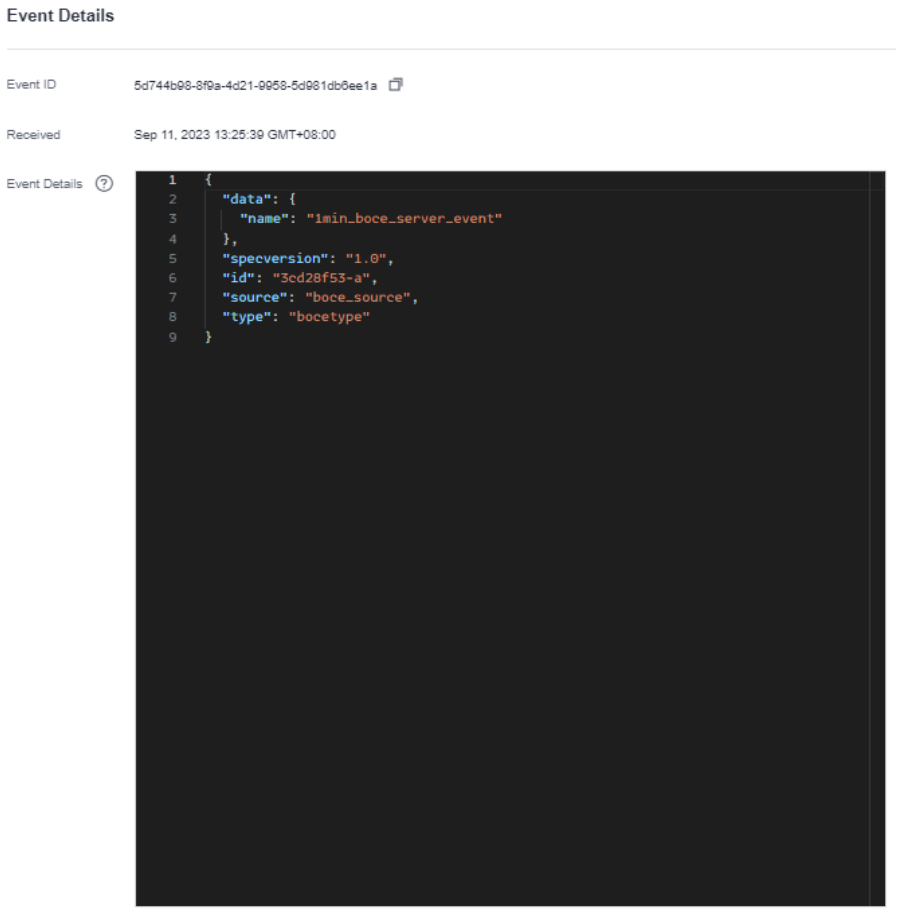
Step 5 Click **View Trace** to view event information, traces, and delivery details.

Figure 4-2 Viewing event traces



Step 6 Click an event ID to view event details, as shown in **Figure 4-3**.

Figure 4-3 Viewing event details



----End

4.6 Monitoring

4.6.1 Viewing Monitoring Data

Scenario

Cloud Eye monitors event channel metrics in real time. You can view these metrics on the Cloud Eye console.

Prerequisites

You have created an event channel.

Procedure

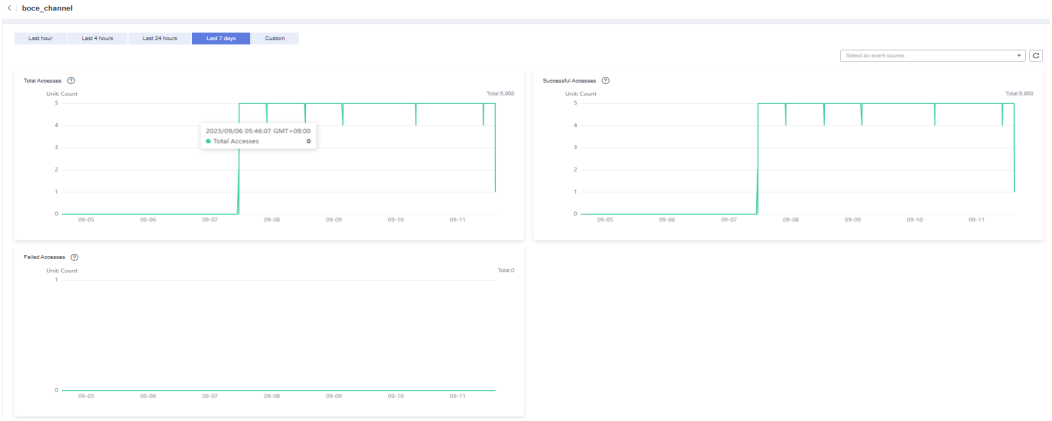
Step 1 Log in to the **EG console**.

Step 2 Choose **Event Channels**.

Step 3 Click  in the row that contains the target event channel to go to the monitoring page. Data of all accessed events in the last hour is displayed by default.

You can also click **1h**, **3h**, **12h**, **1d**, **7d**, or **30d** to view event accesses in different periods.

Figure 4-4 Viewing event channel monitoring data



NOTE

To customize a time range, click  .

If you enable **Auto Refresh**, the metric data is refreshed every 5 seconds.

Click **View details** to go to the Cloud Eye console.

If you set **Period** to **Raw data**, the raw monitoring data is displayed. If you set **Period** to a specific time, you can select different aggregation methods, including **Avg.**, **Max.**, **Min.**, **Sum**, and **Variance**.

----End

4.6.2 Supported Metrics

Introduction

This section describes the event channel metrics and dimensions reported to Cloud Eye. You can search metrics and alarms on the Cloud Eye console or on the monitoring page of EG.

Metrics

Table 4-4 Metric description

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
pub_num	Total Accesses	Number of times event access is attempted.	≥ 0	Event channel	1
pub_success_num	Successful Accesses	Number of times events are actually accessed.	≥ 0	Event channel	1
pub_success_rate	Success Rate	Percentage of total accesses that are successful.	0%–100%	Event channel	1
pub_failed_num	Failed Accesses	Number of times events could not be accessed.	≥ 0	Event channel	1
pub_failed_rate	Failure Rate	Percentage of total accesses that failed.	0%–100%	Event channel	1
pub_process_time	Processing Time	Average time spent processing an event access.	≥ 0 ms	Event channel	1

Table 4-5 Dimension description

Dimension	Key	Value
Event channel	channel_id	Event channel ID

4.6.3 Configuring Alarm Rules

This section describes the alarm policies of some metrics and how to configure them. In actual services, you are advised to configure alarm rules for metrics based on the following alarm policies.

Table 4-6 Parameters for alarm settings

Parameter	Description
Name	Name of the alarm rule. The system generates a name randomly but you can change it.
Description	Alarm rule description. This parameter is optional.
Alarm Type	Alarm type to which the alarm rule applies. Default: **Metric**.
Resource Type	Resource type. Default: **EventGrid**.
Dimension	Alarm dimension. Default: **Event Channels**.
Monitoring Scope	Resources to monitor. Default: **Specific resources**.
Monitored Objects	Object to monitor. Default: event channel name.
Method	Alarm triggering method. Default: **Configure manually**.
Alarm Policy	Policy that triggers an alarm. For details, see **Table 4-7**. You cannot modify or add alarm policies for metric alarm rules created on the EG console.
Alarm Notification	After you enable this function and configure required parameters, you will be notified of alarms and alarm clearance by notification group or topic subscription.
Notification Recipient	Select **Notification group** or **Topic subscription**.
Notification Group	Select a notification group. If no notification group is available, create one by referring to **Creating a Notification Object or Notification Group**.

Parameter	Description
Notification Object	Select a notification contact and topic. If no topic is available, create one by referring to **Creating a Notification Object or Notification Group**.
Notification Window	Alarm notifications are only sent during the configured validity period.
Trigger Condition	Condition for triggering a notification.
Enterprise Project	Enterprise project to which the alarm rule belongs. For details, see **Creating an Enterprise Project**.

Table 4-7 Alarm policy parameters

Period	Number of Times	Comparison	Value	Interval	Severity
Raw data	1 time	≥	Number	Every 10 minutes	Critical
Max.	2 consecutive times	>	Number	Every 15 minutes	Major
Min.	3 consecutive times	≤	Number	Every 30 minutes	Minor
Sum	4 consecutive times	<	Number	Every hour	Information
Variance	5 consecutive times	=	Number	Every 3 hours	Information

Procedure

Step 1 Log in to the **EG console**.

Step 2 Choose **Event Channels**.

Step 3 Click **Monitor** in the **Operation** column to go to the monitoring page.

Step 4 Hover over a metric and click + to create an alarm rule for it.

Step 5 Specify the alarm rule details.

For details about how to create an alarm rule, see **Creating an Alarm Rule**.

----End

5 Event Subscriptions

5.1 Creating an Event Subscription

Event subscriptions bind event sources, channels, and targets, and route events of sources to targets based on specified rules.

Constraints

- A subscription can be bound with up to five targets.
- An event can be transmitted three times in an EG channel.

Prerequisites

- (Optional) You have **created an event source**.
- You have set an event target.
- You have created an enterprise project. For details, see **IAM Projects and Enterprise Projects**.
  - **Enterprise Project** is not displayed for non-enterprise users and enterprise projects with only the **default** value.
  - If you have multiple enterprise projects, **default** is selected by default.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click **Create Event Subscription**.

Step 4 Click  next to the default subscription name.

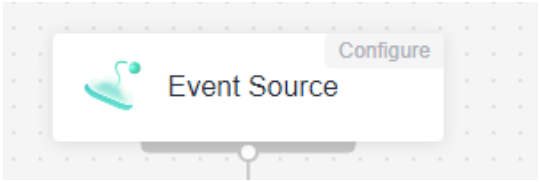
Step 5 Enter a new subscription name and description, and click **OK**.

Step 6 For enterprise users, select an enterprise project. You can click **View Enterprise Projects** to view the enterprise project list.

Step 7 Configure an event source.

1. Click **Event Source**, as shown in **Figure 5-1**.

   Figure 5-1 Configuring an event source

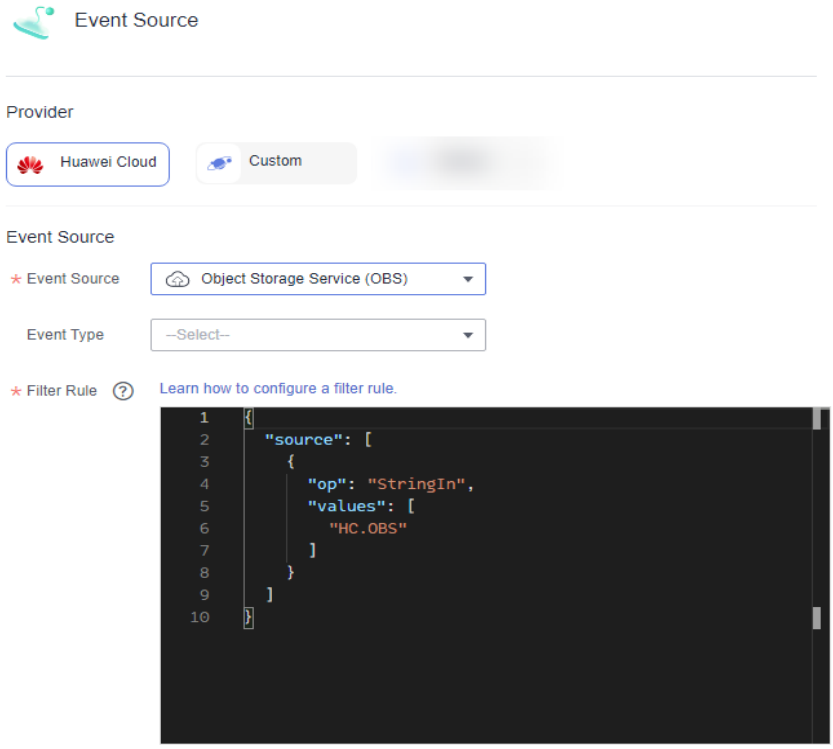
2. Select an event source provider.
   - **Cloud services**: cloud service event source.
   - **Custom**: custom event source
3. Set event source parameters.

   When selecting **Cloud services**, set the parameters listed in **Table 5-1**.

   Table 5-1 Cloud service event source parameters

Parameter	Description
Event Source	Select a cloud service event source.
Event Type	(Optional) Select a predefined event type.
Filter Rule	Enter an event filter rule. Only events that match these filter rules will be routed to the associated targets. For more information about filter rules, see **Filter Rule Parameters** and **Example Filter Rules**.

Figure 5-2 Configuring a cloud service event source



If **Event Source** is set to **OBS Application Service**, refer to **Table 5-2**. In addition, you need to add the **Tenant Administrator** permission to your Huawei Cloud account. For details, see **Assigning Dependency Roles**.

Table 5-2 OBS application event source parameters

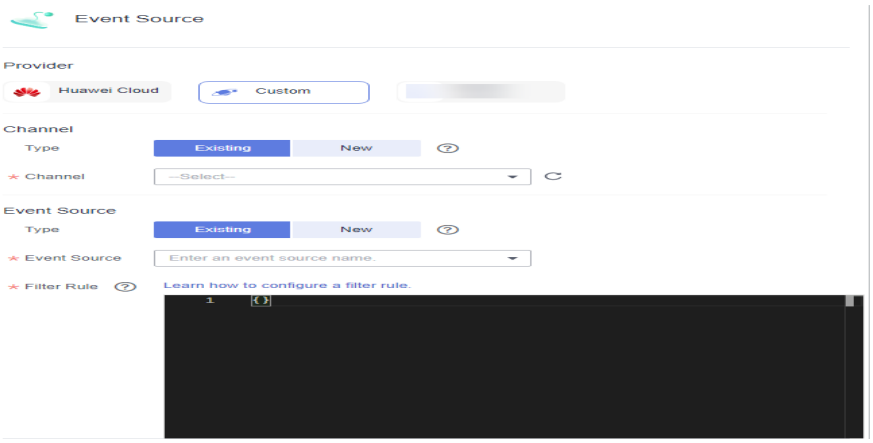
Parameter	Description
Source	Select an event source.
Bucket	Select an OBS bucket.
Event Type	Select event types to filter.
Object Name Prefix	Enter an object name prefix.
Object Name Suffix	Enter an object name suffix.
Object Name Encoding	Whether to encode object names of OBS events.
Filter Rule	Enter an event filter rule. Only events that match these filter rules will be routed to the associated targets. For more information about filter rules, see **Filter Rule Parameters** and **Example Filter Rules**.

When selecting **Custom**, set the parameters listed in **Table 5-3**.

Table 5-3 Custom event source parameters

Parameter	Description
Channel	Select an existing custom event channel, for example, **channel**.
Event Source	Enter or select a custom event source that has been associated with the selected custom event channel.
Filter Rule	Enter an event filter rule. Only events that match these filter rules will be routed to the associated targets. For more information about filter rules, see **Filter Rule Parameters** and **Example Filter Rules**.

Figure 5-3 Configuring a custom event source

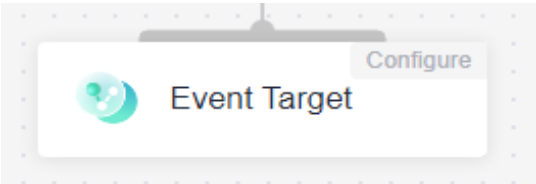


4. Click **OK**.

Step 8 Configure an event target.

1. Click **Event Target**, as shown in **Figure 5-4**.

Figure 5-4 Configuring an event target



2. Select an event target provider.
   - Cloud services: cloud service event target.

- **Custom**: custom event target

3. Set event target parameters.

   When selecting **Cloud services**, set the following parameters.

   - **Event Target**: Select an event target.

     If you set **Event Target** to **FunctionGraph (function computing)**:

     - **Function**: Select the function to trigger. If no function is available, **create a function** first.
     - **Version/Alias**: Choose to specify a version or alias.
     - **Version**: Select a version of the function. By default, **latest** is selected.
     - **Alias**: Select an alias of the function.
     - **Execute**: Select **Asynchronously** or **Synchronously**.

       NOTE

       Function invocation mode. Default: **Asynchronously**.

       Asynchronously: Immediate responses of function invocation are not required.

       Synchronously: Immediate responses of function invocation are required.

     - **Agency**: Select an agency. If no agency is available, click **Create Agency** to generate one named **EG_TARGET_AGENCY**.

       1) Only agencies with EG as the delegated cloud service are displayed.
       2) Select an agency with the permission **functiongraph:function:invoke***.

     If you set **Event Target** to **Distributed Message Service (DMS) for Kafka**:

     - **Connection**: Select a **DMS for Kafka connection**.
     - **Topic**: Select a message topic.
     - **Enable**: Whether to enable the message key function.
     - **Transform Type**: Defines how message keys are used. There are two options:
       - **Variables**: Keys are variable values from CloudEvents-compliant events.
       - **Constants**: Keys are specified constants. All messages will be sent to the same partition.

       For more information about the transform types, see **Event Content Transformation**.

     If you set **Event Target** to **Cloud service API**:

     - **Cloud Service**: Select a cloud service.

- **API**: Select a cloud service API and configure the API information.
- **Agency**: Select the created agency.

If you set **Event Target** to **Simple Message Notification (SMN)**:

- **Topic**: Select a message topic.
- **Agency**: Select an agency. If no agency is available, click **Create Agency** to generate one named **EG_SMN_PUBLISHER_AGENCY**.
  - Only agencies with EG as the delegated cloud service are displayed.
  - Select an agency with the permission **smn:topic:publish**.
- **Message Subject**: Configure the subject through constants or variables.
- **Type**: Type of the message subject. Two types are available:
  - **Constants**: The subject does not change from specified. All messages will use the same subject.
  - **Variables**: The subject in the template is a variable value from CloudEvents-compliant events. Max.: 512 characters.

    NOTE

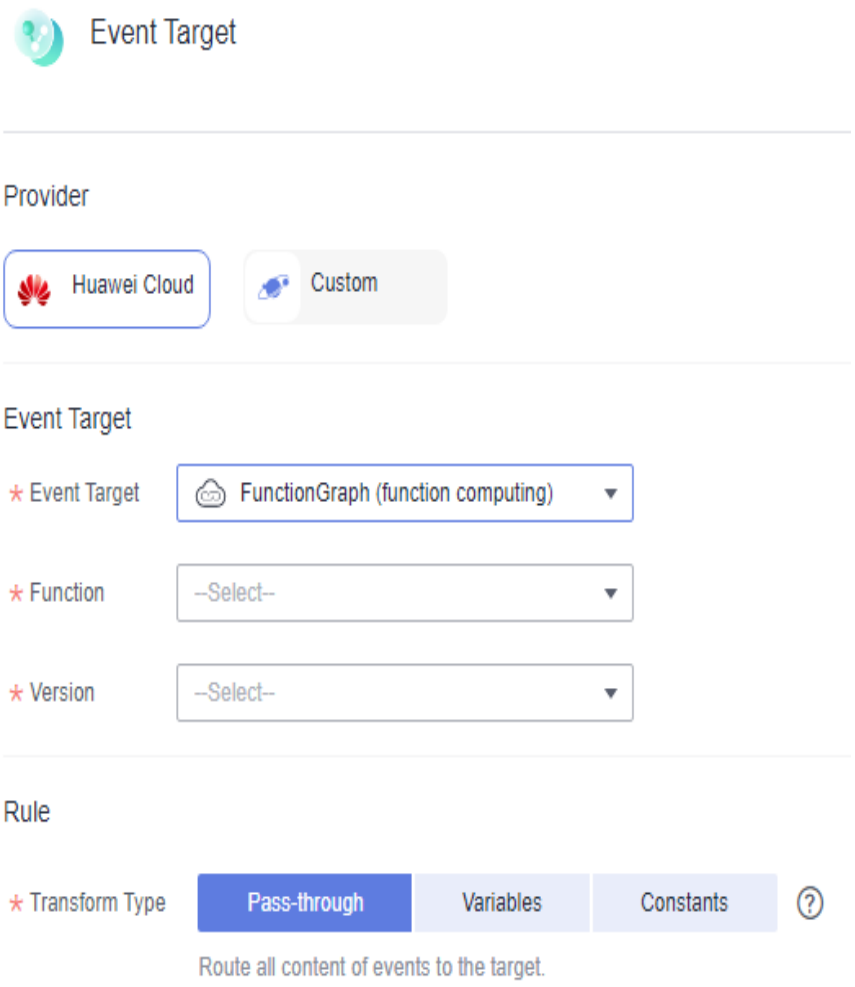
    The **Subject** parameter is optional.

Rule:

- **Transform Type**: EG transforms CloudEvents-compliant events for targets. The following three types are supported:
  - **Pass-through**: Directly route CloudEvents-compliant events to the target.
  - **Variables**: Route variables in CloudEvents-compliant events to the target.
  - **Constants**: Route constants in events to the target.

  For more information about the transform types, see **Event Content Transformation**.

Figure 5-5 Configuring a cloud service event target



When selecting **Custom**, set the following parameters.

- **URL**: Enter the URL of the event target starting with **http://** or **https://** and using the **POST** method.

  NOTE

  The default timeout for pushing events to custom targets is 9 seconds. Exceeding this limit will cause delivery failure.

  HTTP has high security risks, which may cause data eavesdropping or tampering. For data security, you are advised to use HTTPS. If you must use HTTP, ensure that the custom event does not contain any sensitive information. You are responsible for all related risks.

Table 5-4 Event body parameters

Parameter	Required	Type	Description
datacontenttype	No	String	Data content type
data	Yes	Array of Data objects	Data
subject	No	String	Subject
specversion	No	String	Specification version
id	Yes	String	Id
source	Yes	String	Event source
time	Yes	String	Time
type	Yes	String	Event type
ttl	Yes	String	Timeout
dataschema	No	String	Data schema

The following is an example of the event body (**OBS Application Service** as the event source):

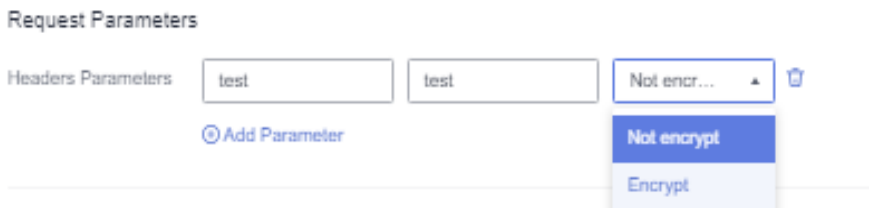
```
"datacontenttype":"application/json",
"data":
{
  "obs":{
          "bucket":{
                      "bucket":"bucket-input-my",
                      "name":"bucket-input-my",
                      "arn":"",
                      "ownerIdentity":{"ID":"f9e40463cxxxxxxxx9efd3a7ec854e"}
                    },
          "Version":"1.0",
          "configurationId":"a6b0bcf8-8874-4d8b-84f5-f9068527930f",
          "object":{
                      "versionId":"G00101928EE6072DFFFFD28824BB4AB8null",
                      "oldpsxpth":"",
                    "size":10,
                      "eTag":"c9b20f7d442e65ede148e006dfe1308c",
                      "key":"\xe4\xba\x8b\xe4\xbb\xxx\xxx\xxx\xxx\xxx\xxx\x85\xe6\xb5\x8b
\xe8\xaf\x951015-2.txt",
                      "sequencer":"1"
                    }
          },
  "eventVersion":"3.0",
  "responseElements":{
                      "x-obs-id-2":"",
                      "x-obs-request-id":"3f905af2683e6ed7dbaec881b66390ab",
                      "x-amz-request-id":"",
                      "x-amz-id-2":""
                    },
  "eventSource":"OBS",
  "eventTime":"2024-10-15T14:38:12.781Z",
  "requestParameters":{
                      "sourceIPAddress":"xx.xx.xx.xx"
                    },
```



```
 "eventName":"ObjectCreated:Put",
 "eventRegion":"cn-north-4",
 "userIdentity":{
                   "ID":"f9e40463c23xxxxxxxefd3a7ec854e"
                 }
},
"subject":"\xe4\xba\x8b\xe4\xbb\xxx\xxx\xxx\xxx\xxx\xxx\xxx\xe6\xb5\x8b\xe8\xaf
\x951015-2.txt",
"specversion":"1.0",
"id":"3f905af2683e6ed7dbaec881b66390ab",
"source":"HC.OBS.DWR",
"time":"2024-10-15T06:38:13.52240464Z",
"type":"OBS:DWR:ObjectCreated:PUT",
"ttl":"4000","dataschema":""
```


- **Connection**: Select a custom or the default connection. For details, see **Connections**.
- **Headers Parameters**
  - Enter a request header.
  - Enter a value.
  - Specify whether to encrypt the header.

    Figure 5-6 Header parameters

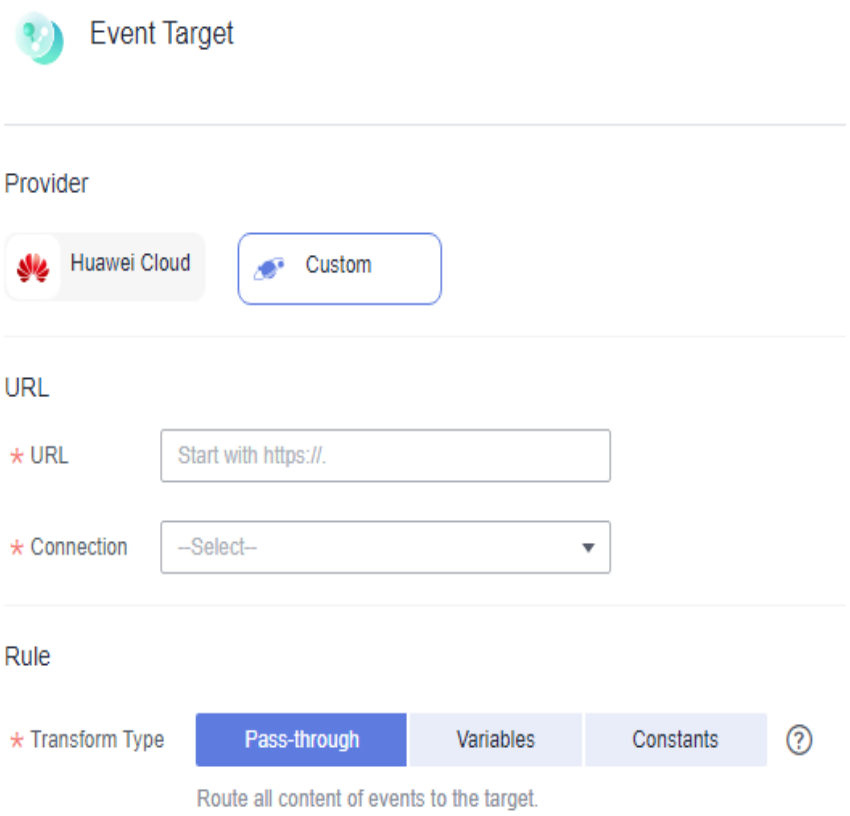
    NOTE

    - For custom HTTPS events, add authorization configurations for the event target to improve security.
    - If the request header and value are invalid, the encryption option is unavailable.
    - **Key**: Max. 256 characters starting and ending with a letter. Only letters and hyphen (-) are allowed.
    - **Value**: Max. 1024 characters, including letters, hyphens (-), underscores (_), spaces, and special characters (~!@#$%^&*()=+|[{}];:'",<.>/?).
- **Transform Type**: EG transforms CloudEvents-compliant events for targets. The following three types are supported:
  - **Pass-through**: Directly route CloudEvents-compliant events to the target.
  - **Variables**: Route variables in CloudEvents-compliant events to the target.
  - **Constants**: Route constants in events to the target.

  For more information about the transform types, see **Event Content Transformation**.

- **Status**: The dead letter queue is disabled by default. If enabled, EG sends failed events to the specified queue. If disabled, such events will be discarded. For details, see **Dead Letter Queue**.

Figure 5-7 Configuring a custom event target



4. Click **OK**.

Step 9 Click **Save**.

The subscription is enabled by default once created.

----End

5.2 Editing an Event Subscription

Modify the description, status, event source, and event target of a subscription.

Constraints

- The event source provider cannot be changed.
- The bound custom event channel cannot be changed.

Modifying the Description

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click **Configure** in the row that contains the desired subscription to go to the details page.

Step 4 Click the edit icon next to the default subscription name.

Step 5 Modify the description and click **OK**.

----End

Changing the Status

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click **Disable** or **Enable** in the row that contains the desired subscription.

----End

Changing the Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click the name of the desired subscription to go to the details page.

Step 4 Click the event source card.

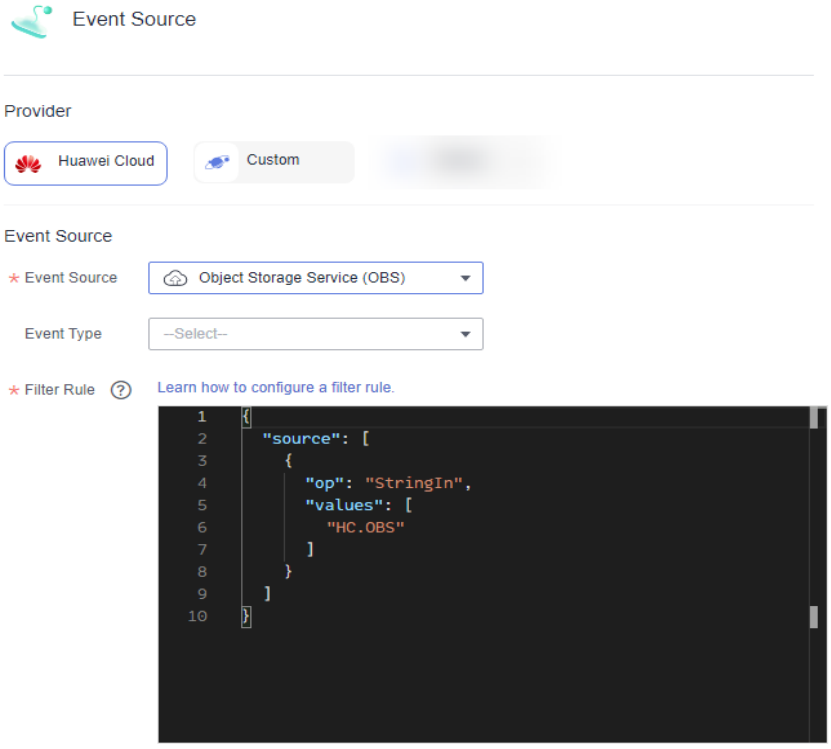
Step 5 Modify the event source parameters.

When selecting **Cloud services**, set the parameters listed in **Table 5-5**.

Table 5-5 Cloud service event source parameters

Parameter	Description
Event Source	Select a cloud service event source.
Event Type	(Optional) Select a predefined event type.
Filter Rule	Enter an event filter rule. Only events that match these filter rules will be routed to the associated targets. For more information about filter rules, see **Filter Rule Parameters** and **Example Filter Rules**.

Figure 5-8 Configuring a cloud service event source

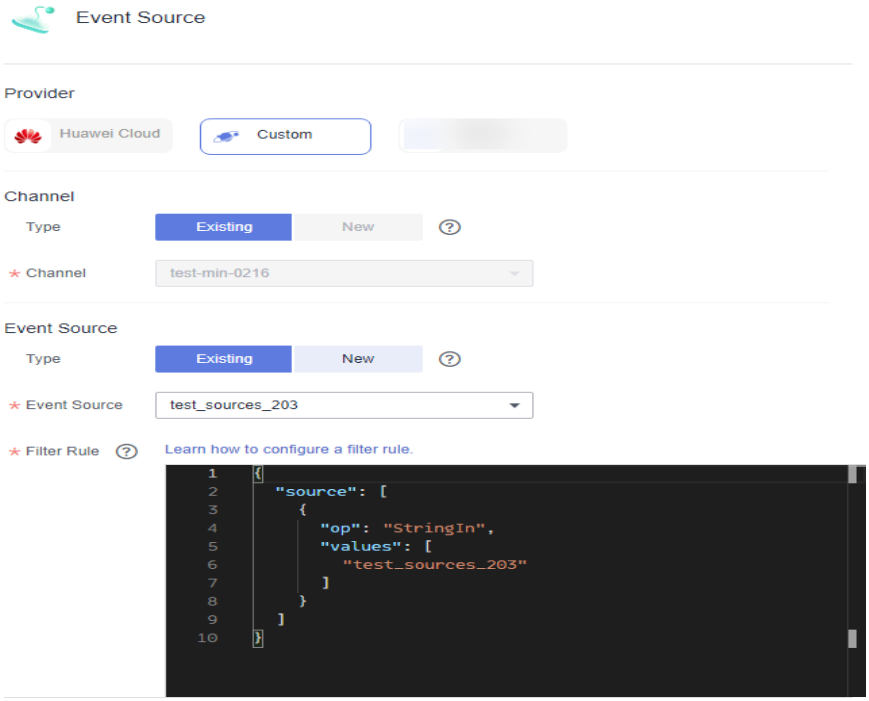


When selecting **Custom**, set the parameters listed in **Table 5-6**.

Table 5-6 Custom event source parameters

Parameter	Description
Channel	Select an existing custom event channel, for example, **channel**.
Event Source	Enter or select a custom event source that has been associated with the selected custom event channel.
Filter Rule	Enter an event filter rule. Only events that match these filter rules will be routed to the associated targets. For more information about filter rules, see **Filter Rule Parameters** and **Example Filter Rules**.

Figure 5-9 Configuring a custom event source



Step 6 Click **OK**.

Step 7 Click **Save**.

----End

Changing the Event Target

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click the name of the desired subscription to go to the details page.

Step 4 Change the event target or add another one.

- Click the event target card to change the event target.
- Click ⊕ to add an event target.
- Click 🗑 to delete an event target.

Step 5 Set the event target provider and relevant parameters.

When selecting **Cloud services**, set the following parameters.

- **Event Target**: Select an event target.

  If you set **Event Target** to **FunctionGraph (function computing)**:
  - **Function**: Select the function to trigger. If no function is available, **create a function** first.
  - **Version/Alias**: Choose to specify a version or alias.
  - **Version**: Select a version of the function. By default, **latest** is selected.
  - **Alias**: Select an alias of the function.

- **Execute**: Select **Asynchronously** or **Synchronously**.

  NOTE

  Function invocation mode. Default: **Asynchronously**.

  Asynchronously: Immediate responses of function invocation are not required.

  Synchronously: Immediate responses of function invocation are required.

- **Agency**: Select an agency. If no agency is available, click **Create Agency** to generate one named **EG_TARGET_AGENCY**.
  i. Only agencies with EG as the delegated cloud service are displayed.
  ii. Select an agency with the permission **functiongraph:function:invoke***.

If you set **Event Target** to **Distributed Message Service (DMS) for Kafka**:

- **Connection**: Select a **DMS for Kafka connection**.
- **Topic**: Select a message topic.
- **Enable**: Whether to enable the message key function.
- **Transform Type**: Defines how message keys are used. There are two options:
  - **Variables**: Keys are variable values from CloudEvents-compliant events.
  - **Constants**: Keys are specified constants. All messages will be sent to the same partition.

  For more information about the transform types, see **Event Content Transformation**.

If you set **Event Target** to **Cloud service API**:

- **Cloud Service**: Select a cloud service.
- **API**: Select a cloud service API and configure the API information.
- **Agency**: Select the created agency.

If you set **Event Target** to **Simple Message Notification (SMN)**:

- **Topic**: Select a message topic.
- **Agency**: Select an agency. If no agency is available, click **Create Agency** to generate one named **EG_SMN_PUBLISHER_AGENCY**.
  - Only agencies with EG as the delegated cloud service are displayed.
  - Select an agency with the permission **smn:topic:publish**.
- **Message Subject**: Configure the subject through constants or variables.
- **Type**: Type of the message subject. Two types are available:
  - **Constants**: The subject does not change from specified. All messages will use the same subject.
  - **Variables**: The subject in the template is a variable value from CloudEvents-compliant events. Max.: 512 characters.

    NOTE

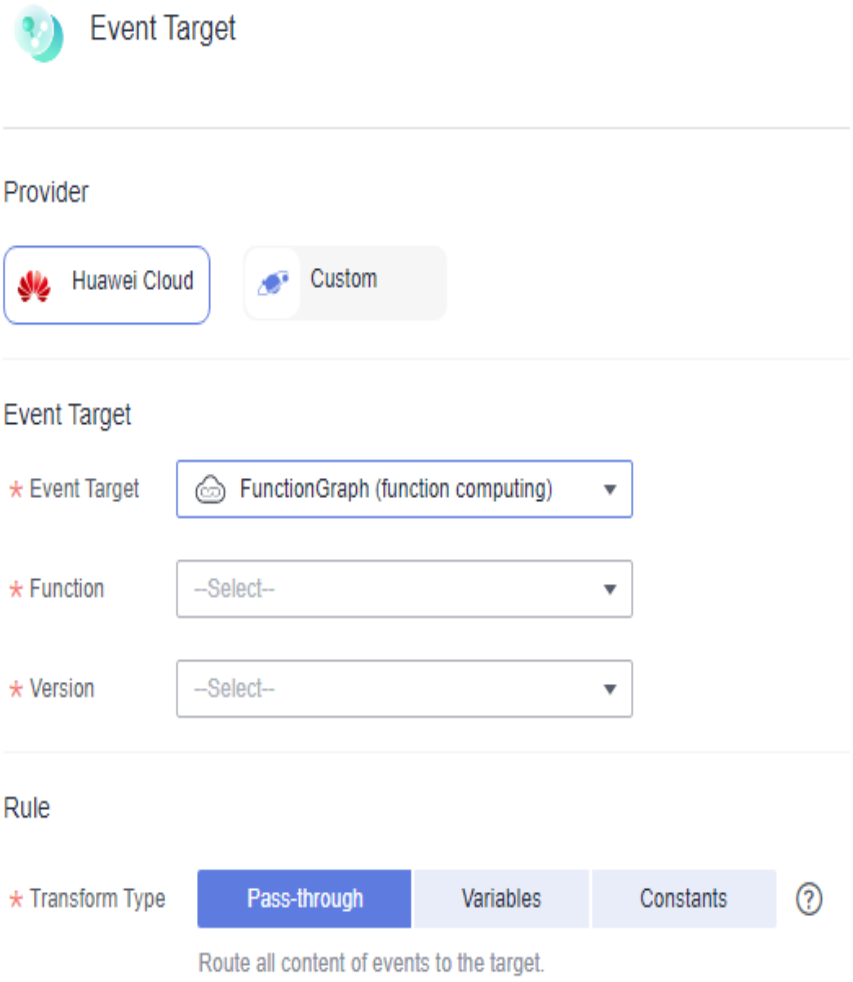
    The **Subject** parameter is optional.

Rule:

- **Transform Type**: EG transforms CloudEvents-compliant events for targets. The following three types are supported:
  - **Pass-through**: Directly route CloudEvents-compliant events to the target.
  - **Variables**: Route variables in CloudEvents-compliant events to the target.
  - **Constants**: Route constants in events to the target.

  For more information about the transform types, see **Event Content Transformation**.

Figure 5-10 Configuring a cloud service event target



When selecting **Custom**, set the following parameters.

- **URL**: Enter the URL of the event target starting with **http://** or **https://** and using the **POST** method.

> **NOTE**
>
> The default timeout for pushing events to custom targets is 9 seconds. Exceeding this limit will cause delivery failure.

HTTP has high security risks, which may cause data eavesdropping or tampering. For data security, you are advised to use HTTPS. If you must use HTTP, ensure that the custom event does not contain any sensitive information. You are responsible for all related risks.

Table 5-7 Event body parameters

Parameter	Required	Type	Description
datacontenttype	No	String	Data content type
data	Yes	Array of Data objects	Data
subject	No	String	Subject
specversion	No	String	Specification version
id	Yes	String	Id
source	Yes	String	Event source
time	Yes	String	Time
type	Yes	String	Event type
ttl	Yes	String	Timeout
dataschema	No	String	Data schema

The following is an example of the event body (**OBS Application Service** as the event source):

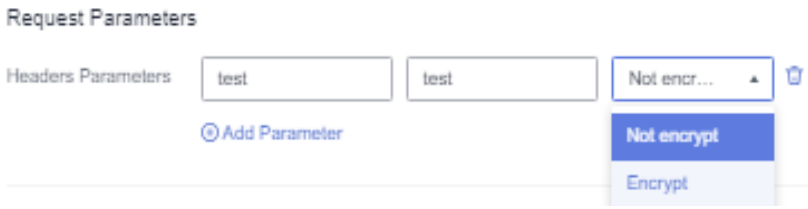
```
"datacontenttype":"application/json",
"data":
{
  "obs":{
          "bucket":{
                      "bucket":"bucket-input-my",
                      "name":"bucket-input-my",
                      "arn":"",
                      "ownerIdentity":{"ID":"f9e40463cxxxxxxxx9efd3a7ec854e"}
                    },
          "Version":"1.0",
          "configurationId":"a6b0bcf8-8874-4d8b-84f5-f9068527930f",
          "object":{
                      "versionId":"G00101928EE6072DFFFFD28824BB4AB8null",
                      "oldpsxpth":"",
                     "size":10,
                      "eTag":"c9b20f7d442e65ede148e006dfe1308c",
                      "key":"\xe4\xba\x8b\xe4\xbb\xxx\xxx\xxx\xxx\xxx\xxx\x85\xe6\xb5\x8b\xe8\xaf
\x951015-2.txt",
                      "sequencer":"1"
                     }
```



```
        },
 "eventVersion":"3.0",
 "responseElements":{
                      "x-obs-id-2":"",
                      "x-obs-request-id":"3f905af2683e6ed7dbaec881b66390ab",
                      "x-amz-request-id":"",
                      "x-amz-id-2":""
                     },
 "eventSource":"OBS",
 "eventTime":"2024-10-15T14:38:12.781Z",
 "requestParameters":{
                      "sourceIPAddress":"xx.xx.xx.xx"
                     },
 "eventName":"ObjectCreated:Put",
 "eventRegion":"cn-north-4",
 "userIdentity":{
                   "ID":"f9e40463c23xxxxxxxxefd3a7ec854e"
                 }
},
"subject":"\xe4\xba\x8b\xe4\xbb\xxx\xxx\xxx\xxx\xxx\xxx\xxx\xe6\xb5\x8b\xe8\xaf\x951015-2.txt",
"specversion":"1.0",
"id":"3f905af2683e6ed7dbaec881b66390ab",
"source":"HC.OBS.DWR",
"time":"2024-10-15T06:38:13.52240464Z",
"type":"OBS:DWR:ObjectCreated:PUT",
"ttl":"4000","dataschema":""
```


- **Connection**: Select a custom or the default connection. For details, see **Connections**.
- **Headers Parameters**
  - Enter a request header.
  - Enter a value.
  - Specify whether to encrypt the header.

    Figure 5-11 Header parameters

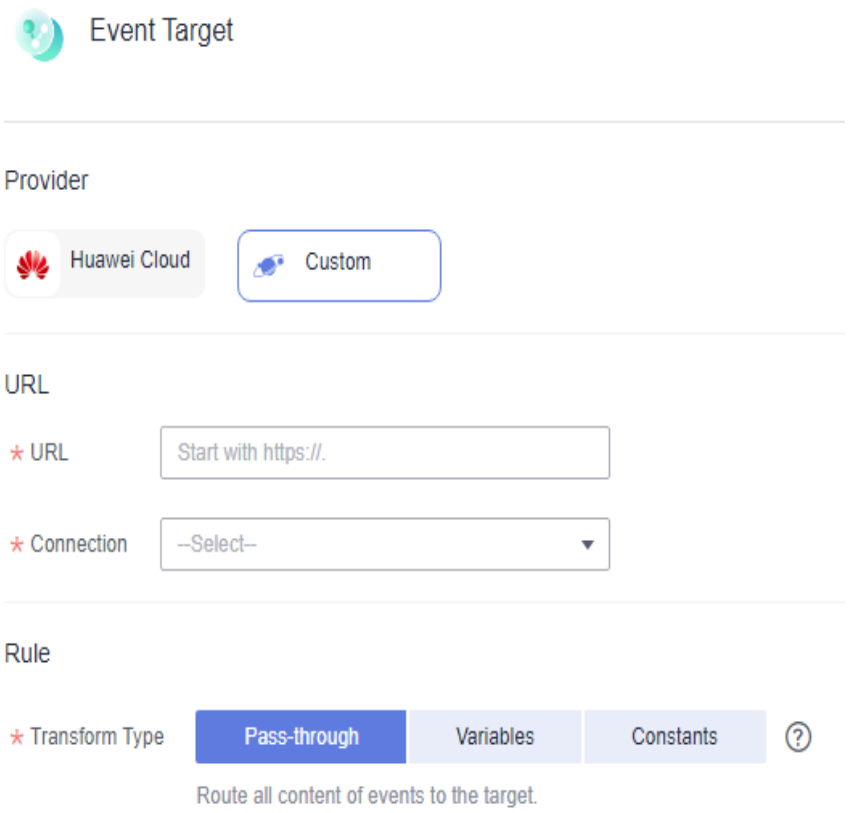
    NOTE

    - For custom HTTPS events, add authorization configurations for the event target to improve security.
    - If the request header and value are invalid, the encryption option is unavailable.
    - **Key**: Max. 256 characters starting and ending with a letter. Only letters and hyphen (-) are allowed.
    - **Value**: Max. 1024 characters, including letters, hyphens (-), underscores (_), spaces, and special characters (~!@#$%^&*()=+|[{}];:'",<.>/?).
- **Transform Type**: EG transforms CloudEvents-compliant events for targets. The following three types are supported:

  - **Pass-through**: Directly route CloudEvents-compliant events to the target.
  - **Variables**: Route variables in CloudEvents-compliant events to the target.
  - **Constants**: Route constants in events to the target.

  For more information about the transform types, see **Event Content Transformation**.
- **Status**: The dead letter queue is disabled by default. If enabled, EG sends failed events to the specified queue. If disabled, such events will be discarded. For details, see **Dead Letter Queue**.

Figure 5-12 Configuring a custom event target



Step 6 Click **OK**.

Step 7 Click **Save**.

----End

5.3 Deleting an Event Subscription

Delete an event subscription that will no longer be used.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click **Delete** in the row that contains the desired event subscription.

Step 4 Click **OK**.

----End

5.4 Dead Letter Queue

Introduction

If the dead letter queue function is enabled, EG sends failed events to the specified queue. If disabled, such events will be discarded.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Subscriptions**.

Step 3 Click **Create Event Subscription**.

Step 4 Click **Event Target**.

Step 5 In the displayed dialog box, select an event target.

Step 6 Enable **Dead Letter Queue** and configure the required parameters.

Figure 5-13 Configuring a dead letter queue

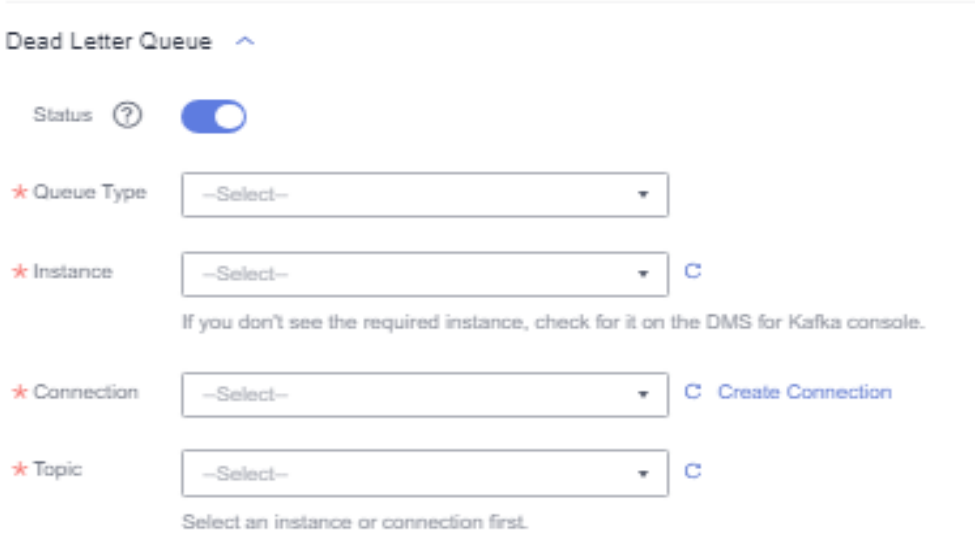


Table 5-8 Dead letter queue parameters

Parameter	Description
Queue Type	Select a queue type.
Instance	Select an instance.
Connection	Select a connection. **Only the connection that matches the queue type can be selected.**
Topic	Select a topic. **Do not use the same topic as the event target, or EG cannot distinguish successful events from failed ones.**

Step 7 Click **OK**.

----End

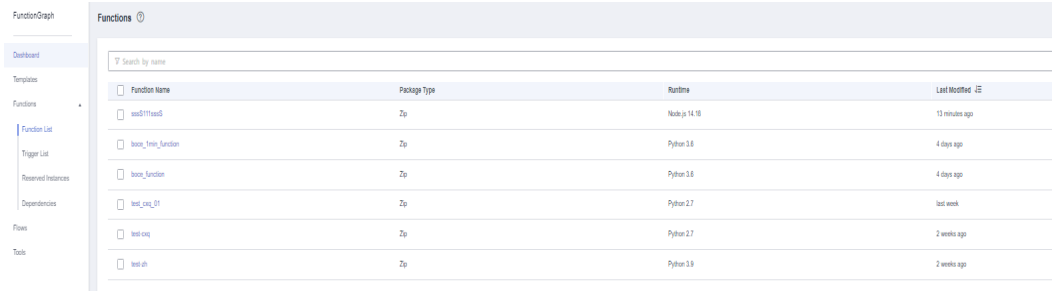
Processing Data in the Dead Letter Queue

Perform the following procedure to process the data in the dead letter queue.

Step 1 Log in to the **FunctionGraph console**. In the navigation pane, choose **Functions** > **Function List**.

Step 2 Click **Create Function** in the upper right. For details, see **Creating an Event Function**.

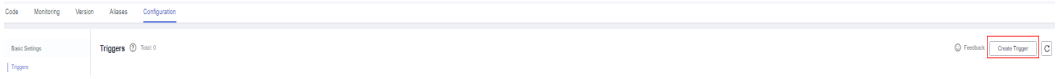
Figure 5-14 Function list



Step 3 Click the created function to go to the details page.

Step 4 Choose **Configuration** > **Triggers** and click **Create Trigger**.

Figure 5-15 Creating a trigger



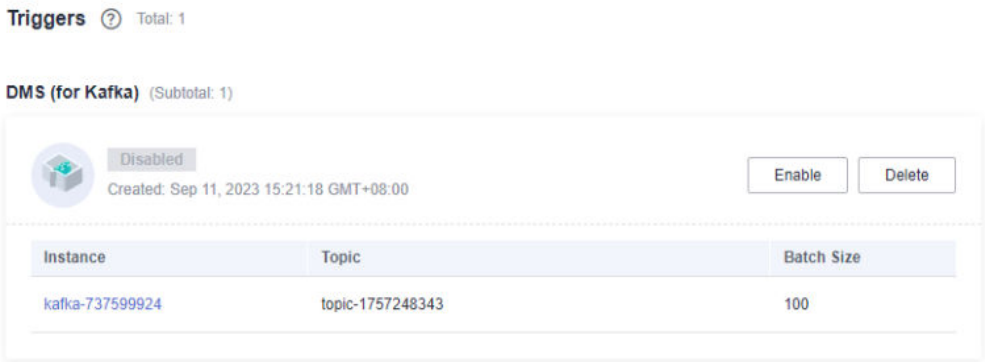
Step 5 Set the following parameters:

- **Trigger Type**: Select **DMS (for Kafka)**.
- **Instance**: Select the same Kafka instance as the dead letter queue.
- **Topic**: Select the same topic as the dead letter queue.
- **Batch Size**: Set the number of messages to be retrieved from the topic each time. Recommended: **10**.
- **Username**: Enter the username of the instance if SSL has been enabled for it.
- **Password**: Enter the password of the instance if SSL has been enabled for it.

Step 6 Click **OK**.

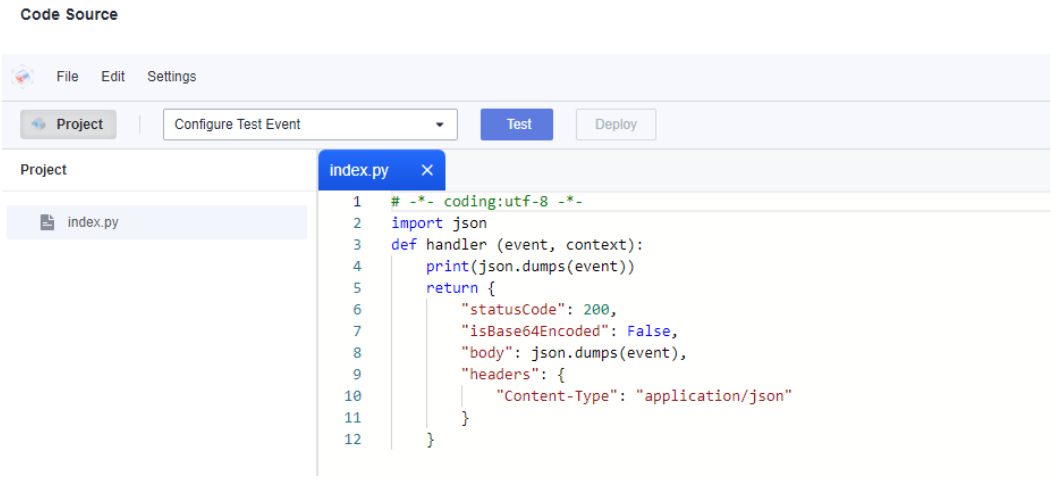
Step 7 Click **Enable** to enable the Kafka trigger.

Figure 5-16 Enabling a Kafka trigger



Step 8 Compile the logic for processing data in the dead letter queue.

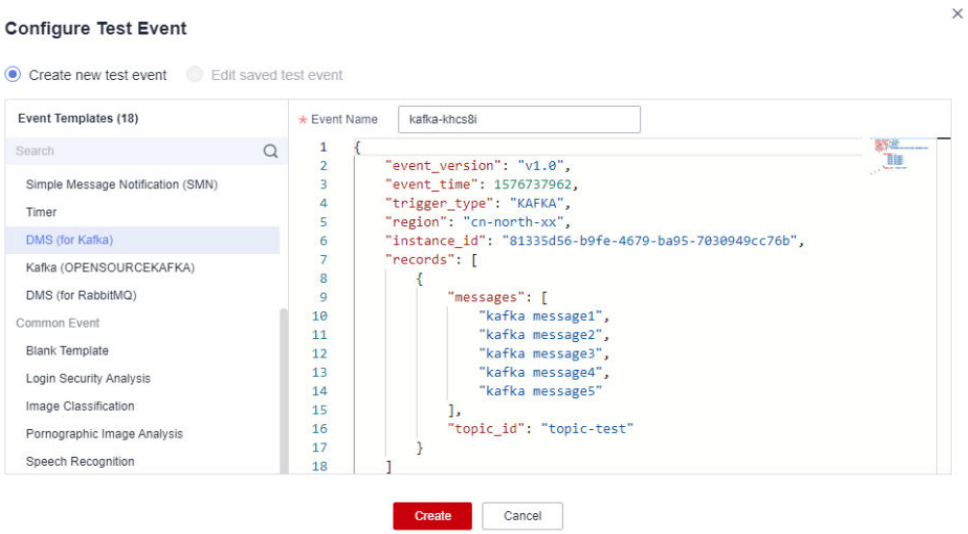
Figure 5-17 Data processing logic



Step 9 Configure a test event.

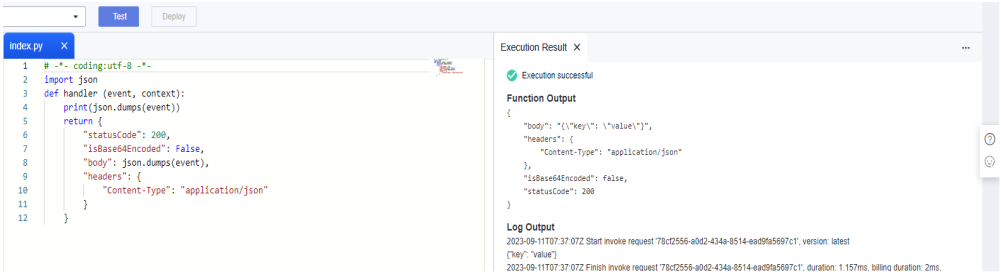
1. Click **Configure Test Event**.
2. Select **DMS (for Kafka)** and click **Create**.

Figure 5-18 Configuring a test event



3. Select the created test event from the drop-down list.
4. Click **Test** and then view the execution result.

Figure 5-19 Execution result



----End

5.5 Monitoring

5.5.1 Viewing Monitoring Data

Scenario

Cloud Eye monitors event subscription metrics in real time. You can view these metrics on the Cloud Eye console.

Prerequisites

You have created an event subscription.

Procedure

Step 1 Log in to the **EG console**.

Step 2 Choose **Event Subscriptions**.

Step 3 Click **Monitoring** in the row that contains the target event subscription to go to the monitoring page. Data of all delivered events in the last hour is displayed by default.

You can also click **1h**, **3h**, **12h**, **1d**, **7d**, or **30d** to view event deliveries in different periods.

NOTE

- The time range can be customized.
- If the event subscription has multiple targets, select one to view its monitoring data. By default, the monitoring data of all targets is displayed.
- If you enable **Auto Refresh**, the metric data is refreshed every 5 seconds.
- Click **View details** to go to the Cloud Eye console.
- If you set **Period** to **Raw data**, the raw monitoring data is displayed. If you set **Period** to a specific time, you can select different aggregation methods, including **Avg.**, **Max.**, **Min.**, **Sum**, and **Variance**.

----End

5.5.2 Supported Metrics

Introduction

This section describes the event subscription metrics and dimensions reported to Cloud Eye. You can search metrics and alarms on the Cloud Eye console or on the monitoring page of EG.

Metrics

Table 5-9 Metric description

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
sub_num	Total Deliveries	Number of times event delivery is attempted.	≥ 0	Event subscription	1
sub_success_num	Successful Deliveries	Number of times events are actually delivered.	≥ 0	Event subscription	1

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
sub_success_rate	Success Rate	Percentage of total deliveries that are successful.	0%–100%	Event subscription	1
sub_failed_num	Failed Deliveries	Number of times events could not be delivered.	≥ 0	Event subscription	1
sub_failed_rate	Failure Rate	Percentage of total deliveries that failed.	0%–100%	Event subscription	1
sub_retry_num	Delivery Retries	Number of times delivery retry is attempted.	≥ 0	Event subscription	1
sub_retry_rate	Retry Rate	Percentage of total deliveries that are retried.	0%–100%	Event subscription	1
sub_process_time	Processing Time	Average time spent processing an event delivery.	≥ 0 ms	Event subscription	1

Table 5-10 Dimension description

Dimension	Key	Value
Event subscription	subscription_id	Event subscription ID
Event subscription - target	target_id	Event target ID

5.5.3 Configuring Alarm Rules

This section describes the alarm policies of some metrics and how to configure them. In actual services, you are advised to configure alarm rules for metrics based on the following alarm policies.

Table 5-11 Parameters for alarm settings

Parameter	Description
Name	Name of the alarm rule. The system generates a name randomly but you can change it.
Description	Alarm rule description. This parameter is optional.
Alarm Type	Alarm type to which the alarm rule applies. Default: **Metric**.
Resource Type	Resource type. Default: **EventGrid**.
Dimension	Alarm dimension. Default: **Event Subscriptions**.
Monitoring Scope	Resources to monitor. Default: **Specific resources**.
Monitored Objects	Object to monitor. Default: event subscription name.
Method	Alarm triggering method. Default: **Configure manually**.
Alarm Policy	Policy that triggers an alarm. For details, see **Table 4-7**. **If a metric alarm policy is created on the EG page, you cannot modify or add other metric alarm policies.**
Alarm Notification	After you enable this function and configure required parameters, you will be notified of alarms and alarm clearance by notification group or topic subscription.
Notification Recipient	Select **Notification group** or **Topic subscription**.
Notification Group	Select a notification group. If no notification group is available, create one by referring to **Creating a Notification Object or Notification Group**.

Parameter	Description
Notification Object	Select a notification contact and topic. If no topic is available, create one by referring to **Creating a Notification Object or Notification Group**.
Notification Window	Alarm notifications are only sent during the configured validity period.
Trigger Condition	Condition for triggering a notification.
Enterprise Project	Enterprise project to which the alarm rule belongs. For details, see **Creating an Enterprise Project**.

Procedure

Step 1 Log in to the **EG console**.

Step 2 Choose **Event Subscriptions**.

Step 3 Click **Monitor** in the **Operation** column to go to the monitoring page.

Step 4 Hover over a metric and click  to create an alarm rule for it.

Step 5 Specify the alarm rule details.

For details about how to create an alarm rule, see **Creating an Alarm Rule**.

----End

6 Event Streams

6.1 Serverless Event Streams

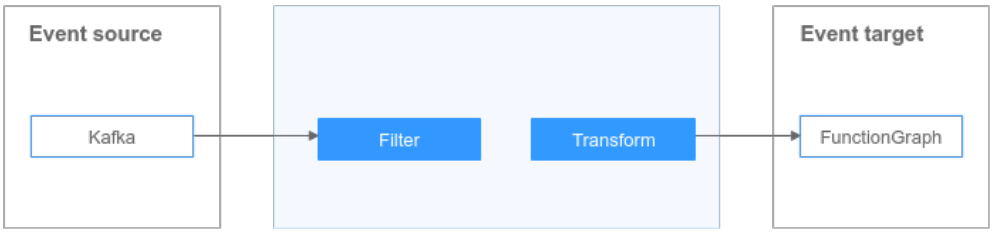
6.1.1 Overview

As service systems scale, data streams from sources surge in volume and velocity, demanding more efficient analysis and processing.

Event streams pull, filter, and transform events generated by event sources in real time, and route them to event targets for lightweight and efficient stream processing.

As shown in the following figure, this approach bypasses traditional subscription mechanisms, enabling seamless transfer between sources and targets for immediate, accurate data delivery.

Figure 6-1 Event stream



6.1.2 Event Source

6.1.2.1 Distributed Message Service (DMS) for Kafka

Configure a DMS for Kafka instance as the source of an event stream.

Prerequisites

1. You have purchased a Kafka instance on the DMS for Kafka console. For details, see **Buying a Kafka Instance**.

2. The security group of the Kafka instance allows access from the subnet and port of the Kafka instance in the inbound rules. For details, see **Viewing the Subnet CIDR Block**,**Kafka Instance Port**, and **Adding a Security Group Rule**.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Configure a Kafka event source.

1. Click **Event Source**.
2. Select **Distributed Message Service (DMS) for Kafka** for **Event Provider**.
3. Set event source parameters.

Table 6-1 Kafka parameters

Parameter	Description
Instance	Select a Kafka instance.
Access Mode	Select **Ciphertext Access** or **Plaintext Access**.
Security Protocol	If you select **Ciphertext Access** for **Access Mode**, the corresponding security protocol will be displayed.
Topic	Select a topic.
Consumer Group	Enter a consumer group name containing 3 to 64 characters. – If the consumer group does not exist, it will be created in the Kafka instance when the event stream is enabled. For details, see **Querying the Kafka Consumer Group List**. – If the consumer group has been created for the Kafka instance, enter the created consumer group name. For details, see **Creating a Kafka Consumer Group**.
Concurrency	Enter the number of concurrent messages. Range: 1–1000. This parameter is autofilled with the number of partitions for the selected topic. Recommended: retain this default number.
Consumption Offset	Select a consumption offset. – **Latest**: Consumption starts from the latest message in the queue. – **Earliest**: Consumption starts from the earliest message in the queue.

Parameter	Description
SASL Mechanism	This parameter is available when SASL_SSL authentication is enabled for the Kafka instance. Select an SASL authentication mechanism. – **PLAIN**: a simple username and password verification mechanism. – **SCRAM-SHA-512**: uses the hash algorithm to generate credentials for usernames and passwords to verify identities. SCRAM-SHA-512 is more secure than PLAIN.
SASL Certificate URL	This parameter is available when SASL_SSL authentication is enabled for the Kafka instance. Enter an SASL certificate URL. For details about the URL, see **How Do I Obtain the SASL Certificate Address of a DMS for Kafka Instance?**. – The package must be in ZIP format. The number of files in the package cannot exceed two. The size of the package and file cannot exceed 1 MB. – The certificate name must be fixed to **client.jks**. – **If the SASL certificate is modified, perform either of the following operations to ensure that the certificate takes effect:** 1. **If the SASL certificate file name is changed, obtain the certificate address again, enter the address, and click Save.** 2. If the file name remains the same but one of the parameter **Topic**, **Consumer Group**, **SASL Mechanism**, **SASL Certificate Key**, **Username**, or **Password** is modified, click **Save** to reload the certificate.
SASL Certificate Key	This parameter is available when SASL_SSL authentication is enabled for the Kafka instance. Enter an SASL certificate key. The SASL certificate key is **dms@kafka**, which is obtained from the Kafka instance details page.
Username	This parameter is available when SASL_SSL authentication is enabled for the Kafka instance. Enter a username.
Password	This parameter is available when SASL_SSL authentication is enabled for the Kafka instance. Enter a password.

Step 5 Click **Save**.

Step 6 Configure an event target by referring to **Routing to FunctionGraph**.

Step 7 After the event source and target are configured, click **Save** in the upper right.

NOTE

If sending events to the target fails, the whole batch of events will be retried until the processing is successful or the source message expires. The target must be able to process duplicate events.

----End

6.1.2.2 Open-Source RocketMQ

Add an open-source RocketMQ event source to an event stream.

Currently, RocketMQ 4.9.7 and 5.1.4 are supported.

Prerequisites

- A cluster instance of Open-Source RocketMQ is available.
- When the source is Open-Source RocketMQ, only FunctionGraph (function computing) can be selected for the target.

Creating an Open-Source RocketMQ Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Click the event source.

Step 5 Enter the event source information by referring to **Table 6-2**.

Table 6-2 Parameters of Open-Source RocketMQ

Parameter	Description
Event Provider	Open-Source RocketMQ
Addresses	Enter one or more connection addresses.
Group	Enter a group ID.
Topic	Enter a topic.
VPC	Select a VPC. **After an event stream is created, the VPC cannot be modified; otherwise, an error message will be displayed.**
Subnet	Select a subnet. **After an event stream is created, the subnet cannot be modified; otherwise, an error message will be displayed.**

Parameter	Description
SSL	Specify whether to enable SSL.
ACL	Specify whether to enable ACL. **When ACL is enabled, you need to configure the username and secret key.**
Tag	Enter one or more tags.
Consumption Timeout (ms)	Enter an integer ranging from 1,000 to 900,000.
Consumption Threads	Enter an integer ranging from 20 to 64.
Max. Messages	Enter an integer ranging from 1 to 32.

Step 6 Click **Next**. The rule configuration page is displayed. For details about how to configure rules, see **Filter Rule Parameters**.

Step 7 Click **Next** to complete the rule configuration. You can continue to configure the event target of FunctionGraph by referring to **Routing to FunctionGraph**.

NOTE

When **Event Source** is set to **Open-Source RocketMQ** and **Event Target** is set to **FunctionGraph**, **Execute** can be set to **Synchronously** or **Asynchronously**.

Step 8 After the event source and target are configured, click **Save** in the upper right.

NOTE

- The MQ collection function takes effect in minutes after it is started for the first time.
- In broadcast mode, retry upon consumption failure is not supported. New messages will continue to be consumed.
- If a message fails to be sent to the target, RocketMQ retries the message. The target must be able to process duplicate events. When the retry upper limit is reached, the source message is added to the dead letter queue of the corresponding topic in RocketMQ, and the EG event is not delivered. For details, see **Managing the Dead Letter Queues**

----End

6.1.2.3 DMS for RocketMQ

Add a DMS for RocketMQ event source to an event stream.

Prerequisites

- You have purchased a RocketMQ instance in DMS for RocketMQ.
- When the source is DMS for RocketMQ, only FunctionGraph (function computing) can be selected for the target.

Creating a DMS for RocketMQ Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Click the event source.

Step 5 Configure the event source by referring to **Table 6-3**.

Table 6-3 DMS for RocketMQ parameters

Parameter	Description
Event Provider	Select **DMS for RocketMQ**.
Instance	Select an instance.
Group	Enter a group ID.
Topic	Enter a topic.
SSL	Specify whether to enable SSL.
ACL	Specify whether to enable ACL. • **When ACL is enabled, you need to configure the username and secret key.** • **Ensure that the ACL status of the created RocketMQ instance is the same as that of the RocketMQ event source.**
tag	Enter one or more tags.
Consumption Timeout (ms)	Enter an integer ranging from 1,000 to 900,000.
Consumption Mode	Select **Concurrently** or **Orderly**.
Consumption Threads	Enter an integer ranging from 20 to 64.
Max. Messages	Enter an integer ranging from 1 to 32.
Max. Retries	Enter the maximum number of retries. • **–1**: unlimited retry; **0**: no retry. • **If the RocketMQ version is 4.x, each message can be retried for a maximum of 16 times by default. Table 6-4 describes the retry interval.**

Parameter	Description
Retry Interval (ms)	Enter an integer ranging from 1,000 to 30,000.

Table 6-4 RocketMQ 4.x retry interval

Retry	Interval	Retry	Interval
1	10 seconds	9	7 minutes
2	30 seconds	10	8 minutes
3	1 minute	11	9 minutes
4	2 minutes	12	10 minutes
5	3 minutes	13	20 minutes
6	4 minutes	14	30 minutes
7	5 minutes	15	1 hour
8	6 minutes	16	2 hours

Table 6-5 Combination of production and consumption sequence

Production Sequence	Consumption Sequence	Effect
Orderly (message group)	Orderly	Messages preserve exact send order during consumption within a group.
Orderly (message group)	Concurrently	Messages are processed as close to order as possible.
Disorderly (without message group)	Orderly	Consumption follows Apache RocketMQ storage queue order which may differ from send order.
Disorderly (without message group)	Concurrently	Messages are processed as close to order as possible.

Step 6 Click **Next**. The rule configuration page is displayed. For details about how to configure rules, see **Filter Rule Parameters**.

Step 7 Click **Next** to complete the rule configuration. You can continue to configure the event target of FunctionGraph by referring to **Routing to FunctionGraph**.

NOTE

When **Event Source** is set to **DMS for RocketMQ** and **Event Target** is set to **FunctionGraph**, **Execute** can be set to **Synchronously** or **Asynchronously**.

Step 8 After the event source and target are configured, click **Save** in the upper right.

NOTE

- The MQ collection function takes effect in minutes after it is started for the first time.
- In broadcast mode, retry upon consumption failure is not supported. That is, if a message fails to be consumed, the message will not be retried and the consumer continues to consume new messages.
- If a message fails to be sent to the target, RocketMQ retries the message. The target must be able to process duplicate events. When the retry upper limit is reached, the source message is added to the dead letter queue of the corresponding topic in RocketMQ, and the EG event is not delivered.

----End

6.1.3 Event Rule

By default, messages are transparently transmitted to the target. Rule configuration is supported now.

An event rule transforms CloudEvents-compliant events before they are delivered to targets.

Event streams support event filtering and transformation. For details about the rules, see **Event Rules**.

6.1.4 Event Target

6.1.4.1 Routing to FunctionGraph

Configure FunctionGraph as the event target when creating an event stream.

Prerequisites

You have enabled FunctionGraph and created a function as the event target.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Configure the event source by referring to **Distributed Message Service (DMS) for Kafka**.

Step 5 Configure the event target.

1. Click **Event Target**.

2. Select **FunctionGraph (function computing)** for **Target**.
3. Set event target parameters.

Figure 6-2 Event target - FunctionGraph

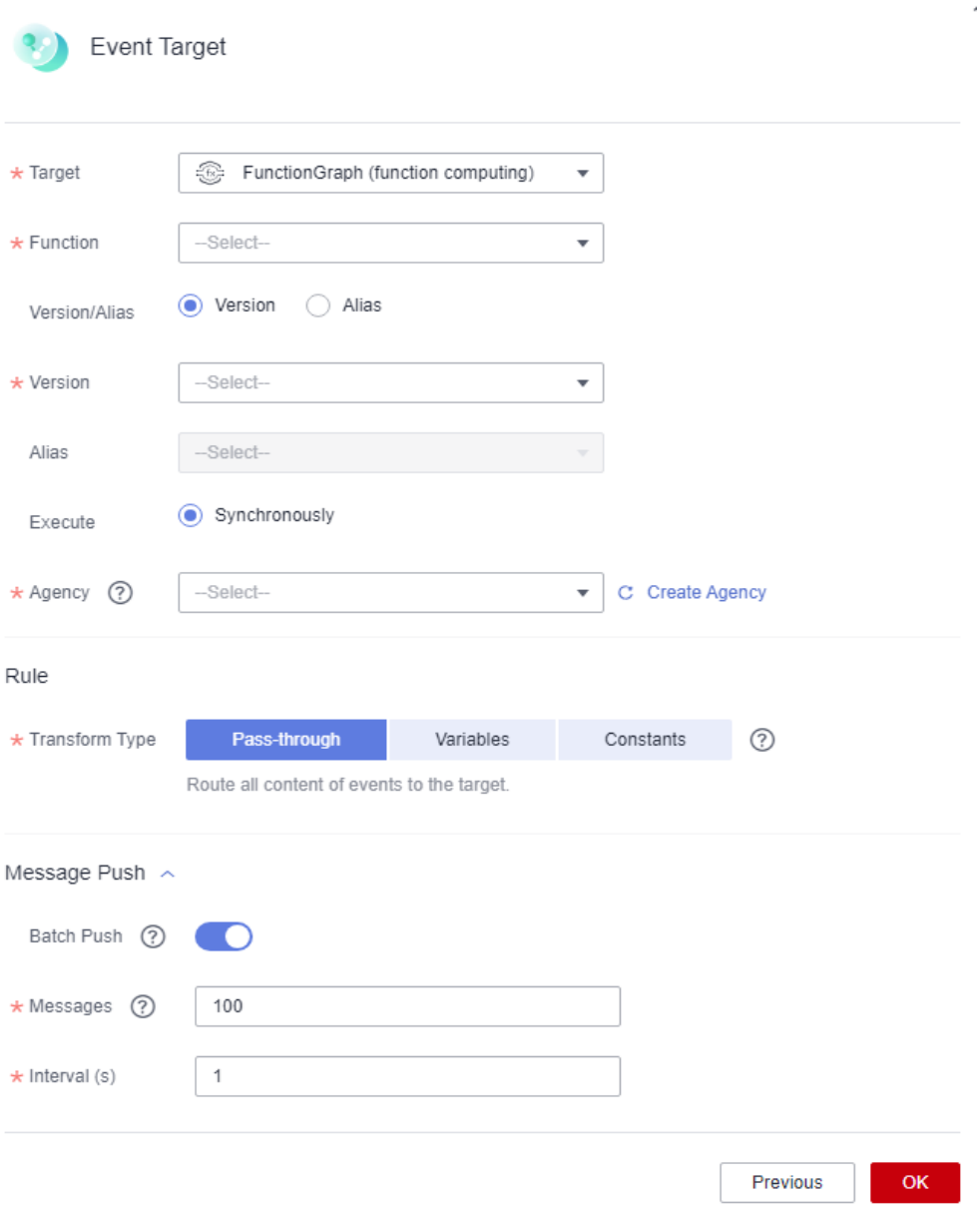


Table 6-6 FunctionGraph (function computing) parameters

Parameter	Description
Function	Select the function to trigger. If no function is available, **create a function** first.
Version/Alias	Choose to specify a version or alias.
Version	Select a function version. Default: **latest**.
Alias	Select a function alias.

Parameter	Description
Execute	Default: **Synchronously**.
Agency	Select an agency. If no agency is available, click **Create Agency** to generate one named **EG_INVOKE_FG_AGENCY**. – Only agencies with EG as the delegated cloud service are displayed. – Select an agency with the permission **functiongraph:function:invoke***.
Rule	
Transform Type	EG transforms CloudEvents-compliant events for targets. The following three types are supported: – **Pass-through**: Route the complete structure of native events directly to the target. – **Variables**: Route only parameters extracted from events with JSONPath to the target. – **Constants**: Route only constants in events to the target as a trigger. For more information about the transform types, see **Event Content Transformation**.
Message Push	
Batch Push	Specify whether to enable batch push to aggregate multiple events.
Messages	The maximum number of aggregated records that can be pushed at a time. Default: **100**. Range: 1–10,000. This parameter is available only when **Batch Push** is enabled.
Interval (s)	The interval between batch pushes, in seconds. Default: **1**. Range: 0–15. This parameter is available only when **Batch Push** is enabled.

Step 6 Click **OK**.

Step 7 After the event source and target are configured, click **Save** in the upper right.

----End

6.1.4.2 Routing to DMS for Kafka

Configure DMS for Kafka as the event target when creating an event stream.

Constraints

Custom certificates are not supported in the event target DMS for Kafka.

Prerequisites

You have enabled DMS for Kafka as the event target.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Configure the event source by referring to **Distributed Message Service (DMS) for Kafka**.

Step 5 Configure the event target.

1. Click **Event Target**.
2. Select **Distributed Message Service (DMS) for Kafka** for **Target**.
3. Set event target parameters.

Figure 6-3 Distributed Message Service (DMS) for Kafka

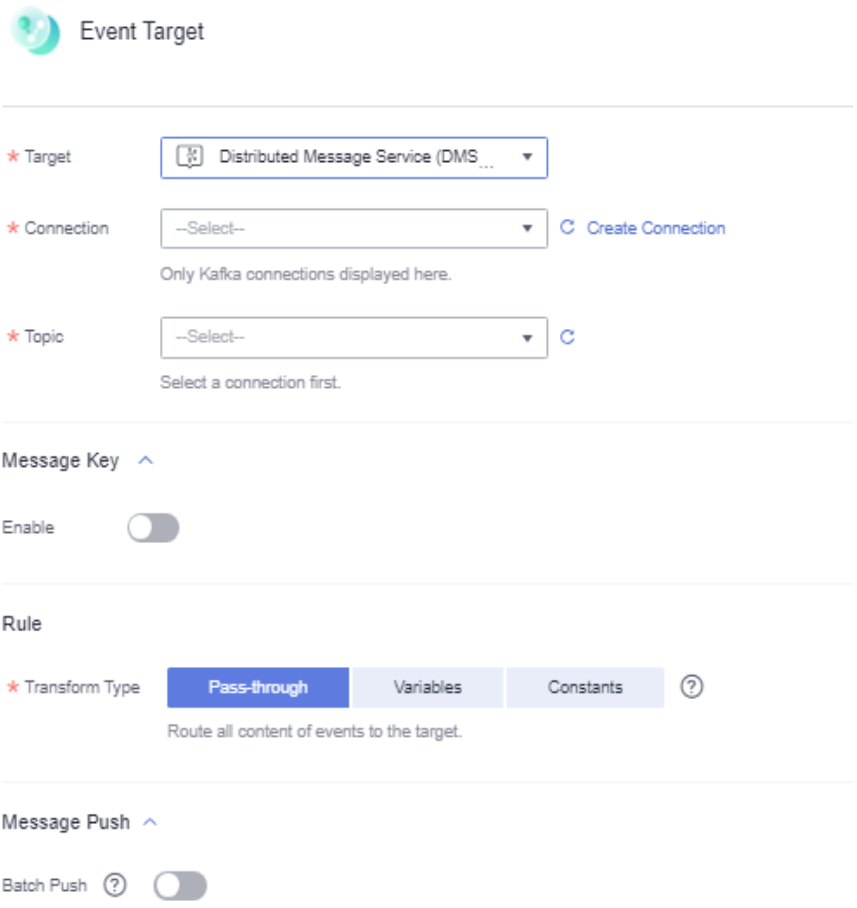


Table 6-7 Distributed Message Service (DMS) for Kafka parameters

Parameter	Description
Connection	Select a connection. If no connection is available, create a Kafka connection first. For details, see **Connections**.
Topic	First select a connection, and then select a topic.
Message Key	
Disable	Do not use a message key.
Enable	**Variable**: The key is a variable value from CloudEvents-compliant events. **Constant**: The key is a specified constant. All messages will be sent to the same partition.
Rule	
Transform Type	EG transforms CloudEvents-compliant events for targets. The following three types are supported: – **Pass-through**: Route the complete structure of native events directly to the target. – **Variables**: Route only parameters extracted from events with JSONPath to the target. – **Constants**: Route only constants in events to the target as a trigger. For more information about the transform types, see **Event Content Transformation**.
Message Push	
Batch Push	Specify whether to enable batch push to aggregate multiple events.
Messages Interval (s)	The maximum number of aggregated records that can be pushed at a time. Default: **100**. Range: 1–10,000. This parameter is available only when **Batch Push** is enabled. The interval between batch pushes, in seconds. Default: **1**. Range: 0–15. This parameter is available only when **Batch Push** is enabled.

----End

6.1.4.3 Routing to OBS

Configure OBS as the event target when creating an event stream.

Prerequisites

- You have enabled OBS as the event target.
- Select **DMS for RocketMQ** as the event source.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Serverless Event Streams**.

Step 3 Click **Create Serverless Event Stream**, click the icon in the upper left corner, enter the event stream name and description, and click **OK**.

Step 4 Configure the event source by referring to **Distributed Message Service (DMS) for Kafka**.

Step 5 Configure the event target.

1. Click **Event Target**.
2. Select **Object Storage Service (OBS)** as the target.
3. Set event target parameters.

Table 6-8 OBS parameters

Parameter	Description
AK	Enter an AK. **NOTE** For details about how to obtain an AK/SK, see **How Do I Obtain an Access Key (AK/SK)?**
SK	Enter an SK.
Bucket	Select or enter an OBS bucket name.
Dumping Directory	Enter a directory **of an object in the bucket. Use slashes (/) to separate different directories.**
Time Directory Format	Select a format. **NOTE** Data is saved to a hierarchical time directory in the dumping directory. For example, if the time directory is accurate to day, the directory will be in the format of *bucket name/file directory/year/month/day*.
Rule	

Parameter	Description
Transform Type	EG transforms CloudEvents-compliant events for targets. The following three types are supported: – **Pass-through**: Route the complete structure of native events directly to the target. – **Variables**: Route only parameters extracted from events with JSONPath to the target. – **Constants**: Route only constants in events to the target as a trigger. For more information about the transform types, see **Event Content Transformation**.
Message Push	
Batch Push	Specify whether to enable batch push to aggregate multiple events.
Messages Interval (s)	The maximum number of aggregated records that can be pushed at a time. Default: **100**. Range: 1–10,000. This parameter is available only when **Batch Push** is enabled. The interval between batch pushes, in seconds. Default: **1**. Range: 0–15. This parameter is available only when **Batch Push** is enabled.

----End

6.1.5 Serverless Event Stream Management

6.1.5.1 Creating an Event Stream

Create an event stream on the EG console.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click **Create Event Stream**.

Step 4 Enter an event stream name and description, and click **OK**.

Step 5 Configure the **event source**.

1. Click **Event Source**.
2. Select an event source provider.
3. Set event source parameters.
4. Click **Next**.

Step 6 Configure rules.

1. Click Rule. The Rule dialog box is displayed on the right.
2. Configure the rule pattern content.
3. Click **Next**.

Step 7 Configure the **event target**.

1. Click **Event Target**.
2. Select a target service.
3. Set event target parameters.
4. Click **OK**.

Step 8 Click **Save**.

The event stream is disabled by default once created.

----End

6.1.5.2 Editing an Event Stream

Modify the name, description, status, event source, and event target of an event stream.

Modifying the Name and Description

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click **Configure** in the row that contains the desired event stream to go to the details page.

Step 4 Click the edit icon next to the default event stream name.

Step 5 Modify the name and description and click **OK**.

----End

Changing the Status

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click **Disable** or **Enable** in the row that contains the desired event stream.

----End

Modifying the Event Source

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click the name of the desired event stream to go to the details page.

Step 4 Click the event source card.

Step 5 Modify the **event source** parameters.

----End

Modifying the Event Target

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click the name of the desired event stream to go to the details page.

Step 4 Click the event target card.

Step 5 Modify the **event target** parameters.

----End

6.1.5.3 Deleting an Event Stream

Delete an event stream that will no longer be used.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams**.

Step 3 Click **Delete** in the row that contains the desired event stream.

Step 4 Click **OK**.

----End

6.1.5.4 RocketMQ Collection Function Error Codes

Error Code	Description	O&M Description	Suggestion
200	The heartbeat is successful.	Set the event stream status to RUNNING and clear the alarm.	This is a normal system response.
601	Unknown error.	Set the event stream status to ERROR and report an alarm. The event stream automatically restarts.	Contact Huawei technical support.
602	Network error.		
502	The consumer does not exist.		
401	Target delivery authentication failed.	Refresh the token and set the event stream status to ERROR. The event stream automatically restarts.	Wait for automatic recovery.

Error Code	Description	O&M Description	Suggestion
600	Upgrading	No action is required.	The collection function is being upgraded. Wait for the upgrade.
403	The target function is disabled.	Set the event stream status to ERROR and report an alarm.	Check whether the function is normal.
516	The topic does not exist.		Check topics.
510	RocketMQ ACL authentication failed.		Check whether the user password is changed. If the password is correct, contact Huawei technical support.

6.1.6 Monitoring

6.1.6.1 Viewing Monitoring Data

Scenario

Cloud Eye monitors event stream metrics in real time. You can view these metrics on the Cloud Eye console.

Prerequisites

You have created an event stream.

Procedure

Step 1 Log in to the **EG console**.

Step 2 Choose **Serverless Event Streams**.

Step 3 Click **Monitoring** next to the event stream name to go to the monitoring page. Data of all event streams in the last hour is displayed by default.

You can also click **1h**, **3h**, **12h**, **1d**, **7d**, or **30d** to view event stream data in different periods.

NOTE

- The time range can be customized.
- If you enable **Auto Refresh**, the metric data is refreshed every 5 seconds.
- Click **View details** to go to the Cloud Eye console.
- If you set **Period** to **Raw data**, the raw monitoring data is displayed. If you set **Period** to a specific time, you can select different aggregation methods, including **Avg.**, **Max.**, **Min.**, **Sum**, and **Variance**.

----End

6.1.6.2 Supported Metrics

Introduction

This section describes the event stream metrics and dimensions reported to Cloud Eye. You can search metrics and alarms on the Cloud Eye console or on the monitoring page of EG.

Metrics

Table 6-9 Metric description

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
streaming_process_num	Event Processes	Number of times event processing is attempted.	≥ 0	Event stream	1
streaming_success_num	Successful Processes	Number of times events are actually processed.	≥ 0	Event stream	1
streaming_success_rate	Success Rate	Percentage of total processing attempts that are successful.	0%–100%	Event stream	1

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
streaming_failed_num	Failed Processes	Number of times events could not be processed.	≥ 0	Event stream	1
streaming_failed_rate	Failure Rate	Percentage of total processing attempts that failed.	0%–100%	Event stream	1
streaming_process_time	Processing Time	Average time spent processing an event.	≥ 0 ms	Event stream	1

Table 6-10 Dimension description

Dimension	Key	Value
Event stream	streaming_id	Event stream ID

6.1.6.3 Configuring Alarm Rules

This section describes the alarm policies of some metrics and how to configure them. In actual services, you are advised to configure alarm rules for metrics based on the following alarm policies.

Table 6-11 Parameters for alarm settings

Parameter	Description
Name	Name of the alarm rule. The system generates a name randomly but you can change it.
Description	Alarm rule description. This parameter is optional.
Alarm Type	Alarm type to which the alarm rule applies. Default: **Metric**.
Resource Type	Resource type. Default: **EventGrid**.

Parameter	Description
Dimension	Alarm dimension. Default: **Event Streams**.
Monitoring Scope	Resources to monitor. Default: **Specific resources**.
Monitored Objects	Object to monitor. Default: event stream name.
Method	Alarm triggering method. Default: **Configure manually**.
Alarm Policy	Policy that triggers an alarm. For details, see **Table 4-7**. **If a metric alarm policy is created on the EG page, you cannot modify or add other metric alarm policies.**
Alarm Notification	After you enable this function and configure required parameters, you will be notified of alarms and alarm clearance by notification group or topic subscription.
Notification Recipient	Select **Notification group** or **Topic subscription**.
Notification Group	Select a notification group. If no notification group is available, create one by referring to **Creating a Notification Object or Notification Group**.
Notification Object	Select a notification contact and topic. If no topic is available, create one by referring to **Creating a Notification Object or Notification Group**.
Notification Window	Alarm notifications are only sent during the configured validity period.
Trigger Condition	Condition for triggering a notification.
Enterprise Project	Enterprise project to which the alarm rule belongs. For details, see **Creating an Enterprise Project**.

Procedure

Step 1 Log in to the **EG console**.

Step 2 Click ☰ in the upper left and choose **Middleware** > **EventGrid**.

Step 3 Choose **Event Streams**.

Step 4 Click **Monitoring** in the **Operation** column to go to the monitoring page.

Step 5 Hover over a metric and click + to create an alarm rule for it.

Step 6 Specify the alarm rule details.

For details about how to create an alarm rule, see **Creating an Alarm Rule**.

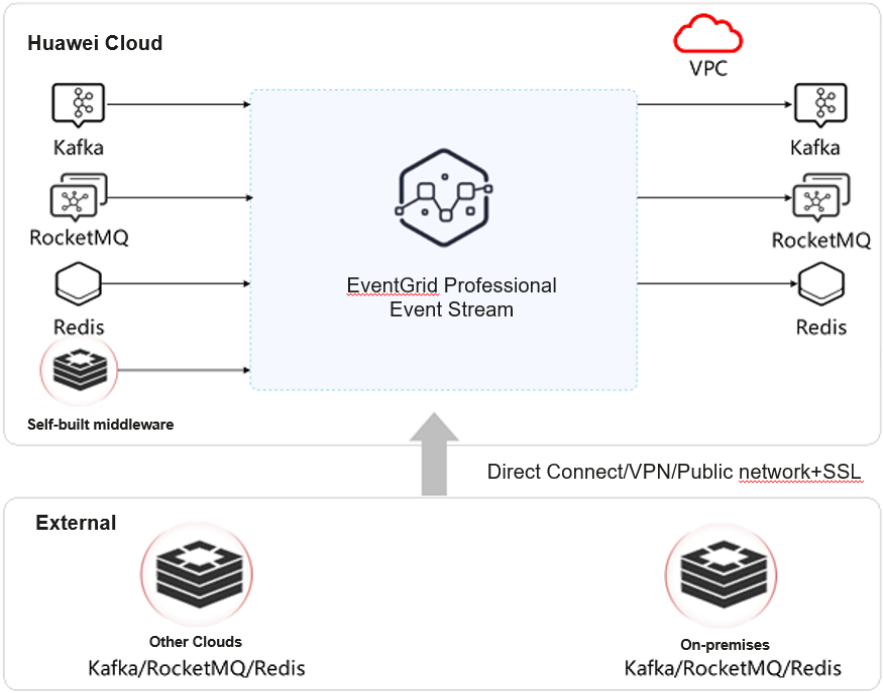
----End

6.2 Professional Event Streams

6.2.1 Overview

Professional event streams provide easy-to-use, stable, and efficient data pipelines for real-time synchronization across systems. They make data flow between middleware simple, greatly reducing data transfer costs. It is suitable for middleware data migration, backup, and disaster recovery in cloud migration, cross-cloud, and cross-region scenarios.

Figure 6-4 Professional event stream architecture



NOTICE

Currently, only the **AP-Singapore** region is supported.

Feature Overview

1. **Sync guide**
   Provides guide for data synchronization using professional event streams.
2. **Scenario-based selection**
   Preconfigured use cases make data synchronization simpler and easier.
3. **Network security**
   Supports SSL and other encryption and authentication mechanisms.
4. **Multiple sync modes**
   Supports object selection, full and incremental sync, and multiple compression modes.
5. **Precheck**
   Validate sync feasibility with actionable diagnostics.
6. **Monitoring**
   Provides full observability with flow control.

6.2.2 Advantages

Stable

High performance and reliability

Data Consistency

Pre-check and consistency check

Multi-dimensional Monitoring and Easy O&M

Fault location and recovery

Data Processing

Data processing (ETL) during synchronization

Lower Costs

Various specifications and pay-per-use billing

Lower Development Costs

Out-of-the-box, rich ecosystem, and codeless interconnection

6.2.3 Scenarios

Migration

Data can be synchronized across cloud platforms, from on-premises to the cloud, or across regions on the cloud. This mode is applicable to short-term unidirectional data sync to cloud during service cutover.

Benefits: Database can be synchronized without affecting services via incremental sync.

Synchronization

Real-time data sync between data sources can be used as the atomic capability of the MAS DR solution to implement DR between middleware across regions and clouds, and between on-premises and clouds. Long-term unidirectional data synchronization and DR switchover are supported.

Benefits: Remote transmission optimization and DR features are provided, which are different from the simple data synchronization solution in the industry.

ETL

Achieves continuous real-time data flow of key services between different systems. Data processing, heterogeneous sync, and long-term unidirectional sync are supported.

6.2.4 Professional Event Stream Clusters

Constraints

To create a professional event stream cluster, you need to submit a service ticket to apply for the whitelist.

Creating a Professional Event Stream Cluster

Step 1 **Logging In to the EG Console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Clusters**.

Step 3 Click **Buy Cluster** in the upper right corner. On the displayed page, configure the following information:

1. Basic info:
   a. **Billing Mode**: **Pay-per-use** is used by default.
   b. **Cluster Name**: Start with a letter or digit, and only include letters, digits, periods (.), underscores (_), and hyphens (-). (Max. 128 characters)
   c. **Description**: Enter a description.
2. Source and target databases:
   a. **Source Database**: Select **Kafka**, **RocketMQ**, or **DCS**.
   b. **Target Database**: Select **Kafka**, **RocketMQ**, or **DCS**.
   c. **ECUs**: The default value is **10 ECU**.

      NOTE

      10 ECUs support 1 to 10 concurrent jobs with 400,000 QPS for Kafka, 30,000 QPS for RocketMQ, and 100,000 QPS for DCS.
3. Network configuration:

a. VPC: Select the created VPC. For details about how to create a VPC, see **Creating a VPC**.

b. Subnet: Select the created subnet. For details about how to create a subnet, see **Creating a Subnet for a VPC**.

Ensure that the VPC and subnet configured for the event stream cluster can communicate with the VPC and subnet configured for the source and target instances of Kafka, RocketMQ, or DCS. Otherwise, the connectivity test fails.

Step 4 After the parameters are configured, click **Confirm**.

----End

6.2.5 Professional Event Stream Jobs

6.2.5.1 Creating a Professional Event Stream Job

6.2.5.1.1 Kafka-to-Kafka Data Synchronization

Constraints

- If the source is a Kafka instance:
  - The source and target can only be DMS Kafka instances. The instance versions must be the same. Currently, 2.7 and 3.x are supported.
  - The number of brokers, broker CPU, memory, and storage space of the source Kafka instance must be the same as those of the target Kafka instance.
  - A topic occupying one partition will be created for both the source and target Kafka instances. Ensure sufficient partitions are available in source and target when creating the event stream job.
- If the source is a Kafka address:
  - The target Kafka instance must be a DMS Kafka instance of version 2.7 or 3.x. The source Kafka instance must be a DMS Kafka instance of version 2.7 or 3.x, or a cloud vendor's or self-built Kafka instance that is compatible with open-source Kafka 2.7 or later.
  - The target Kafka must maintain the same broker count as the source. Its broker CPU, memory, storage space, and topics must be greater than or equal to the source's specifications, while the partition count must be at least equal to the source's partitions plus two.
  - The Kafka sync job creates two topics occupying two partitions total in the target Kafka. Ensure sufficient partitions exist before creating the event stream job.

Prerequisites

- The source and target Kafka instances are available.
- Ensure that the VPC, subnet, and Kafka topic partitions and storage space at the source and target are sufficient.

NOTE

For details about how to create a VPC and subnet, see **Creating a VPC**.

- Ensure that the source and target instances can communicate with the VPC selected when you create the event stream cluster.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Create Job** in the upper right corner. The **Basic Settings** page is displayed.

Step 4 Configure the basic settings. The following uses a non-whitelisted account as an example.

Table 6-12 Basic information parameters

Parameter	Description
Cluster	Select a created cluster. If no cluster is created, create one by referring to **Professional Event Stream Clusters**.
Job Name	Enter a job name.
Scenario	**Sync** is selected by default. Data across sources can be synchronized in real time.
Description	Enter the description of the job.

Step 5 Click **Next: Configure Source and Target**. The **Configure Source and Target** page is displayed.

Table 6-13 Parameters for configuring source and target data

Parameter	Description
Type	Select the configuration type. Options: **Kafka instance** and **Kafka address**. The target data type is **Kafka instance** by default.
Instance Alias	Enter an instance alias, which **identifies the source and target instance. You are advised to set only one alias for the same source or target instance.**

Parameter	Description
Kafka Addresses	Required when **Type** is set to **Kafka address**. Enter Kafka addresses.
Region	Select a region.
Project	Select a project.
Instance	Select a Kafka instance.
Access Mode	**Plaintext** or **Ciphertext**
Security Protocol	• If **Plaintext** access mode is selected, the security protocol is **PLAINTEXT**. • If you select **Ciphertext** access mode, the security protocol can be **SASL_SSL** or **SASL_PLAINTEXT**.
Authentication Mechanism	This parameter is mandatory when **Access Mode** is set to **Ciphertext**. The authentication mechanism can be **SCRAM-SHA-512** or **PLAIN**.
Username	This parameter is mandatory when **Access Mode** is set to **Ciphertext**. Enter a username.
Password	This parameter is mandatory when **Access Mode** is set to **Ciphertext**. Enter a password.

Step 6 Click **Test Connectivity**. After confirming that the instance connectivity of the source and target is normal, click **Next: Advanced Settings**. The **Advanced Settings** page is displayed.

Figure 6-5 Advanced settings

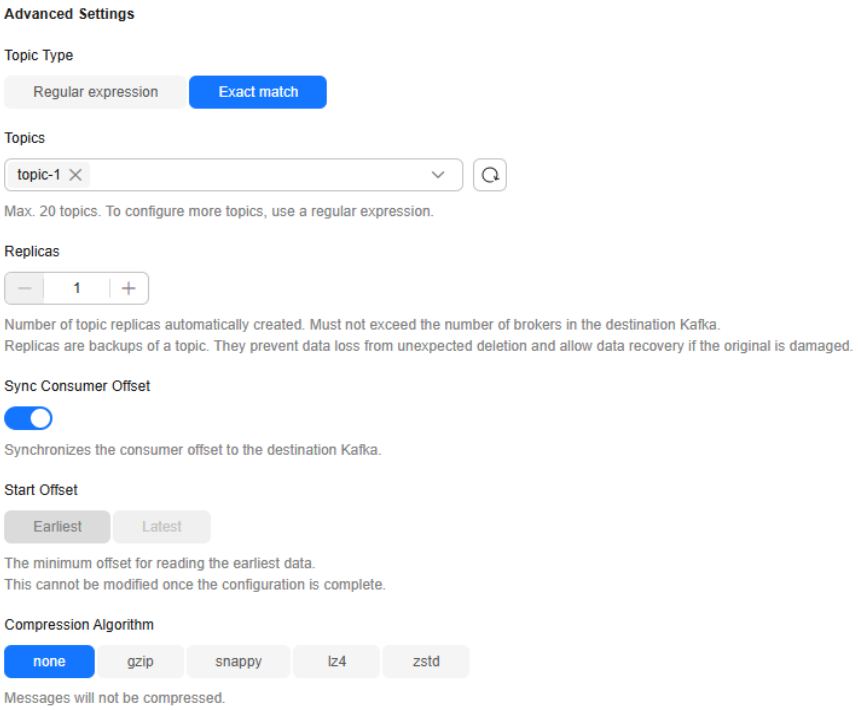


Table 6-14 Configuration parameters

Parameter	Description
Topic Type	Select **Regular expression** or **Exact match**. **NOTE** • If you select **Regular expression**, enter a regular expression in the **Topics (Regular)** text box. For example, **.*** indicates that all topics are matched, and **topic.*** indicates that all topics with the topic prefix are matched. • If you select **Exact match**, you need to select topics.
Replicas	Set the number of replicas. **The number of replicas of the automatically created topic cannot exceed the number of brokers in the target Kafka.**
Sync Consumer Offset	Select whether to enable this function. **If this function is enabled, the consumer offset will be synchronized to the target Kafka.**
Start Offset	Select **Earliest** or **Latest**.

Parameter	Description
Compression Algorithm	Select **none**, **gzip**, **snappy**, **lz4**, or **zstd** as the compression algorithm.

Step 7 Click **Next: Pre-check**. On the displayed page, click **Finish**.

Step 8 Return to the professional event stream job list and click the name of the created event stream. Click **Job Management** to view the synchronization details.

Table 6-15 Parameter description

Parameter	Description
Topic	Topic created when a Kafka instance is created.
Partitions	Number of partitions set when a topic is created. The larger the number of partitions, the higher the consumption concurrency.
Messages to Sync	Number of messages that have not been synchronized in the current topic partition.

NOTE

Sync Rate: Rate at which messages are synchronized in the current job. You can click **Limit Rate** to configure flow control for the source Kafka instance.

----End

6.2.5.1.2 RocketMQ-to-RocketMQ Data Synchronization

Constraints

- RocketMQ stream synchronization supports only normal and ordered messages. Messages of other types will not be synchronized.
- If the source is a RocketMQ instance:
  - The source and target must be DMS RocketMQ instances of the same version. The supported versions are 4.8.0 and 5.x.
  - The broker quantity, broker flavor, and storage space of the target RocketMQ instance must be the same as those of the source RocketMQ instance.
  - The instance type (for example, single-node or cluster) of the target RocketMQ instance must be the same as that of the source RocketMQ instance.
- If the source is a RocketMQ address:

- The source RocketMQ must support the query of cluster information and topic list. Otherwise, the online synchronization will fail.
- The target must be a DMS RocketMQ instance of version 4.8.0 or 5.x. The source must be a DMS RocketMQ instance of version 4.8.0 or 5.x, or a cloud vendor's or self-built RocketMQ instance that is compatible with open-source RocketMQ 4.x or 5.x.
- The number of brokers of the target RocketMQ instance must be the same as that of the source RocketMQ instance. The broker specifications, number of queues, and storage space of the target RocketMQ instance must be greater than or equal to those of the source RocketMQ instance.

Prerequisites

- The source and target RocketMQ instances are available.
- Ensure that the VPC, subnet, source and target RocketMQ topic queues, and storage space are sufficient.
- Ensure that the source and target instances can communicate with the VPC selected when you create the event stream cluster.

NOTE

For details about how to create a VPC and subnet, see **Creating a VPC**.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Create Job** in the upper right corner. The **Basic Settings** page is displayed.

Step 4 Configure the basic settings. The following uses a non-whitelisted account as an example.

Table 6-16 Basic information parameters

Parameter	Description
Cluster	Select a created cluster. If no cluster is created, create one by referring to **Professional Event Stream Clusters**.
Job Name	Enter a job name.
Scenario	**Sync** is selected by default. Data across sources can be synchronized in real time.
Description	Enter the description of the job.

Step 5 Click **Next: Configure Source and Target**. The **Configure Source and Target** page is displayed.

Table 6-17 Parameters for configuring source and target data

Parameter	Description
Type	Select the configuration type. Options: **RocketMQ instance** and **RocketMQ address**. The target data type is **RocketMQ instance** by default.
Instance Alias	Enter an instance alias, which **identifies the source and target instance. You are advised to set only one alias for the same source or target instance.**
Region	Select a region.
Project	Select a project.
Instance	Select a RocketMQ instance.
Username	Enter a username.
Secret Key	Enter a secret key.
NameServer Address	When **Type** is set to **RocketMQ address**, this parameter must be set in the source data. Enter the NameServer addresses.
Broker Address	When **Type** is set to **RocketMQ address**, this parameter must be set in the source data. You can click **Auto Obtain** to obtain the broker addresses if you have entered the NameServer addresses. Note: This feature requires cluster information query in the source. If it fails, manually enter the broker addresses.
SSL	When **Type** is set to **RocketMQ address**, this parameter must be set in the source data. Set whether SSL is enabled.
ACL	When **Type** is set to **RocketMQ address**, this parameter must be set in the source data. Indicates whether access control is enabled.

Step 6 Click **Test Connectivity**. After confirming that the instance connectivity of the source and target is normal, click **Next: Advanced Settings**. The **Advanced Settings** page is displayed.

Figure 6-6 Advanced settings

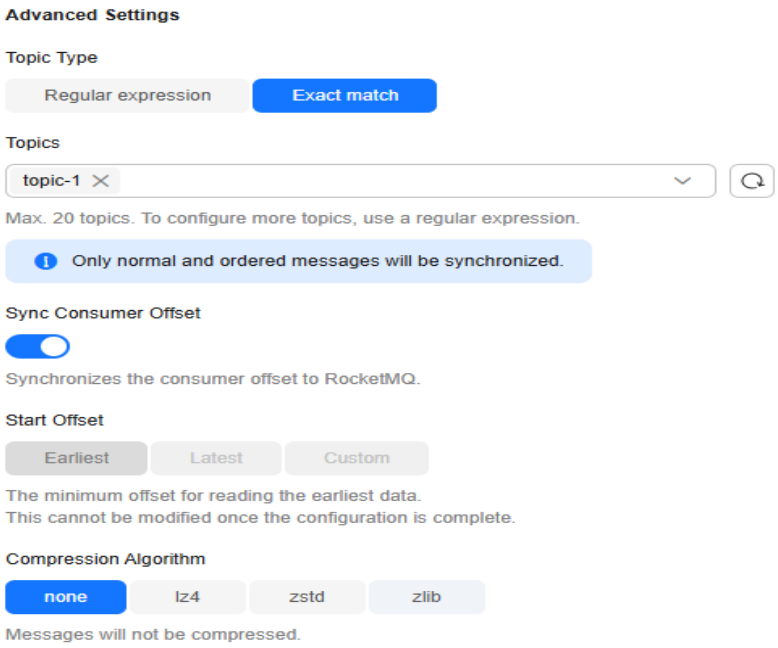


Table 6-18 Configuration parameters

Parameter	Description
Topic Type	If **Type** is set to **RocketMQ instance**, select **Regular expression** or **Exact match**. If **Type** is set to **RocketMQ address**, select **Regular expression** or **Enter**. **NOTE** • If you select **Regular expression**, enter a regular expression in the **Topics (Regular)** text box. For example, **.*** indicates that all topics are matched, and **topic.*** indicates that all topics with the topic prefix are matched. • If you select **Exact match**, you need to select topics. • If you select **Enter**, you need to enter topics.

Parameter	Description
Sync Consumer Offset	Select whether to enable this function. **NOTE** If this function is enabled, the consumer offset will be synchronized to the target RocketMQ.
Start Offset	Select **Earliest**, **Latest**, or **Custom**.
Compression Algorithm	Select **none**, **lz4**, **zstd**, or **zlib** as the compression algorithm.

Step 7 Click **Next: Pre-check**. On the displayed page, click **Finish**.

Step 8 Return to the professional event stream job list and click the name of the created event stream. Click **Job Management** to view the synchronization details.

Table 6-19 Parameter description

Parameter	Description
Topic	Topic created when a RocketMQ instance is created.
Queues	Number of queues set when a topic is created.
Messages to Sync	Number of unsynchronized messages in the topic queues.

----End

6.2.5.1.3 DCS-to-DCS Data Synchronization

Constraints

- You need to submit a service ticket to enable the whitelist for DCS professional event stream jobs.
- If the source is a DCS instance:
  - Only single-node, master/standby, and Redis Cluster Redis 5.0 instances are supported.
  - The source and target Redis instances must have the same specifications, type, and storage space.
  - Enable **eventLog** and disable **appendonly** (unavailable for single-node Redis instances) for the source and target Redis instances.
  - Disable the client IP pass-through function for the source and target Redis instances.
- If the source is a DCS address:

- The source must be a single-node, master/standby, or Redis Cluster instance of Redis 4.0, 5.0, or 6.0. The target must be a single-node, master/standby, or Redis Cluster instance of DCS Redis 5.0.
- If the **SYNC** and **PSYNC** commands are disabled by the source instance, enable them before synchronizing data. Otherwise, the synchronization fails.
- If the source is a DCS Redis instance, set **eventlog** to **no** for both the source and target.
- You are advised to set **repl-timeout** to **300** seconds and **client-output-buffer-limit** to 20% of the maximum memory of the source Redis instance.
- The source and target Redis instances with SSL enabled do not support data synchronization. Disable SSL first.
- If the source and target Redis instances use passwords containing single quotation marks ('), the synchronization will fail. In this case, change the passwords.
- Ensure that the target has enough databases to match the source's highest database index that contains data. For instance, if the source databases range from DB0 to DB127 and the highest database with data is DB99, the target should have at least 100 databases.

Prerequisites

- You have created source and target Redis instances.
- Ensure that DCS, VPC, and subnet resources are sufficient. For details about how to create a VPC and subnet, see **Creating a VPC**.
- Ensure that no command has been written to the target Redis instance.
- Ensure that the source and target Redis instances can communicate with EG.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Create Job** in the upper right corner. The **Basic Settings** page is displayed.

Step 4 Set the basic job information.

Table 6-20 Basic job configuration parameters

Parameter	Description
Cluster	Select a created cluster. If no cluster is created, create one by referring to **Professional Event Stream Clusters**.
Job Name	Enter a job name.

Parameter	Description
Scenario	**Sync** is selected by default. Data across sources can be synchronized in real time.
Description	Enter the description of the job.

Step 5 Click **Next: Configure Source and Target**. The **Configure Source and Target** page is displayed.

Table 6-21 Parameters for configuring source and target data

Parameter	Description
Type	Select the configuration type. Options: **DCS instance** and **DCS address**. The target data type is **DCS instance** by default.
Region	Select a region.
Project	Select a project.
Instance	Select a DCS instance.
Instance Type	When **Type** is set to **DCS address**, this parameter must be set in the source data. Select the instance type. • Single-node • Master/Standby • Redis Cluster
DCS Addresses	When **Type** is set to **DCS address**, this parameter must be set in the source data. Enter one or more DCS addresses.
Password Protected	When **Type** is set to **DCS address**, this parameter must be set in the source data. Select whether to use password. • Yes • No
Username	When **Type** is set to **DCS address**, this parameter must be set in the source data. Enter a username.

Parameter	Description
Password	Enter the password.

Step 6 Click **Test Connectivity**. After confirming that the instance connectivity of the source and target is normal, click **Next: Advanced Settings**. The **Advanced Settings** page is displayed.

Table 6-22 Parameters for DCS instance

Parameter	Description
Sync Mode	Default: **Full+Incremental**.
Limit Sync Rate	Enabled by default.
Sync Rate (MB/s)	Set the synchronization rate. **Enter a value between 1 and 20.**
Use Slave Node	Enabled by default.

Table 6-23 Parameters for DCS address

Parameter	Description
Sync Mode	Default: **Full+Incremental**.
Retry	Select a retry policy. Instantly: Retry immediately after an error is detected. As scheduled: Retry within the specified time window after an error is detected.
Limit Retry Interval for Connection Failure	This parameter is mandatory when **Retry** is set to **Instantly**. Set retry interval for connection failure. If disabled, retries will continue without stopping.
Retry Interval Upon Connection Failure (Minutes)	This parameter is mandatory when **Retry** is set to **Instantly**. Configure the retry interval after a connection failure.
Limit Retry Interval for Other Failures	This parameter is mandatory when **Retry** is set to **Instantly**. Set retry interval for other failures. If disabled, retries will continue without stopping.

Parameter	Description
Retry Interval Upon Other Failures (Minutes)	This parameter is mandatory when **Retry** is set to **Instantly**. Configure the retry interval after other failures occur.
Retry Start Time for Connection Failure	This parameter is mandatory when **Retry** is set to **As scheduled**. Configure the start time of the retry after the connection fails.
Retry End Time for Connection Failure	This parameter is mandatory when **Retry** is set to **As scheduled**. Configure the end time of the retry after the connection fails.
Retry Start Time for Other Failures	This parameter is mandatory when **Retry** is set to **As scheduled**. Configure the retry start time after other failures occur.
Retry End Time for Other Failures	This parameter is mandatory when **Retry** is set to **As scheduled**. Configure the retry end time after other failures occur.
Use Slave Node	Enabled by default.

NOTE

- If the connection between the event stream cluster and the source or target instance fails after the synchronization task (with DCS addresses as the source) is started, the connection can be retried according to the configured policy. If the source or target instance is reconnected within the configured time, the task is automatically resumed. Otherwise, the synchronization task fails.

  The following are some scenarios where the connection may fail:

  1. The source or target instance is powered off, shut down, or restarted.
  2. The network connection between the event stream cluster and the source or target instance is abnormal.
  3. The source or target node or shard is faulty and cannot be accessed.

- If other failures occur on the source or target instance after the synchronization task (with DCS addresses as the source) is started, the task can be retried according to the configured policy. If the data synchronization is restored within the configured time, the task is automatically resumed. Otherwise, the synchronization task fails.

  The following are some scenarios where non-connection failures may occur:

  1. psync is not enabled on the source.
  2. Data cannot be synchronized due to the source status, for example, the source is restarted and RDB is being loaded.
  3. The source Redis commands fail to be synchronized. For example, the source Redis commands are not supported on the target.
  4. The password of the source or target instance is changed.

Step 7 Click **Next: Pre-check**. On the displayed page, click **Finish**.

Step 8 Return to the professional event stream job list and click the name of the created event stream. Click **Job Management** to view the synchronization details.

Table 6-24 Parameter description

Parameter	Description
Source Node	Address of the DCS instance in the source data.
Target Node	Address of the DCS instance in the target data.
Status	Status of DCS event stream synchronization.
Sync Progress (%)	DCS event stream synchronization progress.
Mode	**Full**, **Incremental**, and **Full +Incremental** synchronization modes are supported.
Source Transactions Executed	Number of source data records.
Target Transactions Executed	Number of synchronized target data records.

----End

6.2.5.2 Deleting a Professional Event Stream Job

This section describes how to delete a professional event stream job on the console.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Delete** in the **Operation** column or select multiple event stream jobs and click **Delete**.

Step 4 In the displayed dialog box, enter **DELETE** or click **Auto Enter**.

Step 5 Click **OK**. If a dialog box is displayed indicating that the professional event stream job is deleted, the professional event stream job is deleted.

----End

6.2.5.3 Enabling a Professional Event Stream Job

This section describes how to enable a professional event stream job on the console.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Enable** in the **Operation** column, or select multiple event stream jobs and click **Change Status** > **Enable**. If a dialog box is displayed, indicating that the event stream job is enabled, the professional event stream job is enabled.

----End

6.2.5.4 Disabling a Professional Event Stream Job

This section describes how to disable a professional event stream job on the console.

Procedure

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Disable** in the **Operation** column, or select multiple event stream jobs and click **Change Status** > **Disable**. If a dialog box is displayed indicating that the event stream jobs are disabled, the professional event stream jobs are disabled.

----End

6.2.5.5 Configuring a Professional Event Stream Job

This section describes how to configure a professional event stream job on the console.

Prerequisite

You can configure an event stream only when it is disabled.

Procedure

The following uses Kafka event stream job as an example.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click **Modify** in the **Operation** column. On the **Basic Settings** page, you can modify only **Job Name** and **Description**.

Step 4 Click **Next: Configure Source and Target**. The **Configure Source and Target** page is displayed.

Table 6-25 Whether the source and target data parameters can be modified

Parameter	Modifiable
Type	No
Instance Alias	No
Region	No
Project	No
Instance	No
Access Mode	Yes. **Plaintext** or **Ciphertext**.
Security Protocol	Yes • If **Plaintext** access mode is selected, the security protocol is **PLAINTEXT**. • If **Ciphertext** is selected, the security protocol can be **SASL_SSL** or **SASL_PLAINTEXT**, and the authentication mechanism can be **SCRAM-SHA-512** or **PLAIN**.
Username	Enter a username. This parameter is mandatory when **Access Mode** is set to **Ciphertext**.
Password	Enter a password. This parameter is mandatory when **Access Mode** is set to **Ciphertext**.

Step 5 Click **Next: Advanced Settings**.

Table 6-26 Whether the job object configuration parameters can be modified

Parameter	Modifiable
Topic Type	Yes. Select **Regular expression** or **Exact match**. **NOTE** • If you select **Regular expression**, enter a regular expression in the **Topics (Regular)** text box. • If you select **Exact match**, you need to select topics.
Replicas	Yes. **The number of replicas of the automatically created topic cannot exceed the number of brokers in the target Kafka.**
Sync Consumer Offset	Yes. **NOTE** **If this function is enabled, the consumer offset will be synchronized to the target Kafka.**
Start Offset	No
Compression Algorithm	Yes. Select **none**, **gzip**, **snappy**, **lz4**, or **zstd** as the compression algorithm.

Step 6 Click **Next: Pre-check**. On the displayed page, click **Finish**.

----End

6.2.5.6 Querying Details About a Professional Event Stream Job

This section describes how to query details about a professional event stream job on the console, including basic information, job management, monitoring metrics, and logs.

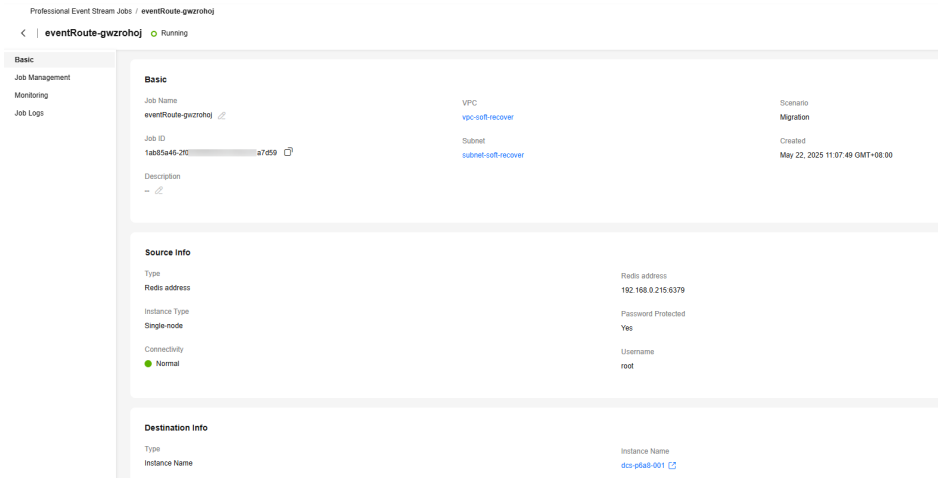
Viewing Basic Information

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click the name of the event stream job to be queried. On the displayed page, view its basic information.

Figure 6-7 Basic information



----End

Viewing Job Management

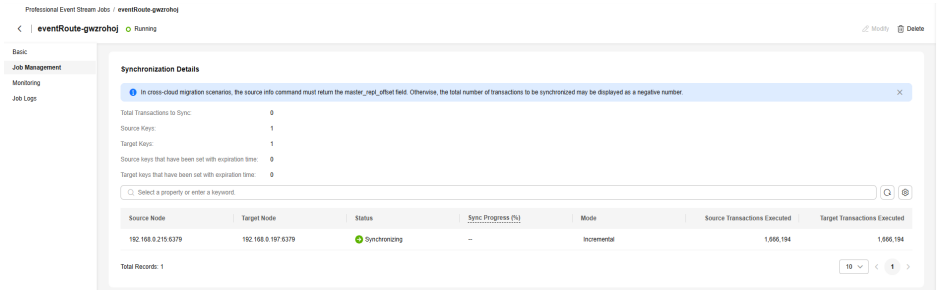
Step 1 Logging In to the EG Console.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click the name of the event stream job to be queried.

Step 4 Click **Job Management** to view job information.

Figure 6-8 Job management



----End

Viewing Monitoring Metrics

Step 1 Logging In to the EG Console.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

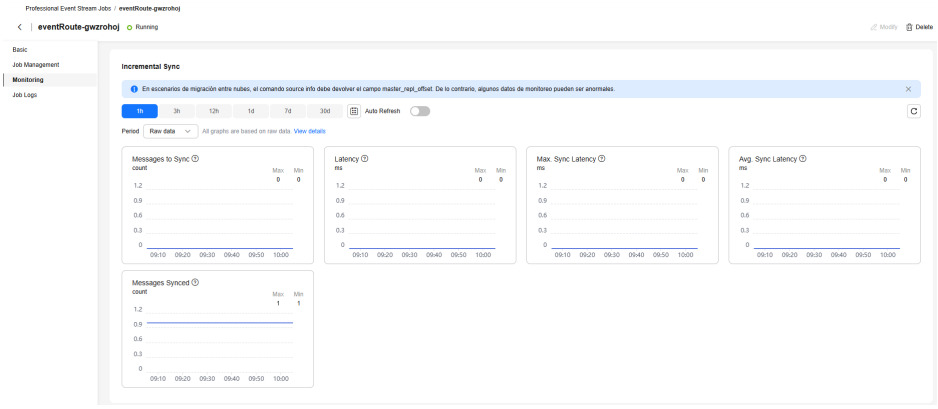
Step 3 Click the name of the event stream job to be queried.

Step 4 Click **Monitoring** to view the monitoring information.

Step 5 Data of all delivered events in the last hour is displayed by default.

You can also click **1h**, **3h**, **12h**, **1d**, **7d**, or **30d** to view event deliveries in different periods.

Figure 6-9 Monitoring metrics



NOTICE

Currently, the metrics **Messages to Sync** and **Messages Synced** in DCS event stream job are calculated independently.

1. **Messages to Sync** indicates the total write operations executed at source.
2. **Messages Synced** which indicates the number of synchronized commands. Full RDB synchronization of existing source data credits only one synced message for the entire dataset transfer, potentially leaving **Messages to Sync** greater than **Messages Synced** after completion.

NOTE

- The time range can be customized.
- If you enable **Auto Refresh**, the metric data is refreshed every 5 seconds.
- Click **View details** to go to the Cloud Eye console.
- If you set **Period** to **Raw data**, the raw monitoring data is displayed. If you set **Period** to a specific time, you can select different aggregation methods, including **Avg.**, **Max.**, **Min.**, **Sum**, and **Variance**.

----End

Viewing Job Logs

Step 1 Logging In to the EG Console.

Step 2 In the navigation pane, choose **Event Streams** > **Professional Event Stream Jobs**.

Step 3 Click the name of the event stream job to be queried.

Step 4 Click **Job Logs**.

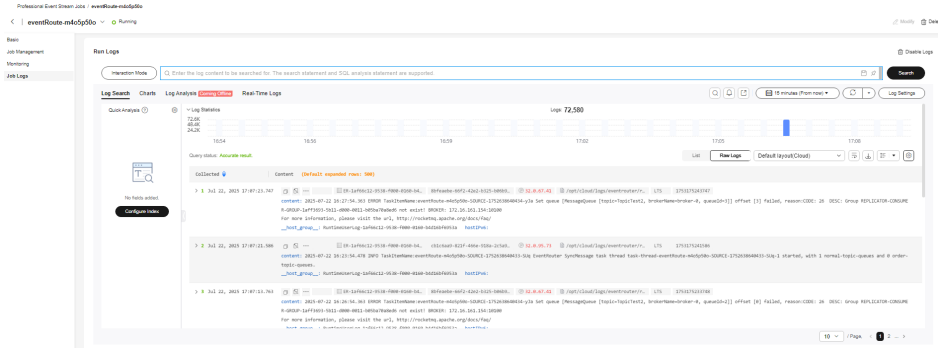
Step 5 If the log function is not enabled, click **Enable Logs** and toggle on **Manually Configure Files**.

Step 6 Select the existing log group and log stream, and click **OK**.

NOTE

Before enabling logs, you need to enable the LTS service and create a log group and log stream. For details, see **Log Management**.

Step 7 View job logs.



----End

6.2.6 Professional Event Stream Pre-check

Pre-check checks whether the entered configuration information meets the requirements. Pre-check contains multiple check items. For details, see **Table 6-27**. Each check item is executed independently. The check result can be successful, failed, or warning.

Table 6-27 Check item introduction

Item	Content
Check the source and target instance versions	Check whether the source and target instance versions match.
Check the source instance connectivity	Check whether the VM where the job is running can connect to the source instance.
Check the target instance connectivity	Check whether the VM where the job is running can connect to the target instance.
Check the source and target instance specifications	Check whether the source and target instance specifications match.

If the pre-check result is warning or failed for an event stream job, the creation of the event stream job is not blocked. The pre-check result is used only for reference.

6.2.6.1 Kafka Pre-check

Checking the Source and Target Instance Versions

- Check whether the source and target instance versions meet the requirements. Currently, the Kafka instance version must be 3.x or 2.7. If the version is not in the matching list, a warning is generated.
  - Cause 1: The source Kafka instance version is not supported. Select a supported instance.

    Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a source instance that meets the version requirements.
  - Cause 2: The target Kafka instance version is not supported. Select a supported instance.

    Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a target instance that meets the version requirements.
- Check whether the source and target Kafka instances are of the same version. If not, a warning is generated.

  Cause: The source and target instances are of different versions. Check their versions.

  Solution: Check whether the source and target Kafka instance versions are the same in the **Configure Source and Target** page. If they are different, modify the source or target instance to ensure that they are the same. Otherwise, there may be compatibility and performance risks.

Checking the Source Connectivity

Check whether the VM where the event stream job is located can connect to the source Kafka instance.

- Cause 1: The source Kafka instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the Kafka console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source Kafka instance.
- Cause 2: The source Kafka instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the Kafka console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source Kafka instance.

  c. Check whether the username and password of the source Kafka instance configured on the **Configure Source and Target** page are correct.

Checking the Target Connectivity

Check whether the VM where the event stream job is located can connect to the target Kafka instance.

- Cause 1: The target Kafka instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the Kafka console, check whether the target instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the target Kafka instance.

- Cause 2: The target Kafka instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the Kafka console, check whether the target instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the target Kafka instance.

  c. Check whether the username and password of the target Kafka instance configured on the **Configure Source and Target** page are correct.

Checking the Source and Target Instance Specifications

Check whether the specifications of the source and target Kafka instances match.

Cause: The specifications of the source and target Kafka instances are inconsistent. Check the specifications of the source and target instances.

Solution: Check whether the specifications of the source and target Kafka instances are the same on the **Configure Source and Target** page. If not, modify the source or target instance to ensure that the specifications are the same.

Checking the Addresses of the Source and Target Instances

Check whether the source and target instance have the same IP address.

Cause: The source and target Kafka instances have the same IP address. Check the node IP addresses of the source and target instances.

Solution: Check the node connection addresses and IP addresses of the source and target instances. If they use the same IP address, modify either the source or target instance to ensure the IP addresses are unique.

6.2.6.2 RocketMQ Pre-check

Checking the Source and Target Instance Versions

- Check whether the source and target instance versions meet the requirements. Currently, the RocketMQ instance version must be 5.x or 4.8.0. If the version is not in the matching list, a warning is generated.
  - Cause 1: The source RocketMQ instance version is not supported. Select a supported instance.

Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a source instance that meets the version requirements.

- Cause 2: The target RocketMQ instance version is not supported. Select a supported instance.

  Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a target instance that meets the version requirements.

- Check whether the source and target RocketMQ instances are of the same version. If not, a warning is generated.

  Cause: The source and target instances are of different versions. Check their versions.

  Solution: Check whether the source and target RocketMQ instance versions are the same in the **Configure Source and Target** page. If they are different, modify the source or target instance to ensure that they are the same.

Checking the Source Connectivity

Check whether the VM where the event stream job is located can connect to the source RocketMQ instance.

- Cause 1: The source RocketMQ instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the RocketMQ console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source RocketMQ instance.

- Cause 2: The source RocketMQ instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the RocketMQ console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source RocketMQ instance.

  c. Check whether the username and password of the source RocketMQ instance configured on the **Configure Source and Target** page are correct.

Checking the Target Connectivity

Check whether the VM where the event stream job is located can connect to the target RocketMQ instance.

- Cause 1: The target RocketMQ instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the RocketMQ console, check whether the target instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the target RocketMQ instance.

- Cause 2: The target RocketMQ instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the RocketMQ console, check whether the target instance is normal.
  b. Check whether the VPC and subnet of the event stream cluster can connect to the target RocketMQ instance.
  c. Check whether the username and password of the target RocketMQ instance configured on the **Configure Source and Target** page are correct.

Checking the Source and Target Instance Specifications

- Check whether the specifications of the source and target RocketMQ instances match.

  Cause: The brokers of the source and target RocketMQ instances are inconsistent.

  Solution: Check whether the source and target RocketMQ instance brokers are the same in the **Configure Source and Target** page. If not, modify the source or target instance to ensure that they are the same.

- Check whether the specifications of the source and target RocketMQ instances match.

  Cause: The specifications of the source and target RocketMQ instances are inconsistent.

  Solution: Check whether the source and target RocketMQ instance types are the same in the **Configure Source and Target** page. If not, modify the source or target instance to ensure that they are the same.

Checking the Addresses of the Source and Target Instances

Check whether the source and target instance have the same IP address.

Cause: The source and target RocketMQ instances have the same IP address. Check the node IP addresses of the source and target instances.

Solution: Check the node connection addresses and IP addresses of the source and target instances. If they use the same IP address, modify either the source or target instance to ensure the IP addresses are unique.

6.2.6.3 DCS Pre-check

Checking the Source and Target Instance Versions

- Check whether the source and target instance versions meet the requirements. The source DCS instance version must be 4.0, 5.0, or 6.0, and the target instance version must be 5.0. If the versions are not in the matching list, an alarm is generated.
  - Cause 1: The source DCS instance version is not supported.

    Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a source instance that meets the version requirements.

  - Cause 2: The target DCS instance version is not supported.

    Solution: Go to the **Configure Source and Target** page of the professional event stream job and select a target instance that meets the version requirements.

- Check whether the source and target DCS instances are of the same version. If not, a warning is generated.

  Cause: The source and target instances are of different versions.

  Solution: Check whether the source and target DCS instance versions are the same in the **Configure Source and Target** page. If not, modify the source or target instance to ensure that they are the same. Otherwise, there may be compatibility and performance risks.

Checking the Source Connectivity

Check whether the VM where the event stream job is located can connect to the source DCS instance.

- Cause 1: The source DCS instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the DCS console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source DCS instance.

- Cause 2: The source DCS instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the DCS console, check whether the source instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the source DCS instance.

  c. Check whether the username and password of the source DCS instance configured on the **Configure Source and Target** page are correct.

Checking the Target Connectivity

Check whether the VM where the event stream job is located can connect to the target DCS instance.

- Cause 1: The target DCS instance cannot be connected. Check whether the network configuration is correct.

  Solution:

  a. On the DCS console, check whether the target instance is normal.

  b. Check whether the VPC and subnet of the event stream cluster can connect to the target DCS instance.

- Cause 2: The target DCS instance cannot be connected. Check whether the entered instance information is correct.

  Solution:

  a. On the DCS console, check whether the target instance is normal.

b. Check whether the VPC and subnet of the event stream cluster can connect to the target DCS instance.
c. Check whether the username and password of the target DCS instance configured on the **Configure Source and Target** page are correct.

Checking the Source and Target Instance Specifications

Check whether the specifications of the source and target DCS instances match.

Cause: The specifications of the source and target DCS instances are inconsistent.

Solution: Check whether the specifications of the source and target DCS instances are the same on the **Configure Source and Target** page. If not, modify the source or target instance to ensure that the specifications are the same.

Checking the Source and Target Instance Configuration

Cause: In the sync scenario, change the value of **eventlog** to **true** and disable the client IP pass-through configuration of the source and target DCS instances.

Solution: Check whether the **eventlog** parameter has been set to **true** and whether the client IP pass-through configuration has been disabled for the source and target DCS instances. If not, modify the configuration.

Checking the Source and Target Instance IDs

Cause: The IDs of the source and target DCS instances are the same.

Solution: Check whether the source and target DCS instances are the same. If they are the same, change them to different instances.

Checking the Addresses of the Source and Target Instances

Check whether the source and target instance have the same IP address.

Cause: The source and target DCS instances have the same IP address. Check the node IP addresses of the source and target instances.

Solution: Check the node connection addresses and IP addresses of the source and target instances. If they use the same IP address, modify either the source or target instance to ensure the IP addresses are unique.

7 Events

Events are data that complies with specific specifications. Events that event sources publish to EG must comply with the CloudEvents specification.

EG uses KMS for static encryption. By default, all stored data and metadata are automatically encrypted, which meets data security and compliance requirements without extra configuration or fees.

EG supports the following events:

- Huawei Cloud service: events produced by Huawei Cloud service event sources
- Custom: events produced by custom event sources connected to EG with SDKs

Constraints

- Max. size per event: 64 KB
- Max. size of all events per request: 256 KB
- Max. events per request: 20

Example Event

The following is an example of an event published to EG:

```
{
   "events":[{
   "id": "4b26115b-778e-11ec-833e-cf74******",
      "specversion": "1.0",
   "source": "HC.OBS",
   "type": "object:put",
   "datacontenttype": "application/json",
   "subject": "xxx.jpg",
   "time": "2022-01-17T12:07:48.955Z",
   "data": {
      "name": "test01",
      "state": "enable"
   }
   }]
}
```

Table 7-1 describes the parameters in this example.

Table 7-1 Event parameters

Parameter	Type	Required	Example Value	Description
id	String	Yes	4b26115b-778e-*******-833e-cf74af	Event ID, which identifies an event
specversion	String	Yes	1.0	Version of the CloudEvents specification
source	String	Yes	HC.OBS	Event source that produces the event
type	String	Yes	object:put	Event type related to the event source
datacontenttype	String	No	application/json	Content format of the **data** parameter Only **application/json** is supported.
subject	String	No	xxx.jpg	Event subject
time	Timestamp	No	2022-01-17T12:07:48.955Z	Time when the event was produced

data	Struct	No	{
			"name": "test01",
			"state": "enable"
			}
				Content of the event in JSON format

Sending Events in Batches

The following is an example of the request body for sending events in batches:

```
{
   "events":[{
   "id": "eg-test-001",
   "specversion": "1.0",
   "source": "HC.OBS",
   "type": "object:put",
   "datacontenttype": "application/json",
   "subject": "xxx.jpg",
   "time": "2022-01-17T12:07:48.955Z",
   "data": {
      "name": "test01",
      "state": "enable"
   }
   },
   {
   "id": "eg-test-002",
   "specversion": "1.0",
   "source": "HC.OBS",
   "type": "object:put",
   "datacontenttype": "application/json",
   "subject": "xxx.jpg",
   "time": "2022-01-17T12:07:48.955Z",
   "data": {
      "name": "test01",
```



```
        "state": "enable"
    }
    },
    {
    "id": "eg-test-003",
    "specversion": "1.0",
    "source": "HC.OBS",
    "type": "object:put",
    "datacontenttype": "application/json",
    "subject": "xxx.jpg",
    "time": "2022-01-17T12:07:48.955Z",
    "data": {
        "name": "test01",
        "state": "enable"
    }
    },...]
}
```

Response body returned when all events are successfully sent:

```
{"failed_count":0,"events":[{"error_code":null,"error_msg":null,"event_id":"eg-test-003"},
{"error_code":null,"error_msg":null,"event_id":"eg-test-003"},
{"error_code":null,"error_msg":null,"event_id":"eg-test-002"}]}
```

Status code: 200

Response body returned when the number of events per request exceeds the upper limit:

```
{"failed_count":1,"events":[{"error_code":"00533013","error_msg":"Too many events for a
request.","event_id":"eg-test-003"},{"error_code":null,"error_msg":null,"event_id":"eg-test-003"},
{"error_code":null,"error_msg":null,"event_id":"eg-test-002"}]}
```

Status code: 400

Response body returned when the size of an event exceeds the upper limit:

```
{"failed_count":3,"events":[{"error_code":00533012,"error_msg":An event is too large."event_id":"eg-
test-003"},{"error_code":00533012,"error_msg":the number of events exceeds the limit,"event_id":"eg-
test-003"},{"error_code":00533012,"error_msg":the number of events exceeds the limit,"event_id":"eg-
test-002"}]}
```

Status code: 400

Response body returned when the total size of all events per request exceeds the upper limit:

```
{"error_code":"00533007","error_msg":"The total size of a request's all events is too
large.","error_detail":"The total size of a request's all events is too large."}
{"error_code":"00533012","error_msg":"An event is too large.","error_detail":"An event is too large."}
{"error_code":"00533013","error_msg":"Too many events for a request.","error_detail":"Too many events for
a request."}
```

Status code: 400

NOTE

If the status code is **400**:

- The total size of all events per request exceeds the upper limit. (Error code: **EG.00533007**; error message: **The total size of a request's all events is too large**)
- The number of events per request exceeds the upper limit. (Error code: **EG.00533013**; error message: **Too many events for a request**)

8 Event Rules

8.1 Introduction

Event rules define how to filter and transform events.

- Filter: By configuring **filter rules** in a subscription, specify what events will be routed to the relevant target. For more information about **filter rules**, see **Filter Rule Parameters** and **Example Filter Rules**.
- Transform: By configuring the **transform type** in a subscription, determine how to transform events for the relevant target. For more information about event content transformation, see **Event Content Transformation**.

8.2 Filter Rule Parameters

Only events that match your filter rules will be routed to the associated targets. These filter rules must have the same structure as the events.

This section describes the restrictions of filter rules as well as the operators, condition expressions, and matching fields.

Restrictions

Event filter rules must meet the following requirements:

- Top-level fields can only be **source**, **type**, **subject**, or **data**.
- Top-level fields must include **source**, and **source** only supports the **StringIn** operator.
- The **data** field allows max. 5 subfields, and each can have max. 5 levels.
- Each field can have max. 5 conditions in an OR relationship.
- Multiple fields are ANDed with each other.
- A field that appears more than once at the same level will be used where it appears the last time.

Operators

Table 8-1 lists the operators that can be used in event filter rules.

Table 8-1 Operators

Operator	Input Value	Condition Value	Description
StringIn	String/ String[]	String[] values	Check if the input value matches any condition value.
StringNotIn	String/ String[]	String[] values	Check if the input value does not match any condition value.
StringStartsWith	String/ String[]	String[] values	Check if the input value prefix matches any condition value.
StringNotStartsWith	String/ String[]	String[] values	Check if the input value prefix does not match any condition value.
StringEndsWith	String/ String[]	String[] values	Check if the input value suffix matches any condition value.
StringNotEndsWith	String/ String[]	String[] values	Check if the input value suffix does not match any condition value.
NumberIn	Number/ Number[]	Number[] values	Check if the input value matches any condition value.
NumberNotIn	Number/ Number[]	Number[] values	Check if the input value does not match any condition value.
NumberLessThan	Number/ Number[]	Number value	Check if the input value is less than the condition value.
NumberNotLessThan	Number/ Number[]	Number value	Check if the input value is greater than or equal to the condition value.
NumberGreaterThan	Number/ Number[]	Number value	Check if the input value is greater than the condition value.
NumberNotGreaterThan	Number/ Number[]	Number value	Check if the input value is less than or equal to the condition value.
NumberInRange	Number/ Number[]	Number[][] values	Check if the input value is within any condition value range.
NumberNotInRange	Number/ Number[]	Number[][] values	Check if the input value is not within any condition value range.
IsNull	-	None	Check if the input value is null or undefined.
IsNotNull	-	None	Check if the input value is neither null nor undefined.

Operator	Input Value	Condition Value	Description
IsTrue	Boolean	None	Check if the input value is true.
IsNotTrue	Boolean	None	Check if the input value is false.

Condition Expressions

Table 8-2 lists the condition expressions that can be used in event filter rules.

Table 8-2 Condition expressions

Field Name	Type	Required	Description
op	String	Yes	Operator
value	JSON Type	No	Condition value
values	JSON Array	No	Condition value range

Matching Fields

Table 8-3 lists the matching fields that can be used in event filter rules.

Table 8-3 Matching fields

Field Name	Condition Value Type	Example
source	JSON array	Event source. The condition value is in the JSON array. This field can only be used with the StringIn operator. Example: **[{"op": "StringIn", "values": ["HC.OBS"]]**
type	JSON array	Event type. The condition value is in the JSON array. Example: **[{"op": "StringIn", "values": ["object:put"]]**
subject	JSON array	Event body. The condition value is in the JSON array. Example: **[{"op": "StringEndsWith", "values": [".jpg"]]**
data	JSON object	Event data. The condition value is in the JSON object, and can be nested in max. 5 layers. Example: **{"state": [{"op": "StringIn", "values": ["running"]]}]**

8.3 Example Filter Rules

This section provides examples of filter rules of all matching types.

These matching types are available:

- **Exact Match**
- **Exclusion Match**
- **Prefix Match**
- **Prefix Not Matching**
- **Suffix Match**
- **Suffix Not Matching**
- **Value Range Match**
- **Null Match**
- **Non-null Match**
- **True Match**
- **Non-true Match**

Exact Match

Filter events that exactly match a specified string. As shown in the following table, events whose **source** is **HC.OBS** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}]
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Filter events that exactly match a specified number. As shown in the following table, events whose **age** in **data** is **10** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"age":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"age":[{
	"op": "NumberIn",
	"values":[10]
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"age":10
		}
		}]
		}

Exclusion Match

Filter events that do not match a specified string. As shown in the following table, events whose **type** is not **object:get** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"type": [{
	"op": "StringNotIn",
	"values": ["object:get"]
	}]
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Filter events that do not match a specified number. As shown in the following table, events whose **age** in **data** is not **11** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"age":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"age":[{
	"op": "NumberNotIn",
	"values":[11]
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"age":10
		}
		}]
		}

Prefix Match

Filter events whose prefix matches a specified value. As shown in the following table, events whose **type** starts with **object:** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"type": [{
	"op": "StringStartsWith",
	"values": ["object:"]
	}]
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Prefix Not Matching

Filter events whose prefix does not match a specified value. As shown in the following table, events whose **source** does not start with **HC** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"source": [{
	"op": "StringNotStartsWith",
	"values": ["HC"]
	}]
	}
		None

Suffix Match

Filter events whose suffix matches a specified value. As shown in the following table, events whose **subject** ends with **jpg** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"subject": [{
	"op": "StringEndsWith",
	"values": ["jpg"]
	}]
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Suffix Not Matching

Filter events whose suffix does not match a specified value. As shown in the following table, events whose **subject** does not end with **txt** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"subject":[{
	"op": "StringNotEndsWith",
	"values": ["txt"]
	}]
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Value Range Match

Filter events that match a specified value range. As shown in the following table, events whose **size** in **data** is less than **20** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberLessThan",
	"value":20
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":10
		}
		}]
		}

As shown in the following table, events whose **size** in **data** is greater than or equal to **2** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberNotLessThan",
	"value":2
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":10
		}
		}]
		}

As shown in the following table, events whose **size** in **data** is greater than **9** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberGreaterThan",
	"value":9
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":10
		}
		}]
		}

As shown in the following table, events whose **size** in **data** is less than or equal to **9** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberNotGreaterThan",
	"value":9
	}]
	}
	}
		None

As shown in the following table, events whose **size** in **data** is from 1 to 20 are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberInRange",
	"values":[
	[
	1,
	20
	]
	]
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":10
		}
		}]
		}

As shown in the following table, events whose **size** in **data** is less than **1** or greater than **20** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size":10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "NumberNotInRange",
	"values":[
	[
	1,
	20
	]
	]
	}]
	}
	}
		None

Null Match

Filter events with a null value or undefined field. As shown in the following table, events whose **size** and **age** in **data** are **null** or undefined are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size": null
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "IsNull"
	}],
	"age":[{
	"op": "IsNull"
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":null
		}
		}]
		}

Non-null Match

Filter events whose certain field is not **null**. As shown in the following table, events whose **size** and **name** in **data** are not **null** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size": 10
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "IsNotNull"
	}],
	"name":[{
	"op": "IsNotNull"
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size":10
		}
		}]
		}

True Match

Filter events whose certain field is **true**. As shown in the following table, events whose **size** and **name** in **data** are **true** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": true,
"state": "enable",
"size": true
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"size":[{
	"op": "IsTrue"
	}],
	"name":[{
	"op": "IsTrue"
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": true,
		"state": "enable",
		"size":true
		}
		}]
		}

Non-true Match

Filter events whose certain field is not **true**. As shown in the following table, events whose **name** in **data** is not **true** are matched.

Event from Source	Filter Rule	Matched Event

{
"events":[{
"id": "4b26115b-778e-11ec-******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:00.955Z",
"data": {
"name": "test01",
"state": "enable",
"size": null
}
}]
}
	{
	"source": [{
	"op": "StringIn",
	"values": ["HC.OBS"]
	}],
	"data":{
	"name":[{
	"op": "IsNotTrue"
	}]
	}
	}
		{
		"events":[{
		"id": "4b26115b-778e-11ec-******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:00.955Z",
		"data": {
		"name": "test01",
		"state": "enable",
		"size": null
		}
		}]
		}

8.4 Event Content Transformation

EG transforms CloudEvents-compliant events so that they can be processed by specified targets.

Supported transform types: pass-through, variables, constants.

Constraints

When using the variable transformation type, the event stream cannot transform fields in the source message content.

Pass-through

Directly route CloudEvents-compliant events to the target. Example:

Before Transformation	Transform Type	After Transformation

{
"events":[{
"id": "4b26115b-73e-cf74a******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:48.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	Pass-through	{
		"events":[{
		"id": "4b26115b-73e-cf74******",
		"specversion": "1.0",
		"source": "HC.OBS",
		"type": "object:put",
		"datacontenttype": "application/json",
		"subject": "xxx.jpg",
		"time": "2022-01-17T12:07:48.955Z",
		"data": {
		"name": "test01",
		"state": "enable"
		}
		}]
		}

Variables

Route variables in CloudEvents-compliant events to the target by using a template. Example:

Before Transformation	Transform Type	After Transformation

{
"events":[{
"id": "4b26115b-73e-cf74a******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:48.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	Parameter
	{"name":"$.data.name"}
	Template
	My name is ${name}
	If the event target is **FunctionGraph (function computing)**, the template must be in JSON format. Example:
	{"name":"${name}"}
		My name is test01
		NOTE
		If the event target is **FunctionGraph (function computing)**, the transformation result is as follows:
		{"name": "test01"}

Example of complex transformation from OBS to EG and then to FunctionGraph:

Before Transformation	Transform Type	After Transformation

<pre>{
"specversion": "1.0",
"id": "******9db447aa3******",
"source": "HC.OBS.DWR",
"type": "OBS:DWR:ObjectCreated:PUT",
"datacontenttype": "application/json",
"dataschema": "",
"subject": "test.txt",
"time": "2023-08-01T11:41:51.712759419Z",
"ttl": "4000",
"data": {
"eventVersion": "3.0",
"eventSource": "OBS",
"eventRegion": "cn-north-4",
"eventTime": "2023-08-01T19:41:47.879Z",
"eventName": "ObjectCreated:Put",
"userIdentity": {
"ID": "******fef0f08c******"
},
"requestParameters": {
"sourceIPAddress": "1**.1**.1**.1**"
},
"responseElements": {
"x-obs-request-id": "******47aa3cdafb******",
"x-obs-id-2": "",
"x-amz-request-id": "",
"x-amz-id-2": ""
},
"obs": {
"Version": "1.0",
"configurationId": "******4aaac1******",
"bucket": {
"name": "test",
"ownerIdentity": {
"ID": "******f1234567******"
},
"bucket": "test",
"arn": ""
},
"object": {
"key": "test.txt",
"eTag": "******48ce552c3******",
"size": 13,
"versionId": "******BE7FFFF******",
"sequencer": "1",
"oldpsxpth": ""
}
}
}
}</pre>
	Parameters
	<pre>{
	"eventVersion": "$.data.eventVersion",
	"eventTime": "$.data.eventTime",
	"requestParameters": "$.data.requestParameters.sourceIPAddress",
	"configurationId": "$.data.obs.configurationId",
	"eTag": "$.data.obs.object.eTag",
	"sequencer": "$.data.obs.object.sequencer",
	"key": "$.data.obs.object.key",
	"size": "$.data.obs.object.size",
	"arn": "$.data.obs.bucket.arn",
	"name": "$.data.obs.bucket.name",
	"ownerIdentity": "$.data.obs.bucket.ownerIdentity.ID",
	"Region": "$.data.eventRegion",
	"eventName": "$.type",
	"userIdentity": "$.data.userIdentity.ID"
	}</pre> Template
	<pre>{
	"Records": [
	{
	"eventVersion": "${eventVersion}",
	"eventTime": "${eventTime}",
	"requestParameters": {
	"sourceIPAddress": "${requestParameters}"
	},
	"obs": {
	"configurationId": "${configurationId}",
	"object": {
	"eTag": "${eTag}",
	"sequencer": "${sequencer}",
	"key": "${key}",
	"size": "${size}"
	},
	"bucket": {
	"arn": "${arn}",
	"name": "${name}",
	"ownerIdentity": {
	"PrincipalId": "${ownerIdentity}"
	}
	}
	},
	"Region": "${Region}",
	"eventName": "${eventName}",</pre>
		<pre>{
		"Records": [
		{
		"eventVersion": "3.0",
		"eventTime": "2023-08-01T19:41:47.879Z",
		"requestParameters": {
		"sourceIPAddress": "1**.1**.1**.1**"
		},
		"obs": {
		"configurationId": "******4aaac1******",
		"object": {
		"eTag": "******48ce552c3******",
		"sequencer": "1",
		"key": "test.txt",
		"size": "13"
		},
		"bucket": {
		"arn": "",
		"name": "test",
		"ownerIdentity": {
		"PrincipalId": "******f1234567******"
		}
		}
		},
		"Region": "cn-north-4",
		"eventName": "OBS:DWR:ObjectCreated:PUT",
		"userIdentity": {
		"principalId": "******fef0f08c******"
		}
		}
		]
		}</pre>

Before Transformation	Transform Type	After Transformation

	"userIdentity": {
	"principalId": "$
	{userIdentity}"
	}
	}
	]
	}
	The value in the template is the key of the corresponding parameter.	

Constants

Route constants in events to the target. Example:

Before Transformation	Rule	After Transformation

{
"events":[{
"id": "4b26115b-73cf74a******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/
json",
"subject": "xxx.jpg",
"time":
"2022-01-17T12:07:48.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	Parameter
	test01
	If the event target is **FunctionGraph (function computing)**, the rule must be in JSON format. Example:
	{"name": "test01"}
		test01
		NOTE
		If the event target is **FunctionGraph (function computing)**, the transformation result is as follows:
		{"name": "test01"}

More Examples

1. After you set a DMS for RabbitMQ or DMS for RocketMQ **event source** for a subscription, messages will contain the **context** field in **data** after being transformed to CloudEvents-compliant events. If you set the transform type to **Variables** for the **event target**, the rule must also contain the **context** field. Example:

Before Transformation	Transform Type	After Transformation

{
"type": "ROCKETMQ:CloudTrace:RocketmqCall",
"data": {
"context": {
"name": "test01",
"state": "enable"
}
},
"source": "zhang_roc",
"time": "2023-02-01T10:47:07Z",
"datacontenttype": "application/json",
"specversion": "1.0",
"id": "2f885496-570c-4925-82fd-d1ad09******",
"subject": "ROCKETMQ:cn-north-7:eec88b34-9470-483e-8961-edb168******/0de095e33e00d36e2fd2c0019a******:ROCKETMQ:zhang_roc"
}
	Parameter
	{"name":"$.data.context.name"}
	Template
	My name is ${name}
		My name is test01

2. If you set **Event Target** to **FunctionGraph (function computing)** and **Transform Type** to **Pass-through** in **Step 8**, the event content cannot be transferred to the event target as the input value. To do it, you can use **Variables** or **Constants** transform type and configure **input** field in the rule. The following is an example:

Table 8-4 Variables type example

Before Transformation	Rule	After Transformation

{
"events":[{
"id": "4b26115b-73cf74a******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype": "application/json",
"subject": "xxx.jpg",
"time": "2022-01-17T12:07:48.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	Variable
	{"data": "$.data"}
	Template
	{"input": ${data}}
		{
		"input": {
		"name": "test01",
		"state": "enable"
		}
		}

Table 8-5 Constants type example

Before Transformation	Rule	After Transformation

{
"events":[{
"id":
"4b26115b-73cf74a******",
"specversion": "1.0",
"source": "HC.OBS",
"type": "object:put",
"datacontenttype":
"application/json",
"subject": "xxx.jpg",
"time":
"2022-01-17T12:07:48.955Z",
"data": {
"name": "test01",
"state": "enable"
}
}]
}
	Constant
	{
	"input": {
	"name": "test01"
	}
	}
		{
		"input": {
		"name": "test01"
		}
		}

9 Event Targets

Event targets are destinations that receive and process events.

EG supports the following event targets:

- Huawei Cloud services connected to EG.
- Custom event processing services

10 Network Management

10.1 Connections

You can connect to the private webhook through the VPC and subnet connection.

If you use the default connection, ensure that your custom event target supports public access.

Custom connections can also be based on DMS for Kafka.

NOTE

A client or proxy client provides a webhook URL to receive data from a specified server. The client updates accordingly once the server pushes data to the URL.

Constraints

- Webhook URLs must support TLS 1.2 and secure encryption algorithms.
- Kafka instance parameters cannot be modified once the connection is created.
- If the connection to delete is associated with subscriptions, disassociate it first.

Creating a Webhook Connection

Before creating a connection, ensure that you have VPC permissions.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Connections**.

Step 3 Click **Create Connection**.

NOTE

When you create your first connection, your authorization will be required and an agency will be automatically created. For details, see **Authorization**.

Step 4 Configure the connection by referring to **Table 10-1**.

Table 10-1 Connection parameters

Parameter	Description
Type	Select **WEBHOOK**.
Name	Connection name. The name cannot be modified once the connection is created.
Description	Describe the connection.
VPC	Select a VPC. The VPC cannot be changed once the connection is created.
Subnet	Select a subnet. The subnet cannot be changed once the connection is created.

Step 5 Click **OK**.

----End

Creating a DMS for Kafka Connection

Before creating such a connection, ensure that you already have a DMS for Kafka instance.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Connections**.

Step 3 Click **Create Connection**.

NOTE

When you create your first connection, your authorization will be required and an agency will be automatically created. For details, see **Authorization**.

Step 4 Configure the connection by referring to **Table 10-2**.

Table 10-2 Kafka connection parameters

Parameter	Description
Type	Select **DMS for Kafka**.
Name	Connection name. The name cannot be modified once the connection is created.
Description	Describe the connection.
Instance	Select a Kafka instance.
Access Mode	Select **Ciphertext Access** or **Plaintext Access**.
Security Protocol	If you select **Ciphertext Access** for **Access Mode**, the corresponding security protocol will be displayed.

Parameter	Description
SASL_SSL Authentication	Available when SASL_SSL authentication is enabled for the Kafka instance. Select an authentication mechanism. • **PLAIN**: a simple username and password verification mechanism. • **SCRAM-SHA-512**: uses the hash algorithm to generate credentials for usernames and passwords to verify identities. SCRAM-SHA-512 is more secure than PLAIN.
Username	Available when SASL_SSL authentication is enabled for the Kafka instance. Enter a username.
Password	Available when SASL_SSL authentication is enabled for the Kafka instance. Enter a password.
Acknowledgments	Number of acknowledgments the producer requires the server to return before considering a request complete. • **None**: The producer will not wait for any acknowledgment from the server at all. The record will be immediately added to the socket buffer and considered sent. No guarantee can be made that the server has received the record. • **Leader only**: The leader will write the record to its local log but will respond without waiting until receiving full acknowledgement from all followers. If the leader fails immediately after acknowledging the record but before the followers have replicated it, the record will be lost. • **All**: The leader will wait for the full set of in-sync replicas to acknowledge the record. This is the strongest available guarantee because the record will not be lost as long as there is just one working replica.

Step 5 Click **OK**.

----End

Editing a Connection

Only the description of a connection can be modified.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Connections**.

Step 3 Click **Edit** in the row that contains the desired connection.

Step 4 Modify the description and click **OK**.

----End

Deleting a Connection

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Connections**.

Step 3 Click **Delete** in the row that contains the desired connection.

Step 4 Click **OK**.

----End

10.2 Endpoints

An endpoint is an EG access address for you to push events from a custom source.

EG supports the following endpoints:

- Public endpoints: fixed public domain names for specific regions
- Private endpoints: EG private domain names you create for pushing custom events

Constraints

- Creating an endpoint will generate a VPC endpoint with fees. Delete the created endpoint when you no longer need it.
- The VPC and subnet cannot be changed once the endpoint is created.
- If the related DNS and VPCEP resources have been deleted, the private endpoint may fail to be deleted. In this case, contact EG O&M personnel.

Creating a Private Endpoint

Before creating a private endpoint, ensure that you have DNS and VPCEP permissions.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Endpoints**.

Step 3 Click **Create Endpoint**.

Step 4 Configure the endpoint by referring to **Table 10-3**.

Table 10-3 Endpoint parameters

Parameter	Description
Name	Endpoint name. The name cannot be modified once the endpoint is created.
VPC	Select a VPC. The VPC cannot be changed once the endpoint is created.
Subnet	Select a subnet. The subnet cannot be changed once the endpoint is created.
Description	Describe the endpoint.

Step 5 Click **OK**.

----End

Editing a Private Endpoint

Modify the description of an endpoint.

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Endpoints**.

Step 3 Click **Edit** in the row that contains the desired endpoint.

Step 4 Modify the description and click **OK**.

----End

Deleting a Private Endpoint

Step 1 Log in to the **EG console**.

Step 2 In the navigation pane, choose **Network Management** > **Endpoints**.

Step 3 Click **Delete** in the row that contains the desired endpoint.

Step 4 Click **OK**.

----End

11 IAM Projects and Enterprise Projects

Constraints

- **Enterprise** is available on the management console only if you have enabled the enterprise project, or your account is the primary account. To enable this function, contact customer service.
- Currently, only subscriptions and channels can be managed using enterprise projects.

Creating an IAM Project and Assigning Permissions

- Creating an IAM Project

  Go to the management console, hover over the username in the upper right corner, and choose **Identity and Access Management** from the drop-down list. Choose **Projects** in the navigation pane, and click **Create Project**. On the displayed **Create Project** page, select a region and enter a project name.

- Authorization

  You can assign permissions (for resources and operations) to user groups to associate projects with the user groups. To do so, add users to a user group to control projects that the users can access and the resources on which the users can perform operations. For details, see **Creating a User Group and Assigning Permissions**.

Creating an Enterprise Project and Assigning Permissions

- Creating an Enterprise Project

  Go to the management console, and choose **Enterprise** > **Project Management** in the upper right corner. On the displayed page, click **Create Enterprise Project** to create a project.

- Authorization

  You can add a user group to an enterprise project and configure a policy to associate the enterprise project with the user group. To do so, add users to a user group to control projects that the users can access and the resources on which the users can perform operations. Perform the following procedure:

  a. On the **Enterprise Management** console, click the name of an enterprise project to go to the details page.

b. On the **Permissions** tab, click **Authorize User Group** to go to the **User Groups** page on the IAM console. Associate the enterprise project with a user group and assign permissions to the group.

   For details, see **Creating a User Group and Assigning Permissions**.

- Associating Resources with Enterprise Qualifications

  Enterprise projects help you centrally manage cloud resources.

  – Selecting an enterprise project when buying EG

    On the page for buying EG, select an enterprise project to associate your resources with it.

  – Adding resources

    On the **Enterprise Project Management** page, you can add existing EG resources to a target enterprise project.

    default is the default enterprise project. Resources that are not allocated to any enterprise project under your account are listed in this project.

  For more information, see **Enterprise Management User Guide**.

12 Authorization

Some functions of EG require your authorization and an agency will be automatically created. For details, see **Table 12-1**.

Table 12-1 Agency information

Agency Name	Authorizer	Authorized	Assigning Permissions	Description
EG_DELEGATE_FG_AGENCY	User	FunctionGraph	vpc:ports:delete vpc:ports:get vpc:ports:create vpc:vpcs:get vpc:subnets:get	When creating a function in an event stream, you need to authorize FunctionGraph to query the VPC, subnet, and port to connect to the network.
EG_AGENCY	User	EventGrid	eg:channels:get eg:channels:list eg:channels:putEvents	If the event source is DMS, you need to send events to the EG channel and authorize the channel permissions.

Agency Name	Authorizer	Authorized	Assigning Permissions	Description
EG_TARGET_AGENCY	User	EventGrid	functiongraph:function:invoke functiongraph:function:invokeAsync eg:channels:get eg:channels:list eg:channels:putEvents smn:topic:publish	Used for event sending to the subscription target (including FunctionGraph, EG, and SMN).
EG_DEDICATED_EVENT_STREAM_AGENCY	User	EventGrid	dcs:instance:list dcs:instance:get dms:instance:get dms:instance:list vpc:vpcs:get vpc:ports:create vpc:ports:delete vpc:ports:update vpc:ports:get vpc:subnets:get	Used by the professional event stream to synchronize DMS and DCS data, check the status of scheduled tasks, and clear residual data. During creation, you need to attach NICs and enable the network.

Authorization Scenarios

1. When **you create your first connection**, your authorization will be required. If you agree to authorize, an agency named **EG_DELEGATE_FG_AGENCY** will be automatically created in IAM. View this agency on the IAM console.
2. When **you create your first DMS for RabbitMQ or DMS for RocketMQ event source**, your authorization will be required. If you agree to authorize, agencies named **EG_DELEGATE_FG_AGENCY** and **EG_AGENCY** will be automatically created in IAM. View this agency on the IAM console.

3. When you create an event subscription for the first time and set event target to EG, SMN, or FunctionGraph (for details, see **Step 8**), you need to create an agency and agree to the authorization. After the authorization is successful, EG creates an agency named **EG_TARGET_AGENCY** on the IAM console. View this agency on the IAM console.
4. When you create your first **professional event stream cluster**, your authorization will be required. If you agree to authorize, an agency named **EG_DEDICATED_EVENT_STREAM_AGENCY** will be automatically created in IAM. View this agency on the IAM console.

13 Event Monitoring

13.1 Supported Metrics

Introduction

This section describes the monitoring metrics and dimensions of EG. View these metrics on the EG console.

Namespace

SYS.EG

Metrics

Table 13-1 Event delivery metrics

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
sub_num	Total Deliveries	Number of times event delivery attempts are made. Unit: count	≥ 0	Event subscription	1
sub_success_num	Successful Deliveries	Number of times events are finally delivered. Unit: count	≥ 0	Event subscription	1

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
process_time	Processing Time	Average time spent processing all event deliveries in a period. Unit: ms	≥ 0 ms	Event subscription	1
sub_failed_num	Failed Events	Number of events that fail to be delivered. Unit: count	≥ 0	Event subscription	1
sub_success_rate	Success Rate	Percentage of total deliveries that are successful. Unit: %	0%≤x≤100%	Event subscription	1
sub_failed_rate	Failure Rate	Percentage of total deliveries that failed. Unit: %	0%≤x≤100%	Event subscription	1
sub_retry_num	Delivery Retries	Number of times delivery retry is attempted. Unit: count	≥ 0	Event subscription	1
sub_retry_rate	Retry Rate	Percentage of total deliveries that are retried. Unit: %	0%≤x≤100%	Event subscription	1
sub_process_time	Processing Time	Average time spent processing an event delivery. Unit: ms	≥ 0 ms	Event subscription	1

Table 13-2 Event access metrics

ID	Name	Description	Value Range	Monitored Object	Raw Data Monitoring Period (Minute)
sub_num	Total Accesses	Number of times event access attempts are made. Unit: count	≥ 0	Event channel	1
sub_success_num	Successful Accesses	Number of times events are finally accessed. Unit: count	≥ 0	Event channel	1
sub_fail_num	Failed Accesses	Number of times events could not be accessed. Unit: count	≥ 0	Event channel	1
process_time	Processing Time	Average time spent processing all event accesses in a period. Unit: ms	≥ 0 ms	Event channel	1
pub_success_rate	Success Rate	Percentage of total accesses that are successful. Unit: %	0%≤x≤100%	Event channel	1
pub_failed_rate	Failure Rate	Percentage of total accesses that failed. Unit: %	0%≤x≤100%	Event channel	1

Dimensions

Key	Value
subscription_id	Event subscription ID
channel_id	Event channel ID

13.2 Viewing Monitoring Data

EG monitors event subscriptions and channels, and allows you to query event access and delivery information without any configuration.

Procedure

Step 1 Log in to the **EG console**.

Step 2 On the **Event Subscriptions** page, click **Monitoring** in the row that contains the desired subscription, and view the event delivery monitoring data.

- View the monitoring data of a single event target.
- View the monitoring data of the last hour, last 4 hours, last 24 hours, last 7 days, or a custom time range.
- Specify a period (1 minute, 5 minutes, or 20 minutes) and method (average, maximum, or minimum).

Step 3 On the **Event Channels** page, click **Monitoring** in the row that contains the desired channel, and view the event access monitoring data.

- View the monitoring data of a single event source.
- View the monitoring data of the last hour, last 4 hours, last 24 hours, last 7 days, or a custom time range.
- Specify a period (1 minute, 5 minutes, or 20 minutes) and method (average, maximum, or minimum).

----End

14 Auditing

14.1 EG Operations Recorded by CTS

Operations related to EG can be recorded with Cloud Trace Service (CTS) for query, audit, and backtracking.

Table 14-1 EG operations that can be recorded by CTS

Operation	Resource Type	Trace
Create event subscription	subscription	CreateSubscription
Query event subscription list	subscription	ListSubscriptions
Update event subscription	subscription	UpdateSubscription
Query event subscription details	subscription	ShowDetailOfSubscription
Delete event subscription	subscription	DeleteSubscription
Update event subscription source	subscription	UpdateSubscriptionSource
Create event subscription target	subscription	CreateSubscriptionTarget
Update event subscription target	subscription	UpdateSubscriptionTarget
Query details about event subscription target	subscription	ShowDetailOfSubscriptionTarget
Delete event subscription target	subscription	DeleteSubscriptionTarget

Enable or disable event subscription	subscription	ExecuteSubscriptionOperation
Query EG triggers of a function	subscription	ListTriggers
Query EG triggers of a function flow	subscription	ListWorkflowTriggers
Obtain OBS buckets	subscription	ListObsBuckets
Refurbish OBS	subscription	Refurbishobs
Create custom event channel	channel	CreateChannel
Query event channel	channel	ListChannels
Update event channel	channel	UpdateChannel
Query event channel details	channel	ShowDetailOfChannel
Delete custom event channel	channel	DeleteChannel
Publish event to event channel	channel	PutEvents
Precheck event publishing status	channel	CheckPutEvents
View event trace details	event	ShowDetailOfEventTrace
Query sent message content	event	ShowDetailOfEvent
Query event trace list	event	ListTracedEvents
Create custom event source	source	CreateEventSource
Query event source list	source	ListEventSources
Update custom event source	source	UpdateEventSource
Query event source details	source	ShowDetailOfEventSource
Delete custom event source	source	DeleteEventSource
Query event target catalogs	targetCatalogs	ListEventTarget
Start automatic event schema discovery	schemas	DiscoverEventSchemaFromData

Create custom event schema	schemas	CreateEventSchema
Query event schema list	schemas	ListEventSchema
Update custom event schema	schemas	UpdateEventSchema
Query event schema details	schemas	ShowDetailOfEventSchema
Delete schema	schemas	DeleteEventSchema
Create custom event schema version	schemas	CreateEventSchemaVersion
Query schema version list	schemas	ListEventSchemaVersions
Query event schema version details	schemas	ShowDetailOfEventSchemaVersion
Delete schema version	schemas	DeleteEventSchemaVersion
Create connection	connections	CreateConnection
Query connection list	connections	ListConnections
Query connection details	connections	ShowDetailOfConnection
Delete connection	connections	DeleteConnection
Query service agency	agency	ListAgencies
Create service agency	agency	CreateAgencies
Query quotas	quota	ListQuotas
Obtain API version list	apiVersion	ListApiVersions
Update endpoint	endpoints	UpdateEndpoint
Delete endpoint	endpoints	DeleteEndpoint
Create endpoint	endpoints	CreateEndpoint
Query endpoint	endpoints	ListEndpoints
Query event channel metrics	pubMetrics	ListPubMetrics
Query event subscription metrics	subMetrics	ListSubMetrics
Create event stream	eventStreaming	CreateEventStreaming
Query event stream list	eventStreaming	ListEventStreaming
Update event stream	eventStreaming	UpdateEventStreaming

Query event stream details	eventStreaming	ShowEventStreaming
Delete event stream	eventStreaming	DeleteEventStreaming
Start/Stop event stream	eventStreaming	ResumeEventStreaming
Query event example list	samples	ShowListOfEventSample
Query supported features	feature	QuerySupportFeature
Create professional event stream job	erjobs	CreateEventRouterJob
Query professional event stream job list	erjobs	ListEventRouterJobs
Delete professional event stream job	erjobs	DeleteEventRouterJob
Query professional event stream job details	erjobs	ShowEventRouterJob
Update professional event stream job	erjobs	UpdateEventRouterJob
Operate professional event stream job	erjobs	ExecuteEventRouterJobOperation
Verify professional event stream job	erjobs	ValidateEventRouterJob
Query sync status of a professional event stream job	erjobs	ShowEventRouterTaskSyncDetail
Enable/Disable event stream job logs	erjobs	UpdateEventRouterTaskLogFeature
Query unified service catalog	service	ShowEgServiceOnlineStatus
Query AZs of a professional event stream cluster	erclusters	ListAvailabilityZones
Create professional event stream cluster	erclusters	CreateEventRouterCluster
Query professional event stream clusters	erclusters	ListEventRouterClusters
Delete professional event stream cluster	erclusters	DeleteEventRouterCluster
Query professional event stream cluster details	erclusters	ShowEventRouterCluster

Update professional event stream cluster	erclusters	UpdateEventRouterCluster

14.2 Viewing CTS Traces in the Trace List

Scenarios

Cloud Trace Service (CTS) records operations performed on cloud service resources. A record contains information such as the user who performed the operation, IP address, operation content, and returned response message. These records facilitate security auditing, issue tracking, and resource locating. They also help you plan and use resources, and identify high-risk or non-compliant operations.

What Is a Trace?

A trace is an operation log for a cloud service resource, tracked and stored by CTS. Traces record operations such as adding, modifying, or deleting cloud service resources. You can view them to identify who performed operations and when for detailed tracking.

What Is a Management Tracker and Data Tracker?

A management tracker identifies and associates with all your cloud services, recording all user operations. It records management traces, which are operations performed by users on cloud service resources, such as their creation, modification, and deletion.

A data tracker records details of user operations on data in OBS buckets. It records data traces reported by OBS, detailing user operations on data in OBS buckets, including uploads and downloads.

Constraints

- Before the organization function is enabled, you can query the traces of a single account on the CTS console. After the organization function is enabled, you can only view multi-account traces on the **Trace List** page of each account, or in the OBS bucket or the **CTS/system** log stream configured for the management tracker with the organization function enabled. For details about organization trackers, see **Organization Trackers**.
- You can only query operation records of the last seven days on the CTS console. They are automatically deleted upon expiration and cannot be manually deleted. To store them for longer than seven days, configure transfer to Object Storage Service (OBS) or Log Tank Service (LTS) so that you can view them in OBS buckets or LTS log groups.
- After creating, modifying, or deleting a cloud service resource, you can query management traces on the CTS console 1 minute later and query data traces 5 minutes later.

Prerequisites

1. **Register with Huawei Cloud and complete real-name authentication.**

   If you already have a Huawei Cloud account, skip this step. If you do not have one, do as follows:

   a. Log in to the **Huawei Cloud official website**, and click **Sign Up** in the upper right corner.

   b. Complete the registration as prompted. For details, see **Registering with Huawei Cloud**.

      Your personal information page is displayed after the registration completes.

   c. Complete individual or enterprise real-name authentication by referring to **Real-Name Authentication**.

2. **Grant permissions for users.**

   If you log in to the console using a Huawei Cloud account, skip this step.

   If you log in to the console as an IAM user, first contact your CTS administrator (account owner or a user in the **admin** user group) to obtain the **CTS FullAccess** permissions. For details, see **Assigning Permissions to an IAM User**.

Viewing Traces

After you enable CTS and the management tracker is created, CTS starts recording operations on cloud resources. After a data tracker is created, CTS starts recording operations on data in Object Storage Service (OBS) buckets. CTS stores operation records (traces) generated in the last seven days.

This section describes how to query or export operation records of the last seven days on the CTS console.

Viewing Real-Time Traces in the Trace List of the New Edition

Step 1 Log in to the **CTS console**.

Step 2 In the navigation pane, choose **Trace List**.

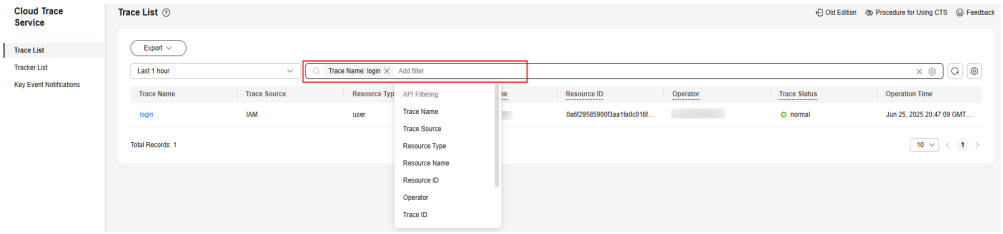
Step 3 In the time range drop-down list above the trace list, select a desired query time range: **Last 1 hour**, **Last 1 day**, or **Last 1 week**. You can also select **Custom** to specify a custom time range within the last seven days.

Step 4 The search box above the trace list supports advanced queries. Combine one or more filters to refine your search.

Table 14-2 Trace filtering parameters

Parameter	Description
Trace Name	Name of a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. For details about the operations that can be audited for each cloud service, see **Supported Services and Operations**. Example: **updateAlarm**
Trace Source	Cloud service name abbreviation. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. Example: **IAM**
Resource Name	Name of a cloud resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. If the cloud resource involved in the trace does not have a resource name or the corresponding API operation does not involve the resource name parameter, leave this field empty. Example: **ecs-name**
Resource ID	ID of a cloud resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. Leave this field empty if the resource has no resource ID or if resource creation failed. Example: *{VM ID}*
Trace ID	Value of the **trace_id** parameter for a trace reported to CTS. The entered value requires an exact match. Fuzzy matching is not supported. Example: **01d18a1b-56ee-11f0-ac81-******1e229**
Resource Type	Type of a resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. For details about the resource types of each cloud service, see **Supported Services and Operations**. Example: **user**

Parameter	Description
Operator	User who triggers a trace. Select one or more operators from the drop-down list. If the value of **trace_type** in a trace is **SystemAction**, the operation is triggered by the service and the trace's operator may be empty. For details about the relationship between IAM identities and operators and the operator username format, see **Relationship Between IAM Identities and Operators**.
Trace Status	Select one of the following options from the drop-down list: • **normal**: The operation succeeded. • **warning**: The operation failed. • **incident**: The operation caused a fault that is more serious than a normal failure, for example, causing other faults.
Enterprise Project ID	ID of the enterprise project to which a resource belongs. To check enterprise project IDs, go to the Enterprise Project Management Service (EPS) console and choose **Project Management** in the navigation pane. Example: **b305ea24-c930-4922-b4b9-******1eb2**
Access Key	Temporary or permanent access key ID. To check access key IDs, hover over your username in the upper right corner of the console and select **My Credentials** from the pop-up list. On the displayed page, choose **Access Keys** in the navigation pane. Example: **HSTAB47V9V*******TLN9**



Step 5 On the **Trace List** page, you can also export and refresh the trace list, and customize columns to display.

- Enter any keyword in the search box and press **Enter** to filter desired traces.
- Click **Export** to export all traces in the query result as an .xlsx file. The file can contain up to 5,000 records.
- Click to view the latest information about traces.
- Click to customize the information to be displayed in the trace list. If **Auto wrapping** is enabled ( ), excess text will move down to the next line; otherwise, the text will be truncated. By default, this function is disabled.

Step 6 (Optional) On the **Trace List** page of the new edition, click **Old Edition** in the upper right corner to switch to the **Trace List** page of the old edition.

----End

Viewing Traces in the Trace List of the Old Edition

Step 1 Log in to the **CTS console**.

Step 2 In the navigation pane, choose **Trace List**.

Step 3 Each time you log in to the CTS console, the new edition is displayed by default. Click **Old Edition** in the upper right corner to switch to the trace list of the old edition.

Step 4 In the upper right corner of the page, set a desired query time range: **Last 1 hour**, **Last 1 day**, or **Last 1 week**. You can also click **Customize** to specify a custom time range within the last seven days.

Step 5 Set filters to search for your desired traces.

Table 14-3 Trace filtering parameters

Parameter	Description
Trace Type	Select **Management** or **Data**. • Management traces record operations performed by users on cloud service resources, including creation, modification, and deletion. • Data traces are reported by OBS and record operations performed on data in OBS buckets, including uploads and downloads.
Trace Source	Select the name of the cloud service that triggers a trace from the drop-down list.
Resource type	Select the type of the resource involved in a trace from the drop-down list. For details about the resource types of each cloud service, see **Supported Services and Operations**.
Operator	User who triggers a trace. Select one or more operators from the drop-down list. If the value of **trace_type** in a trace is **SystemAction**, the operation is triggered by the service and the trace's operator may be empty. For details about the relationship between IAM identities and operators and the operator username format, see **Relationship Between IAM Identities and Operators**.

Parameter	Description
Trace Status	Select one of the following options: • **Normal**: The operation succeeded. • **Warning**: The operation failed. • **Incident**: The operation caused a fault that is more serious than a normal failure, for example, causing other faults.

Step 6 Click **Query**.

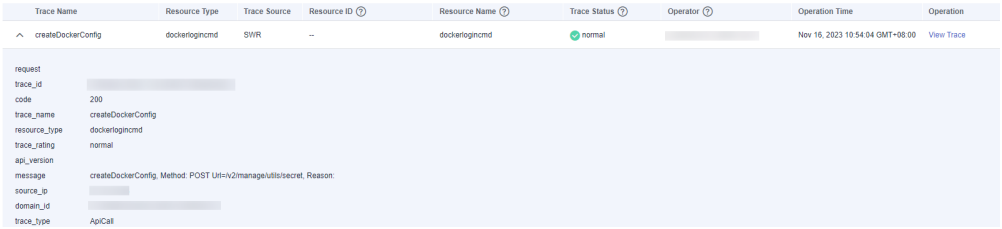
Step 7 On the **Trace List** page, you can also export and refresh the trace list.

- Click **Export** to export all traces in the query result as a CSV file. The file can contain up to 5,000 records.
- Click to view the latest information about traces.

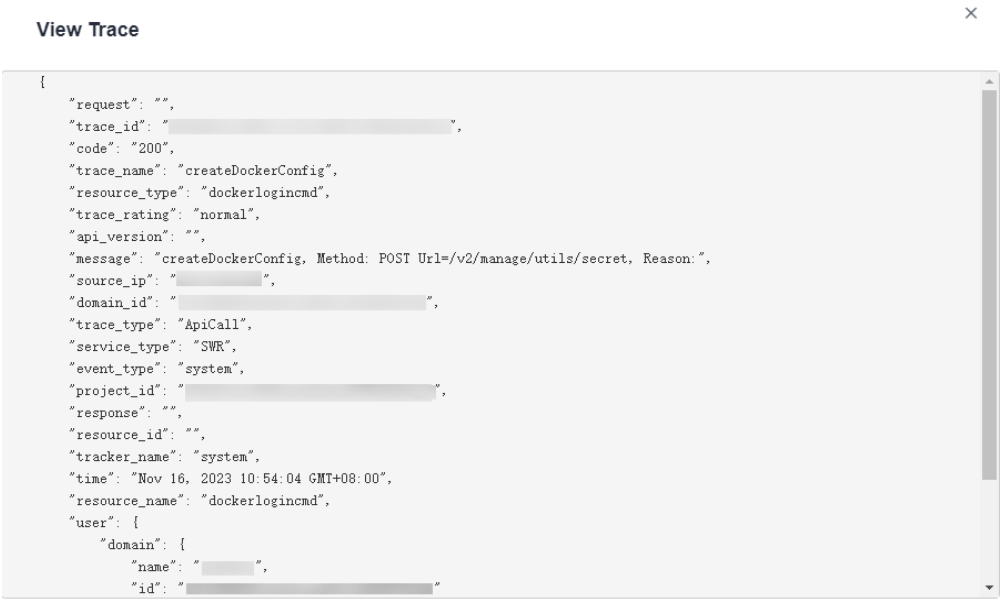
Step 8 In the **Tampered or Not** column of a trace, check whether the trace is tampered with.

- If no, **No** is displayed.
- If yes, **Yes** is displayed.

Step 9 Click on the left of a trace to expand its details.



Step 10 Click **View Trace** in the **Operation** column. The trace details are displayed.



Step 11 (Optional) On the **Trace List** page of the old edition, click **New Edition** in the upper right corner to switch to the **Trace List** page of the new edition.

----End

Helpful Links

- For details about the key fields in the trace structure, see **Trace Structure** and **Example Traces**.
- You can use the following examples to learn how to query a specific trace:
  - Use CTS to audit Elastic Volume Service (EVS) creation and deletion operations from the last two weeks. For details, see **Security Auditing**.
  - Use CTS to locate a fault or creation failure for an Elastic Cloud Server (ECS). For details, see **Fault Locating**.
  - Use CTS to check all operation records for an ECS. For details, see **Resource Tracking**.

