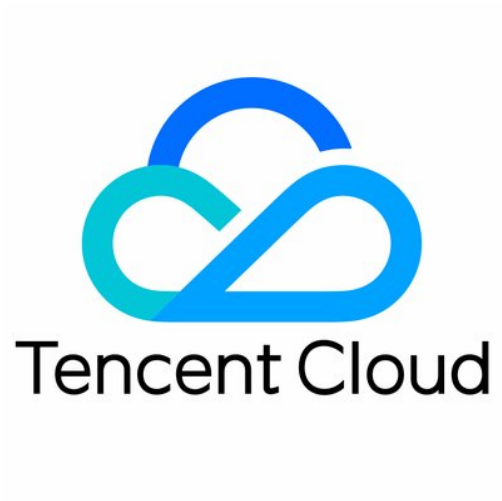
TDMQ for RabbitMQ

Operation Guide (Open-Source Managed Version)

Product Documentation



Contents

Operation Guide (Open-Source Managed Version)
Cluster Management
Creating Cluster

Last updated：2025-04-29 14:35:20

Overview

Cluster is a resource dimension in TDMQ for RabbitMQ. For different clusters, their vhosts, exchanges, and queues are completely isolated from each other. It is a common practice to use respective dedicated clusters for development, test, and production environments.
This document describes how to create an exclusive cluster in the TDMQ for RabbitMQ console.

Directions

1. Log in to RabbitMQ Console.
2. Choose **Cluster Management** > **Cluster List** in the left sidebar, click **Create Cluster** to proceed to the purchase page.
3. On the purchase page, select the target instance specification.

Parameter	Required	Description
Cluster Type	Yes	Select **Exclusive cluster**.
Billing Mode	Yes	TDMQ for RabbitMQ exclusive cluster offers monthly subscription and pay-as-you-go billing modes.
Region	Yes	Select a region close to resources of the deployed client. Cloud products in different regions are not interconnected over private networks and the region cannot be changed after you purchase the service. Proceed with caution.
RabbitMQ Version	Yes	Supports versions 3.8.30 and 3.11.8.
AZ	Yes	Select an AZ based on your business needs. Cross-AZ deployment is supported.
Node	Yes	Select an appropriate node specification according to your business needs. We

Specification		recommend you select our recommended node specification.
Node Count	Yes	Select an appropriate number of nodes based on your business needs.
Single-Node Storage Specification	Yes	Message storage is billed separately, and messages can be retained without limit in the purchased storage capacity. The actual storage usage can be estimated by multiplying the message production traffic by the retention period.
VPC	Yes	Bind the domain name of the new cluster's access point to the selected VPC.
Public network access	No	TDMQ for RabbitMQ provides 3 Mbps public network bandwidth by default. If you have higher bandwidth requirements, you can make an additional payment to purchase it. For details, see Billing Overview.
Mirrored Queue	No	We recommend that you enable mirrored queue to ensure the availability. After you enable it, a policy will be generated by default in the policy list in the instance details. You can delete it or customize a new policy to overwrite it. For more information, see Policy List.
Cluster Name	No	Enter Cluster Name, which should contain 3-64 characters, can only include digits, letters, hyphens (-), and underscores (_).
Image Queue	No	It is recommended to enable the image queue to ensure availability. Once enabled, a default policy is generated under Instance Details-Policy List, which can be deleted, or overwritten by a Custom New Policy. See Policy List.
Tag	No	Tags are used to categorize and manage resources. For more information, see Tag Overview.

4. Select **I have read and agree to TDMQ for RabbitMQ Terms of Service** and click **Buy Now**.

5. On the order payment page, click **Pay** and wait 3–5 minutes. Then, you can see the created cluster on the **Cluster** page.

View The Cluster Status

Last updated：2025-04-29 15:32:34

Overview

This document describes how to view the configuration information and health status of a cluster in the TDMQ for RabbitMQ console.

Directions

1. log in to RabbitMQ Console.
2. In the left navigation bar, select **Cluster** > **Cluster**, after selecting the region, click the ID of the target cluster, on the **Basic Info** page, you can view information such as the cluster's resource overview, health status, basic information, connection method, and Web console access address.

Health Status Description

TDMQ for RabbitMQ has an inspection program in place for each cluster, which checks the cluster's node status, disk utilization, memory utilization, and other metrics and displays different health status when these metrics exceed certain thresholds as detailed below:

Metric	Threshold (N)	Status Description
Node not started	-	Abnormal
Abnormal node connection	-	Abnormal
Available disk space	N ≥ 100 GB	Healthy
	50 MB ≤ N < 100 GB	Alarmed
	N < 50 MB	Abnormal
Memory utilization	N ≤ 60%	Healthy
	60% < N ≤ 90%	Alarmed
	N > 90%	Abnormal

Socket utilization	N ≤ 90%	Healthy
	N > 90%	Abnormal
Erlang process utilization	N ≤ 90%	Healthy
	N > 90%	Abnormal
FD utilization	N ≤ 90%	Healthy
	N > 90%	Abnormal

Upgrading Cluster

Last updated：2025-04-29 14:35:20

Overview

If the current cluster specifications cannot meet your business needs, you can increase the node specification, node quantity, single-node storage specification, and upgrade from a single availability zone to multiple availability zones in the console.

Note:

Currently only support upgrading. To downgrade node specifications, you need to submit a ticket. Downgrading node quantity or single-node storage specification is not supported.

Directions

1. Log in to the RabbitMQ console.
2. On the left sidebar, select **Cluster** > **Cluster**. Then, click **Upgrade** in the **Operation** column on the cluster list page. A pop-up will appear (taking monthly subscription clusters as an example).
3. Select the configuration change type: **node specification** or **disk space**.
4. Adjust the specific configuration items. The operation instructions are as follows:

Change Node Specification

Adjust Disk Space

Target Node Specification: Changing the node specification and the number of nodes will involve resource changes and open-source console restarts. It is recommended to configure the multi-node mirror queue in the console before proceeding with the upgrade to avoid service interruptions and data loss during the upgrade process.

Target node count: Can be upgraded to 3, 5, or 7 node counts.

Notes:

Upgrade from single availability zone to multiple availability zones:

When the number of nodes increases from 1 to 3 or above, and the cluster region supports multiple availability zones, you can choose to configure the cluster to upgrade from a single availability zone to multiple availability zones.

In this way, you can achieve cross-AZ disaster recovery deployment for the cluster and enhance cluster stability.

Single-node storage specification: After adjusting the single-node storage, the new storage specification takes effect on all nodes in the cluster.

Deleting Cluster

Last updated：2025-04-29 14:35:20

Overview

You can delete a TDMQ for RabbitMQ cluster if you no longer need it.
The lifecycle of a TDMQ for RabbitMQ cluster is the process the cluster goes through from start to release. By properly managing the cluster lifecycle, you can ensure that the applications running in the cluster can provide services efficiently and economically. A cluster has the following status:

Status	Status Attribute	Status Description
Creating	Intermediate status	The cluster is being created.
Running	Stable status	The cluster is running normally, indicating that your node status, disk utilization, and other metrics are within the normal range.
Deleting	Intermediate status	The cluster is being deleted in the console or via APIs.
Isolated	Intermediate status	The cluster is in the 7-day isolation period after overdue payment. Production and consumption cannot be performed in a cluster in the isolated status, but the data and configurations saved in the cluster will not be deleted.
Failed to create	Intermediate status	The cluster is purchased with fees successfully deducted in the console or via APIs but fails to be assigned. In this case, contact us for assistance.
Deletion failed	Stable status	TDMQ for RabbitMQ fails to release resources after the cluster is manually deleted or isn't renewed within 7 days after expiration.

Directions

Manual deletion

1. Log in to the RabbitMQ console.
2. On the left sidebar, select **Cluster** > **Cluster**. Then, click **More** > **Terminate** in the **Operation** column on the cluster list page.
3. In the Delete Confirmation pop-up window, delete the cluster after double confirmation. .

Note:

After a cluster is deleted, all its configurations will be cleared and cannot be recovered. Therefore, proceed with caution.

Automatic deletion upon expiration or overdue payment

Monthly subscribed instances can be retained in the TDMQ for RabbitMQ console for up to 7 calendar days after they expire or have overdue payments. You can continue to use them if you renew them within 7 days after expiration. For details, see Payment Overdue.

If the TDMQ for RabbitMQ instance is not renewed within 7 days (inclusive) after expiration, its resources will be released at midnight on the 8th day, and data will be cleared and cannot be restored.

Note:

Production and consumption cannot be performed in a cluster in the isolated status, but the data and configurations saved in the cluster will not be deleted.

For clusters in the 7-day isolation status, you can go to the console to renew them by clicking **Renew** in the **Operation** column on the cluster list page. Successfully renewed clusters can go back to the running status and be normally used.

Accessing Native Console

Last updated：2024-06-26 15:56:25

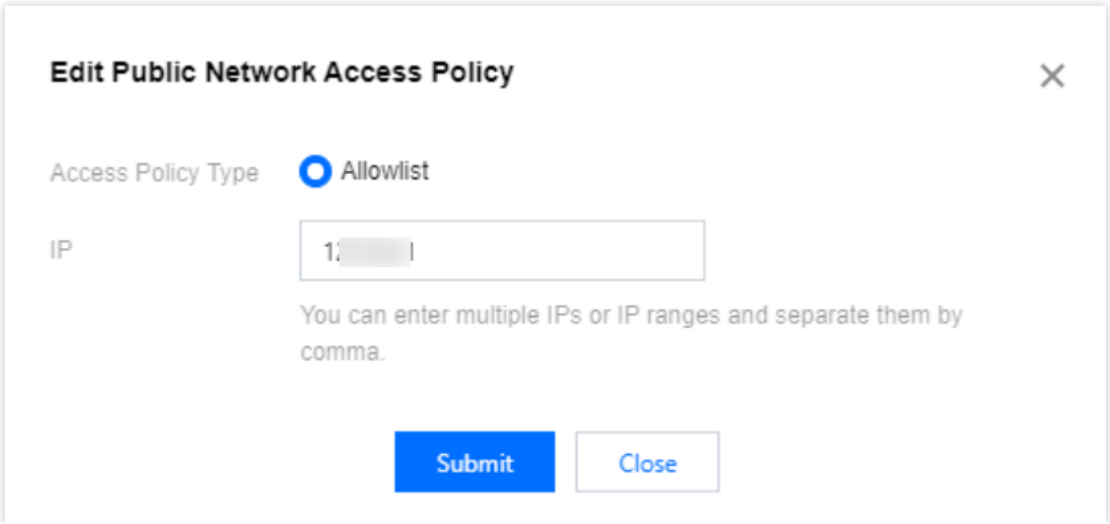
Overview

This document describes how to access the open-source RabbitMQ console in the TDMQ for RabbitMQ console.

Directions

1. Log in to the RabbitMQ console.
2. Select **Cluster** > **Cluster** on the left sidebar, select a region, and click the ID of the target cluster to enter the **Basic Info** page of the cluster.
3. In the **Web Console Access Address** module, you can perform the following operations:
Enable and disable the public or private network access address.
Log in to the open-source RabbitMQ console by using the **Public Network Access Address**, username, and password.
Click **Modify** next to the public network access policy to set the console access allowlist.
You can enter multiple IPs or IP ranges and separate them by comma.
All users are denied access to the console by default.

Note:
Only /28–/32 subnet masks can be configured for IP ranges. If you want to use the /24 mask, submit a ticket for assistance.



Add Network Access Policy
VPC Network Access

Last updated：2025-04-29 14:38:51

Overview

When you purchase a cluster, you select a private network and the corresponding VPC environment (for example, VPC A). This means that only the selected VPC A can access your TDMQ for RabbitMQ service (production data, consumption data, etc.). If you find that other VPC environments (for example, VPC B) need to access the TDMQ for RabbitMQ service in VPC A during subsequent use, you can configure the access method and select the routing policy of the VPC network.

Directions

1. Log in to the TDMQ for RabbitMQ console.
2. Select **Cluster Management** > **Cluster List** in the left sidebar. After selecting a region, click the ID of the target cluster to enter the cluster basic information page.
3. In the **Client Access** module, click **Add Network Access Policy** in the upper right corner.
4. In the pop-up window, select **VPC** as the route type and choose the VPC and subnet.

Notes:

1. When selecting VPC Network Access, you can specify a designated IP. When changing the access method, the IP can remain unchanged by specifying a designated IP.
2. IP addresses ending with ".1" or ".255" are not allowed to use, as the two are generally the default gateway and broadcast address of the subnet respectively.

5. Click Submit to complete the strategy addition.

Public Domain Name Access

Last updated：2025-04-29 14:40:31

Application Scenario

TDMQ for RabbitMQ uses internal network transmission by default. When your consumers or producers are located in a self-built IDC or other cloud services, you can access the data of TDMQ for RabbitMQ via the public network to perform production and consumption. At this point, you need to enable a separate public network route.
TDMQ for RabbitMQ currently provides 3 Mbps of free public network bandwidth by default. If you have higher bandwidth requirements, you can make an additional payment to purchase it. For specific prices, please see the Billing Overview.
This document introduces the directions to enable public network routes, adjust public network bandwidth configurations, and delete public network bandwidth in the TDMQ for RabbitMQ console.
Notes:
A cluster can have only one public network route.

Directions

Enable Public Network Access and Edit Public Network IP

1. Log in to the TDMQ for RabbitMQ console.
2. Select **Cluster Management** > **Cluster List** in the left sidebar. After selecting a region, click the ID of the target cluster to enter the cluster basic information page.
3. If you have enabled public network access, you can see the specific public network bandwidth in the **Basic Information** module. If you have not enabled public network access, see Step 4.

4. In the **Basic Information** module, click **Enable** next to the public network.

5. In the pop-up window, select the necessary **public network bandwidth**, click **Confirm** and complete the payment. Then return to the console. The public network will be shown as enabling. Wait for the public network to be fully enabled.

6. After the public network is enabled, in the **Client Access** module, you can see that one line of **Public Domain Name Access** already displays information such as public network bandwidth and IP.

7. You can click the **Modify** button under the **Access Policy** attribute of Public Domain Name Access to edit the IP addresses in the public network access policy. The policy supports multiple IPs, with IPs separated by commas, and supports entering IPs and IP ranges.

Notes:
Configuration of IP address range currently only supports subnet masks from /28 to /32. For a /24 mask, submit a ticket to apply for whitelisting.
IP cannot be left blank or set to 0.0.0.0.

Adjust Public Network Bandwidth Configuration

1. On the cluster basic information page, click **configuration adjustment** next to the public network bandwidth in the basic information module.

2. In the pop-up window, modify the public network bandwidth, click **Confirm** to complete the public network bandwidth configuration adjustment.

Disable Public Network (Delete Public Network Route)

Notes:
Only when the public network bandwidth is 3 Mbps (without additional purchase of public network bandwidth), can you disable public network access. Meanwhile, the public network route will also be deleted.
1. Enter the cluster basic information page. In the **basic information** or **client access** module, click **configuration adjustment** next to the public network bandwidth. First, adjust the public network bandwidth to 3 Mbps.
2. When the public network bandwidth is 3 Mbps, in the **Basic Information** module, click **Disable** next to the public network bandwidth to disable public network access.

Public Network Bandwidth Management

Last updated：2024-09-29 10:49:15

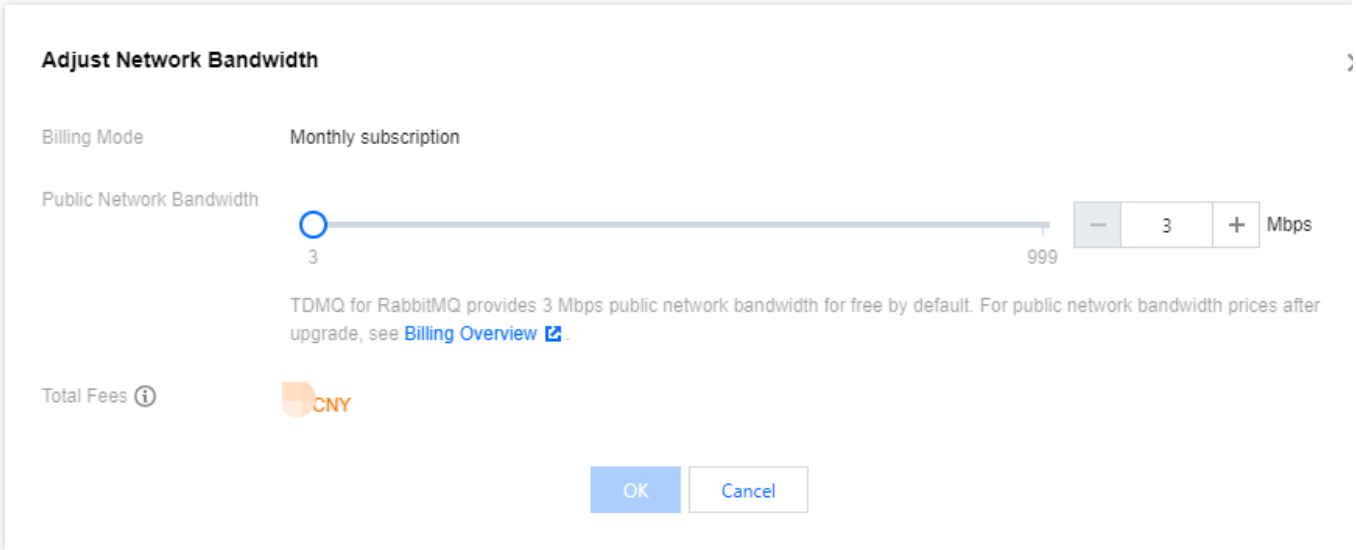
Overview

TDMQ for RabbitMQ uses private network transmission by default. To access it through the public network, you need to open an independent public network route. For detailed directions, see Add Routing Policy. Free public network bandwidth of 3 Mbps is provided by default.
If you require higher bandwidth, TDMQ for RabbitMQ supports public network bandwidth upgrade. You can pay additional fees to purchase it. For detailed pricing, see Billing Overview.
This document describes how to adjust the public network bandwidth configuration and delete public network bandwidth in the TDMQ for RabbitMQ console.

Directions

Adjusting Public Network Bandwidth Configuration

1. Log in to the RabbitMQ console.
2. In the left sidebar, choose **Cluster**, select a region, and click the ID of the target cluster to enter the basic cluster information page.
3. In the **Basic Info** module, click **Adjust Configuration** next to the Public Network Bandwidth.



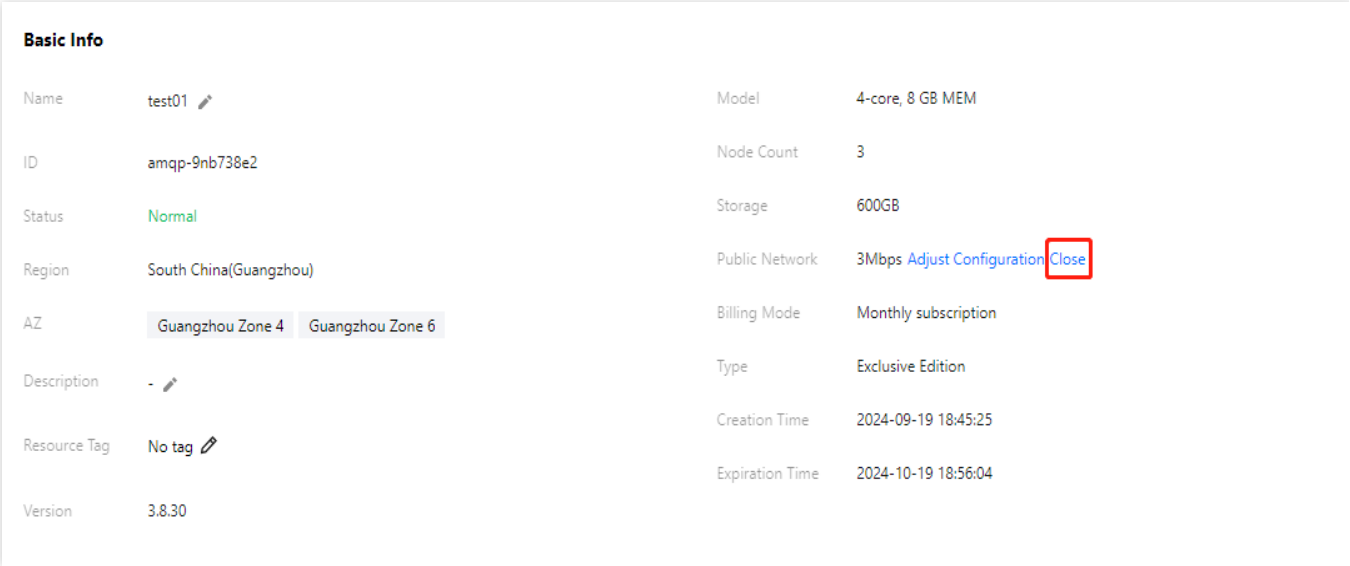
4. In the pop-up window, modify the public network bandwidth and click **OK**. The public network bandwidth configuration is adjusted.

Deleting Public Network Routes

Note:

Public network access can only be disabled when the public network bandwidth is 3 Mbps (without any additional purchased bandwidth), and the public network routing will also be deleted.

1. Log in to the RabbitMQ Console.
2. In the left sidebar, choose **Cluster** . After selecting the region, click the ID of the target cluster to enter the cluster's basic information page.
3. In the **Basic Info** or **Client Access** module, click **Adjust Configuration** next to the public network bandwidth. First, adjust the public network bandwidth to 3 Mbps.
4. Once the public network bandwidth is set to 3 Mbps, in the **Basic Info** module, click **Close** next to the public network bandwidth to disable public network access.



Connecting to Prometheus

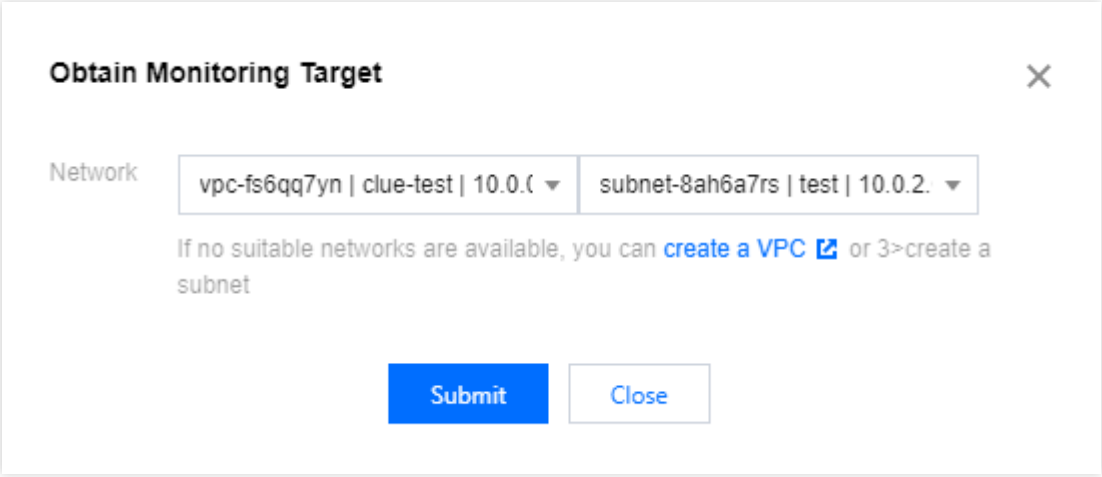
Last updated：2024-06-26 15:56:25

Overview

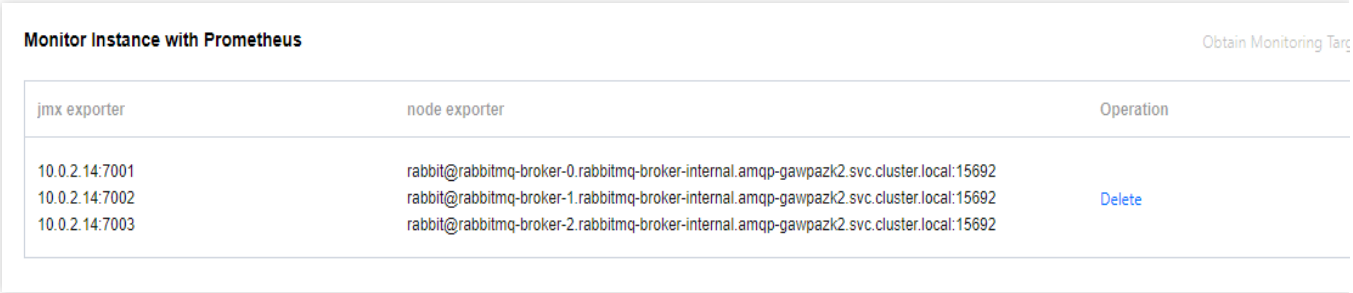
TDMQ for RabbitMQ clusters currently provide Prometheus to scrape the metric information of nodes, including basic monitoring metrics such as queue, channel, and connection, as well as the metrics exposed by the broker's JMX.

Directions

1. Log in to the RabbitMQ console.
2. Select **Cluster** > **Cluster** on the left sidebar, select a region, and click the ID of the target cluster to enter the **Basic Info** page of the cluster.
3. Click **Obtain Monitoring Target** in the top-right corner of the **Other Information** > **Monitor Instance with Prometheus** module and select the VPC and subnet.



4. Click **Submit** to get the set of monitoring targets.



5. Download Prometheus and configure the monitoring scrape address.

5.1 Enter the directory of the Prometheus package and run the following command to decompress it.

```
tar -vxf prometheus-2.30.3.linux-amd64.tar.gz
```

5.2 Modify the `prometheus.yml` configuration file by adding the `jmx_exporter` and `node_exporter` scrape tasks.

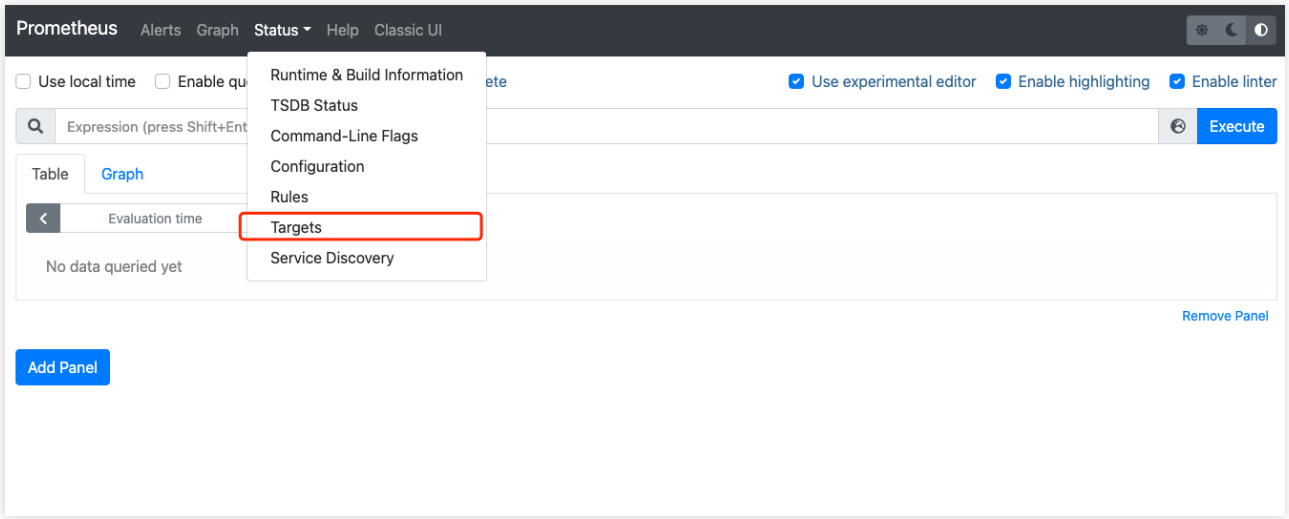
```
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped fro
  - job_name: "prometheus"
    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.
    static_configs:
      - targets: ["localhost:9090"]

  - job_name: "broker-jmx-exporter"
    scrape_interval: 5s
    metrics_path: '/metrics'
    static_configs:
      - targets: ['10.x.x.0:60001','10.x.x.0:60003','10.x.x.0:60005']
        labels:
          application: 'broker-jmx'
  - job_name: "broker-node-exporter"
    scrape_interval: 10s
    metrics_path: '/metrics'
    static_configs:
      - targets: ['10.x.x.0:60002','10.x.x.0:60004','10.x.x.0:60006']
        labels:
          application: 'broker-node'
```

Here, `broker-jmx-exporter` is the tag configured for the `jmx` metric of the broker scraped by Prometheus, `Targets` contains the information of the mapped port, `broker-node-exporter` is the tag configured for the basic metrics of the node of the scraped broker, and `scrape_interval` is the frequency of scraping metric data.

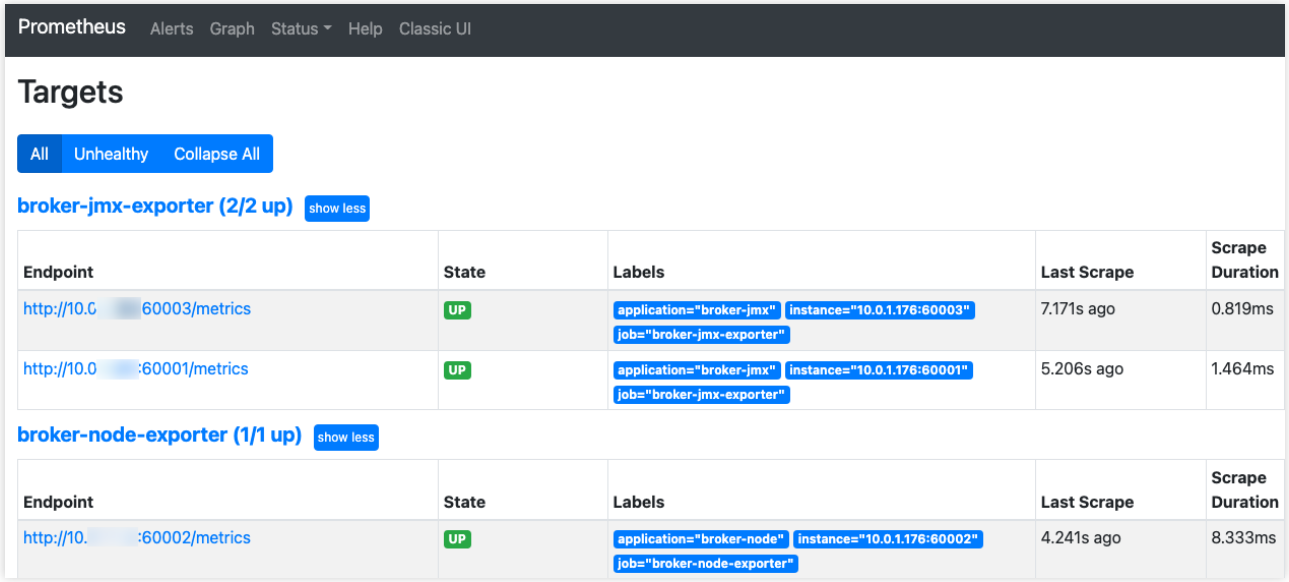
5.3 Start Prometheus.

```
./prometheus --config.file=prometheus.yml --web.enable-lifecycle
```

5.4 Open the UI provided by Prometheus to check whether the connected targets are normal by entering `http://localhost:9090` in the browser for example.



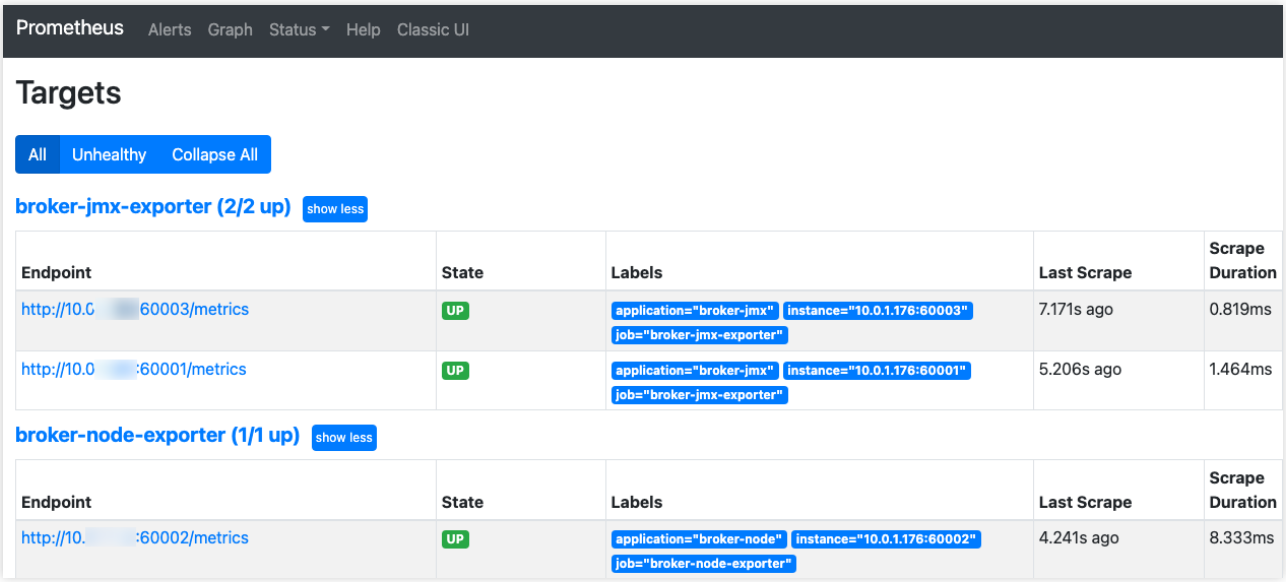
5.5 Confirm that all targets are in `UP` status.



If the targets are in `DOWN` status, check whether the network is accessible or check the `Error` option at the end of the status bar for the cause.

5.6 Query the monitoring metric data.

Click **Graph**, enter a metric name such as `node_memory_MemAvailable_bytes` , and click **Execute** to view the monitoring data.



Node Management

Last updated：2024-06-26 15:56:25

Overview

The node management page displays the list of all nodes in the current cluster and node status metrics.
This document describes how to view the node list in the TDMQ for RabbitMQ console.

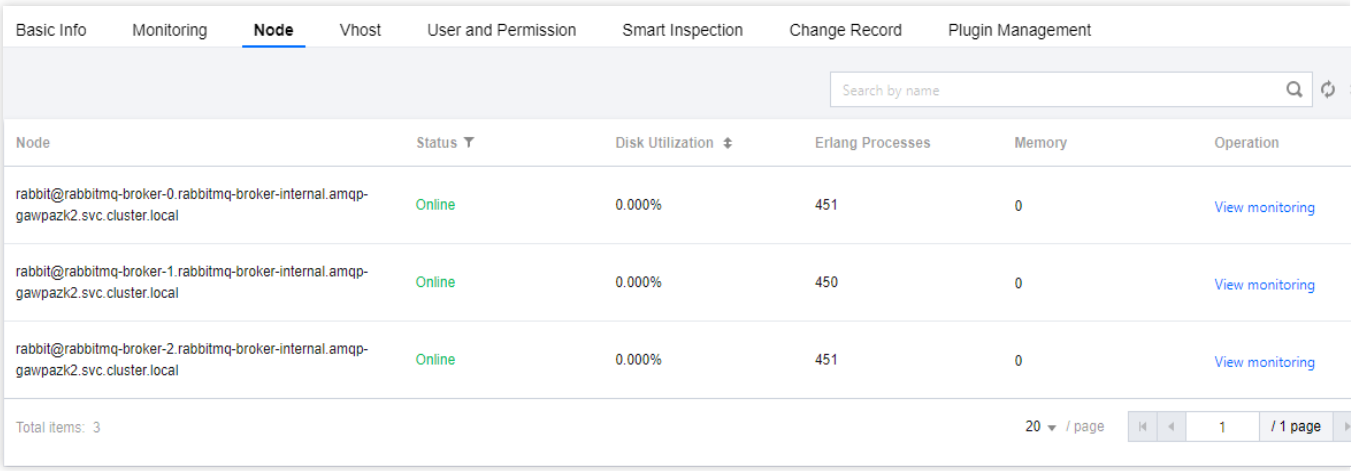
Prerequisites

You have created a cluster as instructed in Cluster Management.

Directions

Viewing a node

1. Log in to the RabbitMQ console.
2. Select **Cluster** > **Cluster** on the left sidebar, select a region, and click the ID of the target cluster to enter the **Basic Info** page of the cluster.
3. In the **Basic Information** tab, by selecting the top **Node** Tab, you can view the node information of the current cluster.



Note:

On the node list page, click **View monitoring** in the **Operation** column to view the detailed monitoring information of the corresponding node.

Vhost Management

Last updated：2025-04-29 11:17:44

Overview

Virtual host (vhost) is a resource management concept in TDMQ for RabbitMQ. It is used for logical isolation. Exchanges and queues of different vhosts are isolated from each other.
Generally, different business scenarios can be isolated by vhost and configured with dedicated settings, such as message retention period.
This document describes how to create multiple vhosts in TDMQ for RabbitMQ so as to use the same TDMQ for RabbitMQ cluster in different scenarios.
Note：
Exchange and queue names must be unique in the same vhost.

Prerequisites

You have created a cluster as instructed in Cluster Management.

Directions

Creating vhost

1. Log in to the RabbitMQ console.
2. In the left sidebar, choose **Cluster Management > Vhost,** click **Create** to enter the Create Vhost page.
3. In the **Create Vhost** window, configure the vhost attributes:
Vhost Name: Enter the vhost name, which cannot be modified after creation and can contain 1–64 letters, digits, "-", and "_".
Image queue: It is recommended to enable the image queue to ensure availability. This image queue can replicate the messages across multiple nodes in the RabbitMQ cluster, ensuring that messages in the queue are not lost in case of a node failure. (It cannot be enabled in a single-node cluster.) For specific parameters, see Default Mirroring Policy.
Remarks: Enter the vhost remarks.
4. Click **Submit**.

Next steps: You can create an exchange and queue in the vhost to produce and consume messages.

Viewing a vhost

On the vhost list page, click the ID of the target vhost to enter the **Basic Info** page, which is divided into two modules:
Overview
Queue Count: The number of queues under the current vhost.
Exchange Count: The number of exchanges under the current vhost.
Channel Count: The number of channels under the current vhost.
User Count: The number of users under the current vhost.
Number of heaped messages: The number of the heaped messages under the current Vhost.
Production rate: The production rate of the current Vhost.
Consumption rate: The consumption rate of the current Vhost.

Connection List
The information of connections under the current vhost and the number of channels of each connection.

Modifying a vhost

The trace plugin can be toggled on/off on the vhost list page.
The Trace plugin can be enabled and disabled on the Vhost list page. It is recommended to use the Trace plugin for small-traffic verification or troubleshooting scenarios, and it is not advised to enable it when the cluster's TPS exceeds 10,000. Please readMessage Query.

1. On the **Vhost** list page, click **Edit** in the **Operation** column of the target vhost to enter the editing page.
2. Modify the description and click **Submit**.

Deleting a vhost

You can modify the description of a vhost in the following steps:
1. On the **Vhost** list page, click **Delete** in the **Operation** column.
2. In the deletion confirmation pop-up window, click **Delete**.
Note:
After a vhost is deleted, all the configurations under it will be cleared and cannot be recovered.

Exchange Management

Last updated：2024-09-10 14:51:00

Overview

A producer sends a message to an Exchange, which subsequently routes the message to one or more queues based on its attributes or content (or discards it). Then, a consumer pulls it from the Queue to consume it.
This document describes how to create, delete, and query an Exchange in the TDMQ for RabbitMQ console.

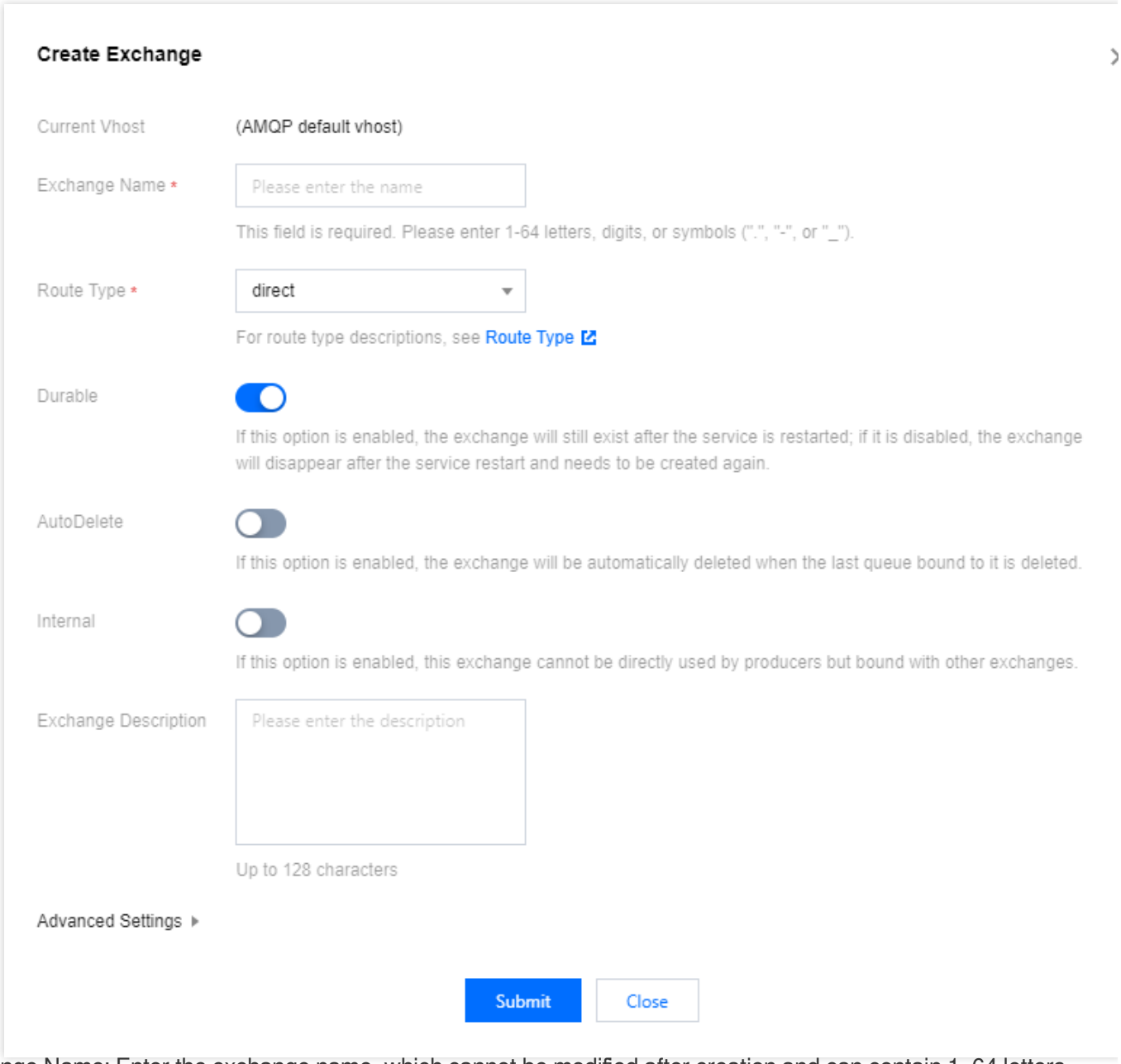
Prerequisites

You have created a corresponding vhost as instructed in Vhost Management.

Directions

Creating an Exchange

1. Log in to the RabbitMQ console。
2. In the left sidebar, choose **Cluster Management >****Exchange** tab, select a Vhost, and click **Create** to enter the Creating Exchange page.
3. In the **Create Exchange** dialog box, enter the following information:



Exchange Name: Enter the exchange name, which cannot be modified after creation and can contain 1–64 letters, digits, hyphens, and underscores.

Routing Type: Select a routing type (direct, fanout, topic, or header), which cannot be changed after creation. For more information, see Exchange.

Direct: A direct Exchange will route messages to the Queue with the BindingKey exactly matching the RoutingKey.

Fanout: A fanout Exchange will route messages to all queues bound to it.

Topic: A topic Exchange supports multi-condition match and fuzzy match, which means that messages will be routed to the Queue bound to it by using routing key pattern match and string comparison.

Header: A header Exchange has nothing to do with Routing Key and matches messages by the `Headers` attribute. Before the Queue is bound to a Headers Exchange, declare a map key-value pair to implement the binding between the message queue and Exchange. When a message is sent to RabbitMQ, the `Headers` attribute of the message

will be obtained to match the key-value pair specified during the Exchange binding, and the message will be routed to the queue only if there is a full match.

Durable: If this option is set to `true`, the Exchange will still exist after the service is restarted; if it is set to `false`, the exchange will disappear after the service is restarted and needs to be created again.

AutoDelete: If this option is set to `true`, when the last queue bound to the Exchange is deleted, the Exchange will be deleted automatically.

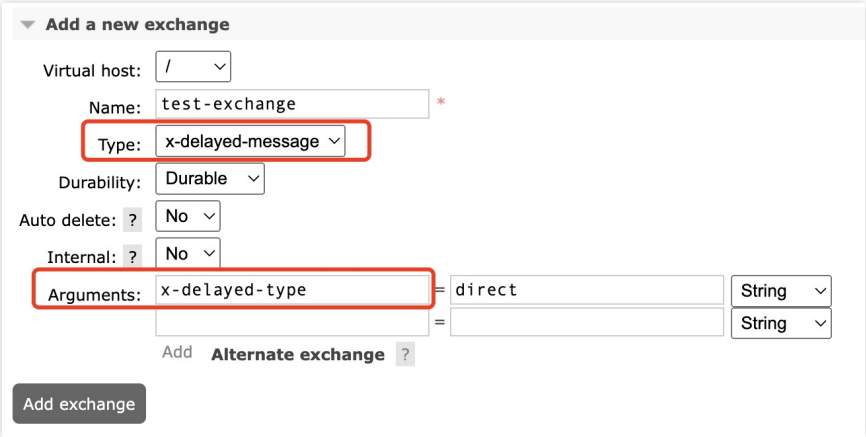
Internal: If this option is set to `true`, the Exchange cannot be directly used by Producers but can only be bound to other Exchanges.

Exchange Description: Enter the exchange description of up to 128 characters.

Add Alternate Exchange: It is optional, and no alternate Exchange is used by default. Messages that are sent to the primary Exchange but cannot be routed will be sent to the alternate Exchange specified here.

Note:

The console will soon support the X-Delayed-Message routing type. If you need to use it, first create it in the open-source console - Exchanges (For detailed directions on how to log in and access the RabbitMQ open-source console, see Accessing Native Console). When creating it, you need to specify the x-delayed-type for Arguments, as shown in the example below:++



Note that the X-Delayed-Message type is not a built-in type of RabbitMQ. It requires the rabbitmq_delayed_message_exchange plugin to be enabled. For details, see Plugin Management.

4. Click **Submit**, and you can see the created exchange in the exchange list.

Editing an exchange

1. In the exchange list, click **Edit** in the **Operation** column of the target exchange.
2. In the pop-up window, you can edit the exchange description.
3. Click **Submit**.

Deleting an exchange

1. In the exchange list, click **Delete** in the **Operation** column of the target exchange.
2. In the pop-up window, click **Delete**.

Note:

After an exchange is deleted, all the configurations under it will be cleared and cannot be recovered.

Queue Management

Last updated：2025-04-29 11:17:44

Overview

A queue is used to store messages. Each message will be put into one or more queues. Producers produce messages and deliver them to queues, and consumers pull messages from queues for consumption.
Multiple consumers can subscribe to the same queue. In this case, messages in the queue will be evenly distributed to such consumers for processing, rather than making each consumer receive and process all the messages.
This document describes how to create, delete, and query a queue in the TDMQ for RabbitMQ console.

Prerequisites

You have created a vhost as instructed in Vhost Management > Creating a vhost.

Directions

Creating a queue

Note:
The creation of regular queues or quorum queues is supported. You can click the following tabs to view the creation methods of different types of queues.
regular queue: Primarily used for high-performance, low-delay scenarios, suitable for situations with high message throughput.
Quorum Queue: Focus on data security and consistency, suitable for scenarios with high requirements for message reliability.
Regular queue
Quorum queue
1. Log in to the RabbitMQ console.
2. In the left sidebar, select **Cluster** > **Queue** tab, choose a Vhost, then click **Create** to enter the Create Queue page.
3. Enter the basic information of the queue.

Queue Name: Enter the queue name, which cannot be modified after creation and can contain 3–64 letters, digits, hyphens, and underscores.
Type: Regular queue.

Durable: Set whether the queue performs persistence.
Node: Select the node where the queue resides.
AutoDelete: After this feature is enabled, the queue will be deleted immediately after its last consumer unsubscribes from it.
Queue Remarks: Enter the queue remarks of up to 128 characters.
4. Click **Next** to configure common parameters.

Message TTL: Messages in the queue will be discarded/sent to the dead letter exchange after the specified time elapses.
Auto expire: The queue will be deleted if it is not used (accessed) within the specified time.
Max length: The maximum number of messages that the queue can contain.
Max length bytes: The upper limit of the queue's storage capacity. If it is reached, messages will be processed according to `Overflow behaviour` .
Overflow behaviour: When the queue capacity reaches the upper limit, the message at the beginning of the queue will be discarded.
drop-head: Discard the messages at the head of the queue when the queue reaches its capacity limit.
reject-publish: When the queue reaches its capacity limit, reject the release of new messages and mark the release operation as failed.
reject-publish-dlx: When the queue reaches the capacity limit, reject the publishing of new message and send the messages to the Dead Letter Exchange (dlx).
Dead Letter Exchange: A message will be delivered to the dead letter exchange if it is not acknowledged within the TTL.
5. Click **Next** to configure other advanced options.

Single active consumer: If this option enabled, it is necessary to ensure that there must be only one consumer consuming from the queue at a time.
Maximum priority: Configure the maximum priority of messages in this queue.
Lazy mode: If this option is enabled, the queue will preferably store pushed messages on the disk to reduce the memory usage.
Master locator: Configure the allocation method of the node where the master is located if the mirrored queue is configured.
min-masters: The node hosting the minimum number of masters will be selected as the node where the master of the current queue is located if the mirrored queue is configured.
client-local: The node the client that declares the queue is connected to will be selected as the node where the master of the current queue is located if the mirrored queue is configured.
random: A random node will be selected as the node where the master of the current queue is located if the mirrored queue is configured.

6. Click **Submit**.

1. Log in to the RabbitMQ console.
2. In the left navigation bar, Select **Cluster** > **Queue** tab, choose a vhost, and click **Create** to enter the **Create Queue** page.
3. Enter the basic information of the queue.

Queue Name: Enter the queue name, which cannot be modified after creation and can contain 3–64 letters, digits, hyphens, and underscores.
Type: Quorum queue.
Node: Select the node where the queue resides.
Queue Remarks: Enter the queue remarks of up to 128 characters.
4. Click **Next** to configure common parameters.

Auto expire: The queue will be deleted if it is not used (accessed) within the specified time.
Max length: The maximum number of messages that the queue can contain.
Max length bytes: The upper limit of the queue's storage capacity. If it is reached, messages will be processed according to `Overflow behaviour` .
Delivery Limit: The allowed retry count when message delivery in the queue fails.
Overflow behaviour: When the queue capacity reaches the upper limit, the message at the beginning of the queue will be discarded.
drop-head: Discard the messages at the head of the queue when the queue reaches its capacity limit.
reject-publish: When the queue reaches its capacity limit, reject the release of new message and mark the release operation as failed.
Dead letter policy: Options include at-most-once and at-least-once. The at-least-once can only be selected when the overflow behavior is set to reject-publish.
Dead letter Exchange: A message will be delivered to the dead letter exchange if it is not acknowledged within the TTL.
5. Click **Next** to configure other advanced options.

Single active consumer: If this option enabled, it is necessary to ensure that there must be only one consumer consuming from the queue at a time.
Max in memory length: The maximum number of messages in the quorum queue memory.
Max in memory bytes: The maximum total message size (in bytes) in the quorum queue.
Initial Cluster Size: The initial cluster size of the Quorum queue.
Leader Locator: Options include client-local and balanced. If network latency significantly affects performance, client-local policy can be selected; if you need to balance the load across nodes, balanced policy can be selected.
6. Click **Submit**.

Viewing queue details

In the **Queue** list, click the ID of the target queue to view its details.

You can view:

Basic Info: Queue type, online consumer, dead letter exchange, and AutoDelete status. Click **More Advanced Options** to view all parameter settings of the queue.

Consumer list: Information of consumers subscribed to this queue.

Viewing binding

Binding relationship: The routing relationships bound to this Queue.

Editing a queue

1. In the queue list, click **Edit** in the **Operation** column of the target queue.
2. In the pop-up window, edit the queue information.
3. Click **Submit**.

Deleting a queue

1. In the queue list, click **Delete** in the **Operation** column of the target queue.
2. In the pop-up window, click **Delete**.

Note:

After a queue is deleted, all the configurations under it will be cleared and cannot be recovered.

Binding

Last updated：2024-09-10 14:49:23

Overview

This document describes how to establish or cancel a binding between an exchange and a queue in the TDMQ console.

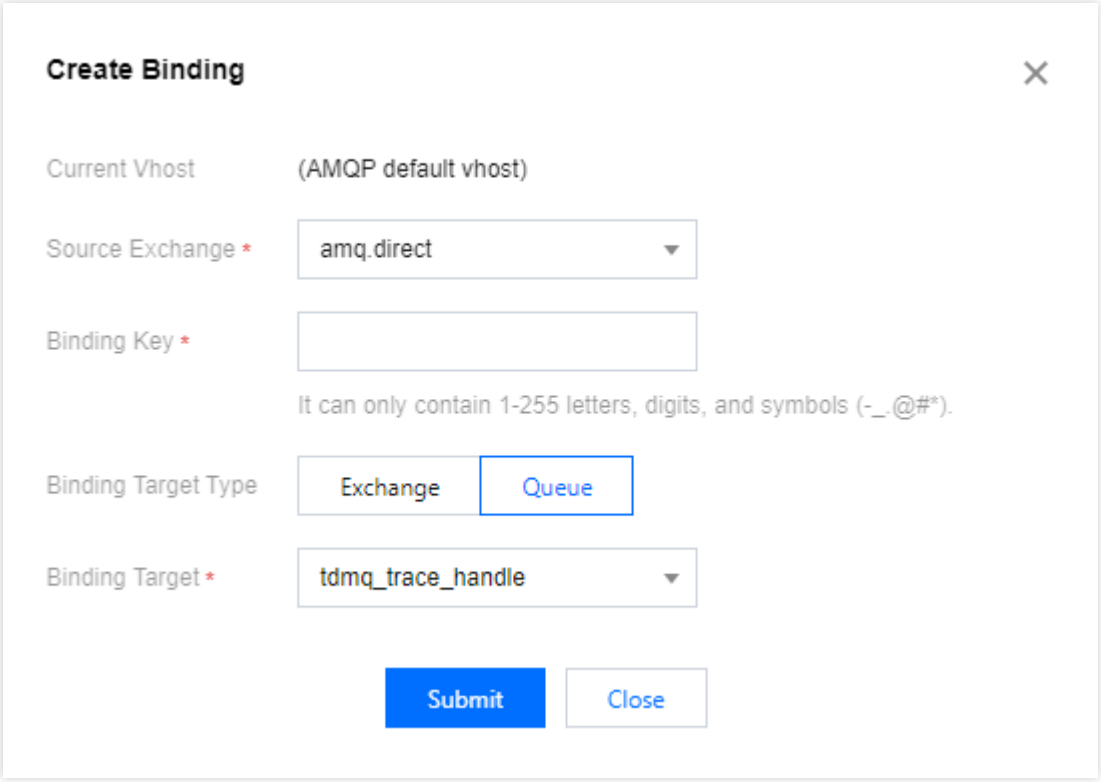
Prerequisites

You have created an exchange as instructed in Exchange Management.
You have created a queue as instructed in Queue Management.

Directions

Creating a binding

There are multiple entries to bind routing relationships:
Entry 1: In the left sidebar, choose **Cluster Management > Exchange** list, and click **bind routing** in the operation bar of the target Exchange.
Entry 2: In the left sidebar, choose **Cluster Management > Queue** list, and click **bind routing** in the operation bar of the target Queue.
Entry 3: In the left sidebar, choose **Cluster Management > Vhost** , and click the ID of the target Vhost. At the top of the page, select **Exchange** or **Queue** tab, and click **bind routing** in the operation bar of the target Queue.
In the pop-up window for creating a binding relationship, set the source Exchange, binding Key, binding target type and binding target, and click **Submit** to complete the binding process.



Unbinding

1. In the binding list, click **Unbind** in the **Operation** column of the target binding.
2. In the pop-up window, click **Delete**.

Note:

Once deleted, the route will no longer be available and cannot be recovered.

User and Permission Management

Last updated：2024-10-18 17:37:45

Glossary

User is the smallest unit for dividing permissions within a TDMQ for RabbitMQ cluster. You can grant users the permissions of configuration and read/write under different vhosts.
User password is an authentication method. You can add a username and password in a client to access TDMQ for RabbitMQ clusters for message production/consumption.
Permission refers to your operation permission for exchanges and queues under a vhost, including configuration and read/write. The configuration permission controls declaring and deleting exchanges and queues. The read/write permission control reading messages from queues, sending messages to exchanges, and binding queues to exchanges.

Use Limits

There can be up to 20 users in a cluster.

Use Cases

You need to securely use TDMQ for RabbitMQ to produce/consume messages.
You need to set production/consumption permissions of different vhosts for different users.
For example, your company has departments A and B, and department A's system produces transaction data and department B's system performs transaction data analysis and display. In line with the principle of least privilege, two users can be created to grant department A only the permission to produce messages to the transaction system vhost and grant department B only the permission to consume messages. This helps greatly avoid problems caused by unclear division of permissions, such as data disorder and dirty business data.

Directions

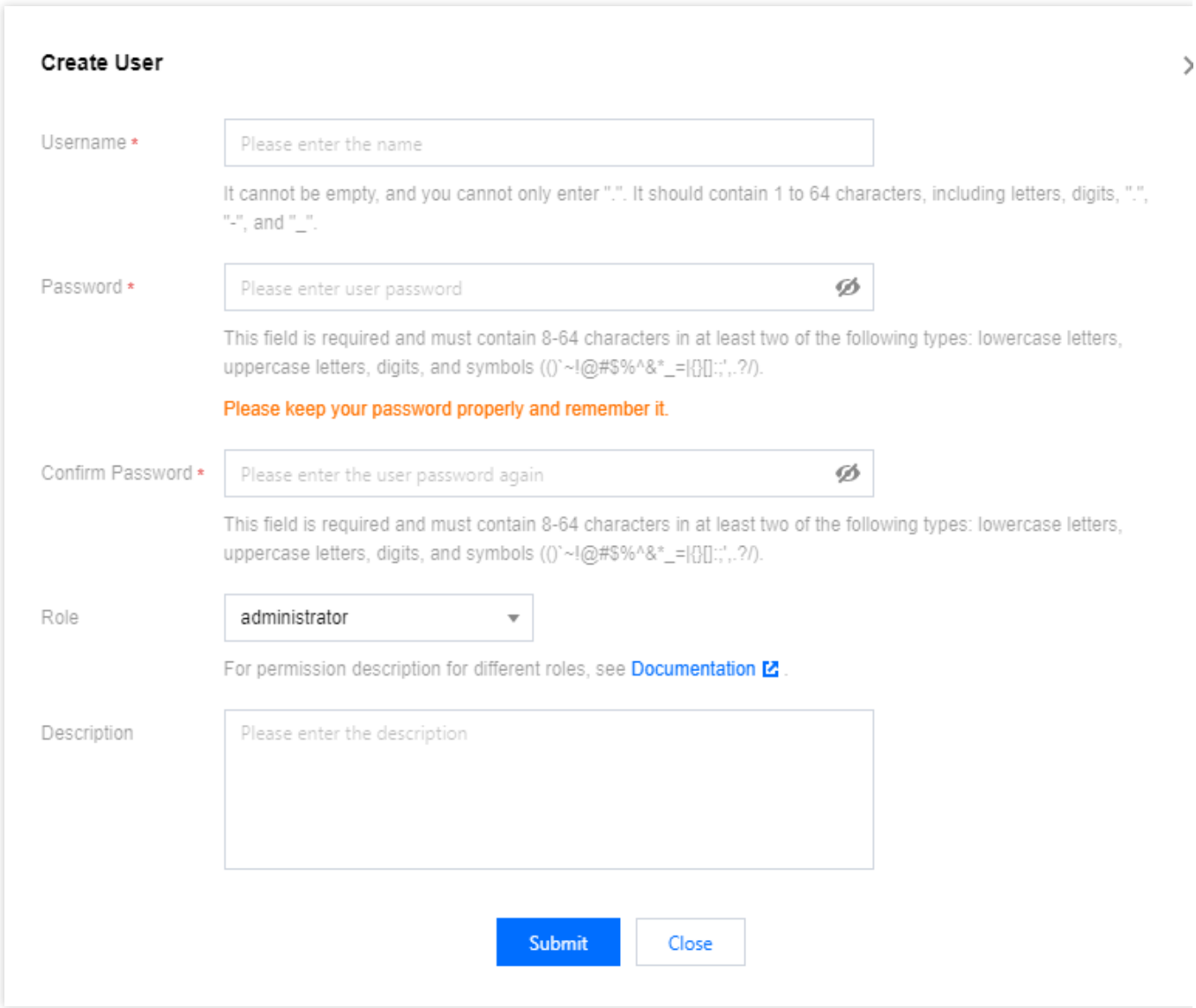
Adding a user

Every cluster has a user named "admin" by default. You can configure permissions for this default user or create new users as needed.

1. Log in to the RabbitMQ console.
2. Select **Cluster** > **Cluster** on the left sidebar, select a region, and click the ID of the target cluster to enter the **Basic Info** page of the cluster.
3. Select the **User and Permission** tab at the top of the page and click **Create User** on the **User Management** tab.
4. On the **Create User** page, enter the username, password, and description:
Username: Cannot be empty, should not only contain (.), must be 1-64 characters long, and can only include letters, digits, periods (.), hyphens (-), and underscores (__).
Password: Cannot be empty, must be 8-64 characters long, and must include at least two of the following: lowercase letters, uppercase letters, digits, special characters [()`~!@#$%^&*_=|{}[]:;',.?/] .
Role: Select the user role.

Role	Permission Description
none	Unable to log in to the Web console, typically applies to normal producers and consumers.
management	Web console; Can view Vhosts under their name, as well as queues, exchanges, and bindings within; Can view and close channels and connections under their name.
policymaker	On the basis of all Management rights: Can view, modify, and delete policies and parameters of Vhosts under their name.
monitoring	On the basis of all Management rights: Can view all Vhost, connection, and channel lists; Can view node-related information (such as disk usage, memory usage, number of processes.).
administrator	Super Administrator, on the basis of all rights in Policymaker and Monitoring: Can create and delete Vhost; Can view, create, and delete users and permissions; Close other users' connections.

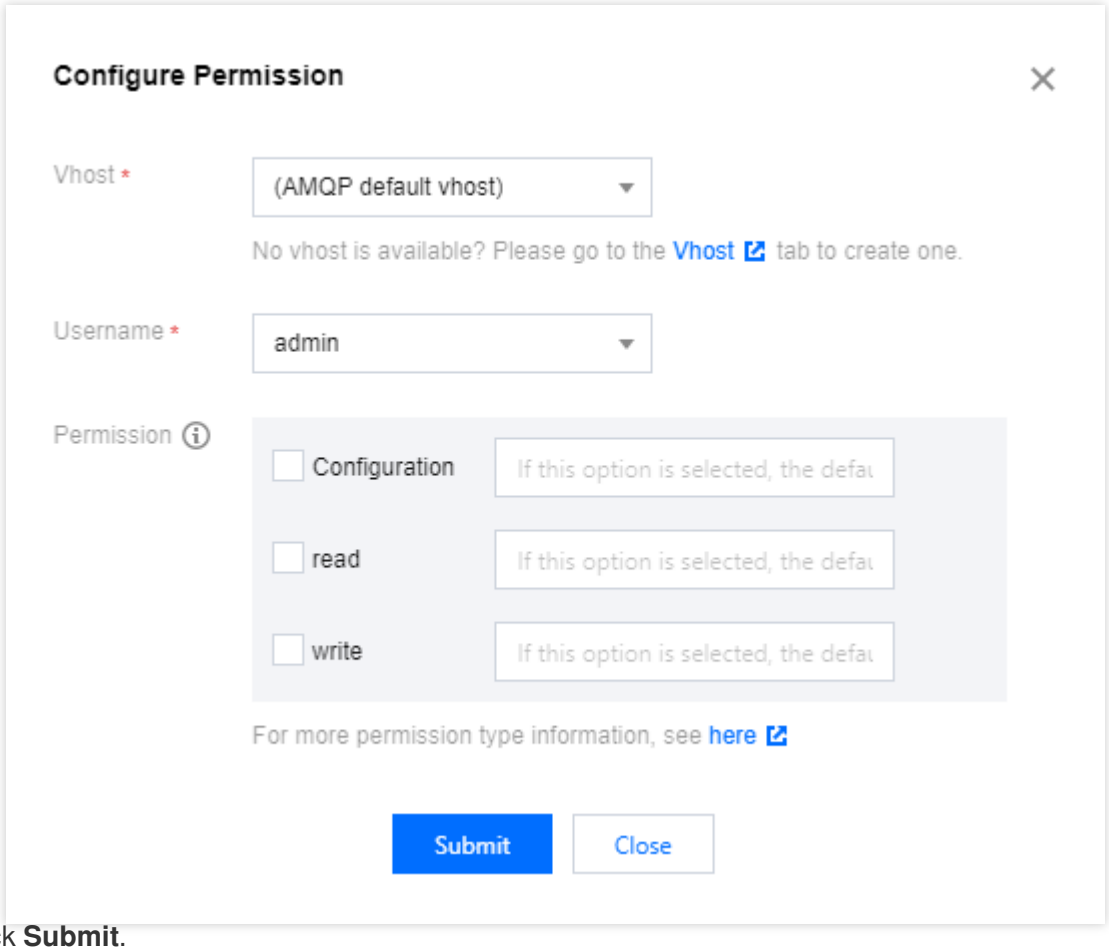
Description (optional): Enter a user description.
5. Click **Submit**.



Configuring a permission

1. On the **User and Permission** tab, select the **Permission List** tab and click **Configure Permission**.
2. On the **Configure Permission** page, select the target vhost and user and set permission rules.

Permission rules can match resources through **regex**. For example, if you select **Configuration** and enter "test.-*" in the input box, then the user will be granted the permission to configure all resources with a name starting with "test-" under the current vhost.



3. Click **Submit**.

4. Add the username and password to the client parameters. For directions on how to add the token parameters to the client code, see Spring Boot Starter (the parameters in this document are the username and password).

5. Check whether the permission is effective. You can run the configured client to access the exchange and queue resources in the vhost and produce/consume messages according to the configured permission. Check whether a no permission error is reported, and if not, the permission has been configured successfully.

Deleting a permission

Before deleting a permission, make sure that the current business no longer uses the user to produce/consume messages; otherwise, a client exception may occur due to the failure to produce/consume messages.

1. In the **User and Permission** tab, click **Delete** in the **Operation** column of the target user.
2. In the pop-up window, click **Delete**.

Policy List
Custom Policy

Last updated：2025-04-29 15:56:25

Overview

In RabbitMQ message queue, besides forced attributes such as durable and Exclusive, when creating a Queue or an Exchange, you can configure some optional attributes to obtain different features, such as x-message-ttl, x-expires, and x-max-length.
However, once the attribute parameters set for a Queue or an Exchange via RabbitMQ client are successfully set, they cannot be changed unless the original Queue or Exchange is deleted and a new one is created.
A policy is a special use of runtime parameters that supports dynamically modifying some attribute parameters. Policies are aimed at the Vhost level, and a policy can match one or more Queues or Exchanges, facilitating batch management. This solves the problem that Exchanges and Queues created by RabbitMQ clients cannot be modified, and greatly enhances the flexibility of the application.
This article introduces the directions to create a custom policy on the console.

Directions

Creating a Policy

When you create a cluster, if mirrored queue is enabled, there will be a default policy under the **Policy** tab in the console, which can be deleted, and you can also recreate or modify the policy.
1. Log in to the RabbitMQ Console.
2. In the left sidebar, choose **Vhost**, choose the region, then click the target Vhost's ID to enter the Basic Info page.
3. Click **Policy** > **Create Policy,** and fill in the basic information of the policy.

Basic Settings

Parameter	Description
Current Vhost	Indicates which Vhost is being created an image policy for
Policy Name	1 - 64 characters, only comprising of digits, letters, ".", "-" and "_"

Match Mode	A regular expression used to match related Queues or Exchanges. Commonly used match mode regular expressions can be found in: .*: Match all queues or exchanges under this Vhost. ^test.*: Matches all Queues or Exchanges whose names start with "test" under this Vhost. .*test.*: Matches all Queues or Exchanges whose names contain "test" under this Vhost. .*test$: Matches all Queues or Exchanges whose names end with "test" under this Vhost.
Policy type	Select a custom policy.
Application Scope	is used to specify the scope where the current Policy takes effect Exchanges And Queues: Act on all Queues or Exchanges that match the Pattern. Queues: Act on all Queues that match the Pattern. Exchanges: Act on all Exchanges that match the Pattern.
Priority	Define the priority of a policy, ranging from 0 to 255. If multiple policies act on the same Queue or Exchange, only the Policy with the highest priority number will be useful.

Strategy definition

You can select the necessary strategy definition fields yourself, as shown in the figure below.

The following are the fields supported by the current cloud console. For more fields, please go to the open-source console to create.

Field Classification	Field Name	Meaning	Data Type Restriction
Queues (All Types)	Max Length	**The maximum number of messages allowed in the queue.** When the number of messages in the queue reaches this limit, old messages will be deleted or new messages will be denied according to the Overflow Behaviour setting.	Number
	Max Length Bytes	**The maximum total message bytes allowed in the queue.** When the total message bytes in the queue reach this limit, old messages will be deleted or new messages will be denied according to the Overflow Behaviour setting.	Number
	Overflow Behaviour	**The handling method when the queue reaches the maximum length or maximum byte limit.** Valid values: drop-head: Delete old messages at the head of the queue reject-publish: Reject newly published messages	String

	Auto Expire	**The auto-expiration time of the queue.** The unit is milliseconds. When the queue is not accessed (for example, no messages are published, consumed, or the queue status is checked) within this time, the queue will be deleted.	Number
	Dead letter exchange	**Dead-letter exchange.** When a message is deleted due to exceeding TTL, reaching the queue maximum length, or being rejected by the consumer, these messages will be sent to the designated dead-letter exchange.	String
	Dead letter routing key	**Dead letter routing key.** This routing key can be used for routing when a message is sent to the dead letter exchange.	String
Queues [Classic]	Message TTL	**Message Time to Live (TTL).** The unit is milliseconds. This value defines the maximum duration a message can stay alive in the queue. When the message's survival time in the queue exceeds this value, the message will be deleted. If the message is consumed (and confirmed) by the consumer or republished to other queues, the timer will be reset.	Number
	Lazy mode	**Lazy mode.** After enabling this mode, RabbitMQ will store messages in the queue to disk as much as possible to reduce memory usage.	String ("lazy")
Queues [Quorum]	Max in memory length	**maximum unacknowledged messages allowed for each consumer before confirmation**	Number
	Max in memory bytes	**The maximum total number of bytes of memory allowed in the queue.** When the total number of bytes occupied by messages in the queue reaches this limit, RabbitMQ will attempt to write messages that exceed the limit to disk to reduce memory usage. Please note that this setting may be not applicable to queues with Lazy mode enabled, because in lazy mode, RabbitMQ will save messages to disk as much as possible by default.	Number
	Delivery limit	**maximum unacknowledged messages allowed for each consumer before confirmation**	Number
Exchanges	Alternate exchange	**Standby switch.** When a message cannot be routed to any queue (for example, there is no	String

		matching routing key or queue), the message will be sent to the designated standby switch.	

4. Click **Complete** to finish policy creation. You can see the created policy in the policy list.

Editing a Policy

1. In the policy list, click **Edit** in the target policy operation column.
2. In the pop-up window, edit the policy information.
3. Click **Complete** to complete the modification.

Deleting the Policy

1. In the policy list, find the policy that needs to be deleted and click **Delete** in the Action column.
2. In the pop-up dialog box, click **Delete** to complete the deletion.

Image Policy

Last updated：2025-04-29 15:30:06

Operation Background

To improve the reliability and fault tolerance of the RabbitMQ cluster, when users create a RabbitMQ cluster or a new Vhost (with at least three cluster nodes), we provide users with the option to enable "mirrored queues". This mirrored queue can replicate messages in the queue across multiple nodes in the RabbitMQ cluster, ensuring that messages in the queue will not be lost if a node fails.

Use Limits

TDMQ for RabbitMQ only allows clusters with **three or more** nodes to enable "mirrored queues", mainly to ensure high availability and fault tolerance of the cluster. In a cluster with three or more nodes, mirrored queues can replicate messages across multiple nodes. This way, it can share the load of each node, enhance performance, and ensure normal operation of the service when a node fails. At the same time, it also provides more flexibility, allowing us to flexibly configure the parameters of mirrored queues according to actual needs. Therefore, this limitation is to provide more stable and reliable services.

Enable Default Image Queue

By configuring the default mirrored queue policy, you can ensure RabbitMQ cluster reliability while optimizing performance and resource utilization. Users can further adjust these parameters according to their needs and scenarios, or delete and create new policies.
The following is the detailed parameter description of the "Default Image Queue" policy provided by tencent distributed message queue RabbitMQ for users:

Parameter Name	Configuration Parameters	Parameter Description
Name	pay-mirror-policy	Policy Name, used to identify and refer to this policy.
Pattern	.*	The match mode of the policy uses Regular Expression Syntax. Means to match any characters. Means to match the preceding character zero or multiple times. So .* means to match the name of any queue.

Apply to	Queues	Application object of the policy. Set to Queues indicates that the policy applies to queues.
Priority	0	Priority of the policy. If a queue matches multiple policies, the policy with higher priority will be applied. 0 represents the lowest priority.
ha-mode	exactly	Replication mode of mirrored queues. exactly: It means that the messages in the queue will be copied to a specified quantity of nodes. all: It means that the messages in the queue will be copied to all nodes. Nodes: It means performing mirroring on the specified nodes, and the node names are specified through mirroring parameters. Selecting "exactly" can reduce network and storage overhead while ensuring availability and improving performance.
ha-params	3	Copy parameters of the mirrored queue. When ha-mode is set to exactly, the number of replicated nodes needs to be set here. By default, it is set to 3. Even if it is expanded to 5 nodes in the future, the performance can still remain at a relatively good level.
ha-promote-on-failure	always	Mirror queue promotion strategy during node fault. Always means that regardless of the causes for node fault, the mirrored queue will be promoted to the main queue. When-synced means that the mirrored queue will be promoted to the main queue only when it is resynchronized after a node fault. The default setting is always to ensure service availability in any failure situation.
ha-promote-on-shutdown	when-synced	Mirror queue promotion strategy during node normal shutdown. - always: It means that the mirrored queue will be promoted to the main queue regardless of the reasons for node shutdown. - when-synced: It indicates that the mirrored queue will be promoted to the main queue only after the node is closed and resynchronized. The default setting is when-synced to avoid unnecessary escalation operations.
ha-sync-mode	manual	Synchronization mode of mirrored queues. - Automatic: It means that the mirrored queue is automatically synchronized with the main queue when the node starts up or reconnects to the cluster. - Manual: It indicates a requirement to manually trigger the synchronization operation to synchronize the mirrored queue with the main queue. The default setting is manual to avoid affecting cluster performance during automatic synchronization when messages accumulate.

Operation Steps

Method one: Enable mirror queue when creating a new cluster: Cluster Purchase Page > **Other Configuration** > **Enable Mirror Queue.** This operation only takes effect on the cluster default Vhost.

Method 2: Enable mirror queue when creating a new Vhost: TDMQ for RabbitMQ Console > **Cluster Management** > **Vhost** > **Creation** > **Enable Mirror Queue.**

Create an Image Policy

When creating a new cluster, if the mirrored queue is enabled, there will be a default policy under the **Policy** tab on the console. It can be deleted. You can also create or modify a policy again.

Operation Steps

1. Log in to the TDMQ for RabbitMQ console.
2. Select **Cluster Management** > **Vhost** in the left sidebar. After selecting a region, click the target Vhost's ID to enter the basic information page.
3. Click **Policy** > **Create Policy**, and fill in the basic information of the policy.

Basic Settings:

Parameter	Description
Current Vhost	Indicates which Vhost is being created with an image policy.
Policy Name	1 - 64 characters, only digits, letters, ".", "-" and "_" are allowed.
Match Mode	A regular expression used to match related Queues or Exchanges. For commonly used match mode regular expressions, refer to: .*: Match all queues or exchanges under this Vhost. ^test.*: Matches all Queues or Exchanges whose names start with "test" under this Vhost. .*test.*: Matches all Queues or Exchanges whose names contain "test" under this Vhost. .*test$: Matches all Queues or Exchanges whose names end with "test" under this Vhost.
Policy type	Select **Image Policy**.
Application Scope	Is used to specify the range where the current Policy takes effect. The image policy is only supported to take effect in the Queue.
Priority	Define the priority of a policy. Optional range: [0, 255]. If multiple policies are applied to the same Queue, the Policy with the highest priority number will be effective.

Policy Definition

Parameter	Description
Mirror mode	Mirror queue mode. Valid values are all, exactly, nodes. all: Means performing mirroring on all nodes in a cluster. exactly: Refers to performing mirroring on a specified number of nodes, and the number of nodes is specified by **mirror parameters**. Nodes: It means performing mirroring on the specified nodes, and the node names are specified through mirroring parameters.
Mirror parameters	Mirror parameters: It means the nodes to which the message will be synchronized. When the mirror mode is set to all, this item can be left blank. When the mirror mode is set to exactly, it is recommended to select 3 for the mirror parameters. The maximum value can be equal to the current number of cluster nodes, and the minimum can be 1. When the mirror mode is set to nodes, the mirror parameters can be selected according to node names to specify which specific nodes. It is recommended to select 3 nodes.
message synchronization method	Message synchronization method in the mirrored queue, optional automatic or manual.
Primary node exit handling	Whether to allow an unsynced mirror to be elected as the primary node when the primary node quits gracefully.
Primary node fault handling	When a primary node fails, whether to allow the election of an unsynced mirror as the master. To ensure availability, it is recommended to keep "allow selecting all mirrors".

4. Click **Complete** to complete policy creation. You can see the created policy in the policy list.

Smart Inspection

Last updated：2024-06-26 15:56:25

Overview

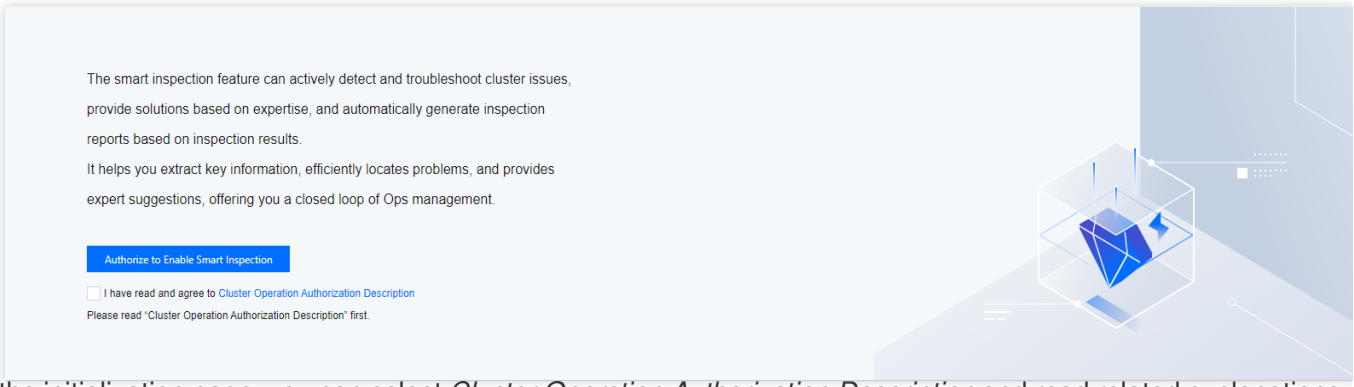
As monitoring metrics increase, there are higher requirements on understanding Ops metrics. TDMQ for RabbitMQ provides smart inspection, which can actively troubleshoot cluster issues and potential risks, provide solution based on expert experience, and automatically summarize health check results into reports.
Smart inspection helps you extract key information, efficiently locate issues, offer professional suggestions, and close Ops experience issues.

Enabling Smart Inspection

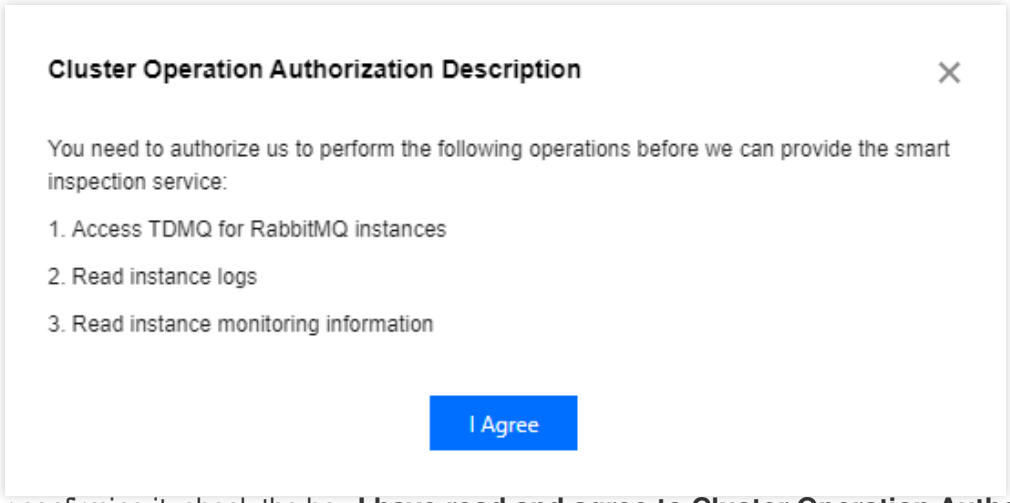
Smart inspection needs to access basic information, logs, and monitoring data of your cluster. Therefore, you need to authorize access when the smart inspection service is enabled for the first time.

Directions

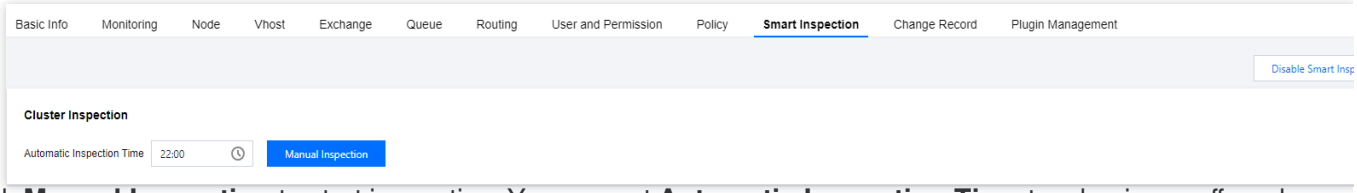
1. Log in to the RabbitMQ console.
2. In the left sidebar, select **Cluster** > **Cluster**, select a region, and click the ID of the target cluster to enter the cluster detail page.
3. At the top of the cluster detail page, select the **Smart Inspection** tab to enter the smart inspection page.



4. On the initialization page, you can select *Cluster Operation Authorization Description* and read related explanations and prompts.



5. After confirming it, check the box **I have read and agree to Cluster Operation Authorization Description**. Then, click **Authorize to Enable Smart Inspection** to enable smart inspection.



6. Click **Manual Inspection** to start inspection. You can set **Automatic Inspection Time** to a business off-peak period. In this case, TDMQ for RabbitMQ conducts smart inspection at the designated time every day.

Viewing Inspection Results

The inspection results summarize and display the inspection results and change trends of the cluster for you to view the recent health status of the cluster. The health status of a cluster includes high risk, low risk, and secure.
High risk: The cluster has encountered severe issues or hidden dangers, which affect cluster availability and require immediate attention to prevent data loss, cluster failure, or other issues.
Low risk: The cluster has significant issues or hidden dangers that might affect cluster availability. It is recommended that you address these issues as soon as possible.
Secure: The cluster is healthy.

Directions

1. Log in to the RabbitMQ console.
2. In the left sidebar, choose **RabbitMQ** > **Cluster**, select a region, and click the ID of the target cluster to enter the Cluster detail page.
3. At the top of the Cluster detail page, select the **Smart Inspection** tab to enter the smart inspection page.
4. On the smart inspection result page, the latest inspection results and result distribution of the cluster are displayed.

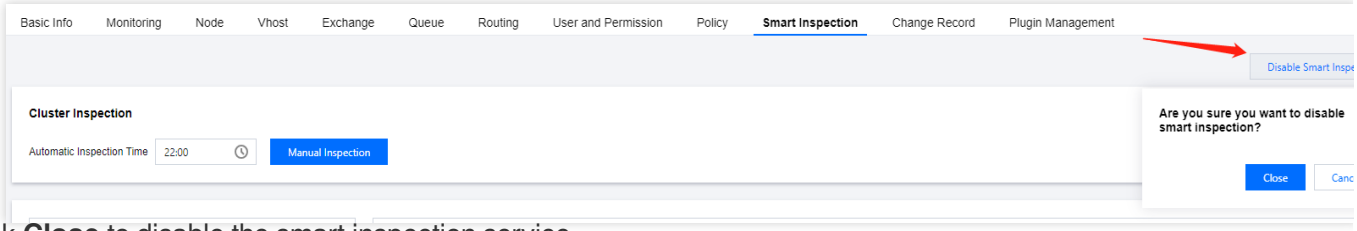
5. Click **View Inspection Report** to view the detailed inspection report, including the inspection time, resource ID, inspection conclusion, inspection items, and detailed inspection results.
6. Click the **Download** icon in the upper left corner of the report to download the inspection report.
7. Click the directory on the right to view and download inspection reports in the lastest 30 days.

Disabling Smart Inspection

When you no longer need the smart inspection feature, you can disable the service. After it is disabled, the system no longer performs scheduled cluster inspection or generates new inspection reports.

Directions

1. Log in to the RabbitMQ console.
2. In the left sidebar, select **RabbitMQ** > **Cluster**, select a region, and click the ID of the target cluster to enter the cluster detail page.
3. At the top of the cluster detail page, select the **Smart Inspection** tab to enter the smart inspection page, and click **Disable Smart Inspection**.



4. Click **Close** to disable the smart inspection service.

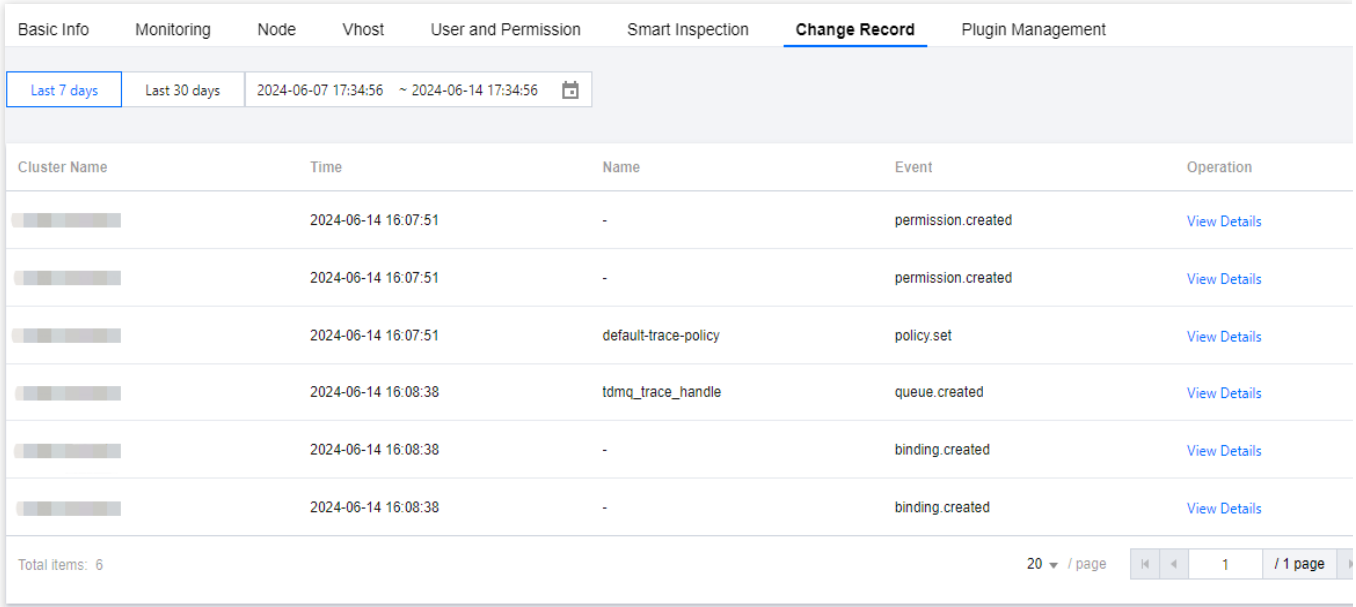
Change Records

Last updated：2024-06-26 16:06:00

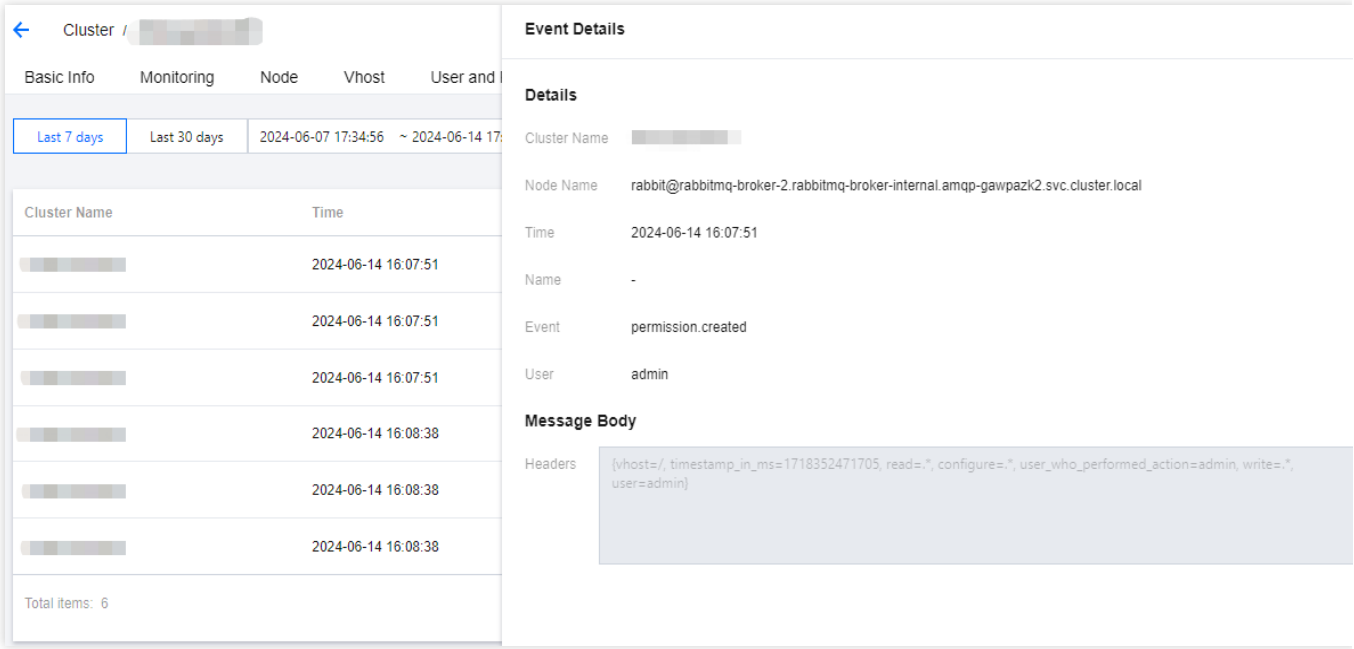
Change records in TDMQ for RabbitMQ manage, store, analyze, and display the change event data uniformly, making it easier for you to view and analyze. You can view details within the Change Record module.
This document explains how to view the details of change records on the TDMQ for RabbitMQ console.

Viewing Change Records

1. Log in to the RabbitMQ Console.
2. In the left sidebar, choose **cluster** , choose the appropriate region, then click the ID of the cluster you want to view, and enter the Cluster Details page.
3. On the top of the Cluster Details page, select the **Change Record** tab to enter the Change Record page.
4. Set the time period (supports Last 7 Days, Last 30 Days, and Custom Time Range), and you can view change events within the corresponding time period.



5. Click the **View Details** in the Operation Column, and you can view the event details on the right sidebar.


Cluster /
Basic Info
Monitoring
Node
Vhost
User and
Last 7 days
Last 30 days
2024-06-07 17:34:56 ~ 2024-06-14 17:
Cluster Name
Time
2024-06-14 16:07:51
2024-06-14 16:07:51
2024-06-14 16:07:51
2024-06-14 16:08:38
2024-06-14 16:08:38
2024-06-14 16:08:38
Total items: 6
Event Details
Details
Cluster Name
Node Name
rabbit@rabbitmq-broker-2.rabbitmq-broker-internal.amqp-gawpazk2.svc.cluster.local
Time
2024-06-14 16:07:51
Name
-
Event
permission.created
User
admin
Message Body
Headers
{vhost=/, timestamp_in_ms=1718352471705, read=.*, configure=.*, user_who_performed_action=admin, write=.*, user=admin}

Monitoring alarm
Viewing Monitoring Metric

Last updated：2024-10-18 17:29:39

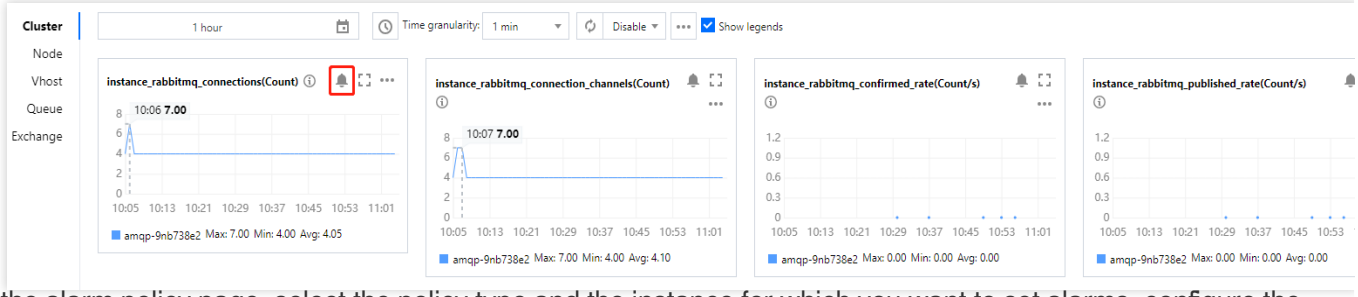
Overview

TCOP products provide monitoring features to all users by default, with no need for manual activation. The platform starts collecting monitoring data only after the user starts using a specific Tencent Cloud product.
TDMQ for RabbitMQ supports monitoring the resources you create under your account, including clusters, nodes, Vhosts, queues, and exchanges, helping you stay informed about resource status in real-time. You can configure alarm rules for monitoring metrics, and when a metric reaches the specified threshold, TCOP can notify you via email, SMS, WeChat, or phone, enabling you to respond to exceptions promptly.

Configuring Alarm Rules

Creating Alarm Rules

You can configure alarm rules for monitoring metrics, and when a metric reaches the specified threshold, TCOP can notify you via email, SMS, WeChat, or phone, helping you promptly address any abnormal situations.
1. On the monitoring page of the cluster, click the alarm button shown below to jump to the TCOP console to configure alarm policies.



2. On the alarm policy page, select the policy type and the instance for which you want to set alarms, configure the alarm rules, and set the notification template.
Monitoring Type: Select **Cloud Product Monitoring**.
Policy Type: Select **TDMQ** / **RabbitMQ Exclusive Edition**.
Alarm Object: Select the RabbitMQ resource for which you want to configure the alarm policy.
Trigger Condition: Support **Select template** and **Manual configuration**. By default, manual configuration is selected. For manual configuration instructions, see the following guidelines. To create a template, see Create trigger

condition template.

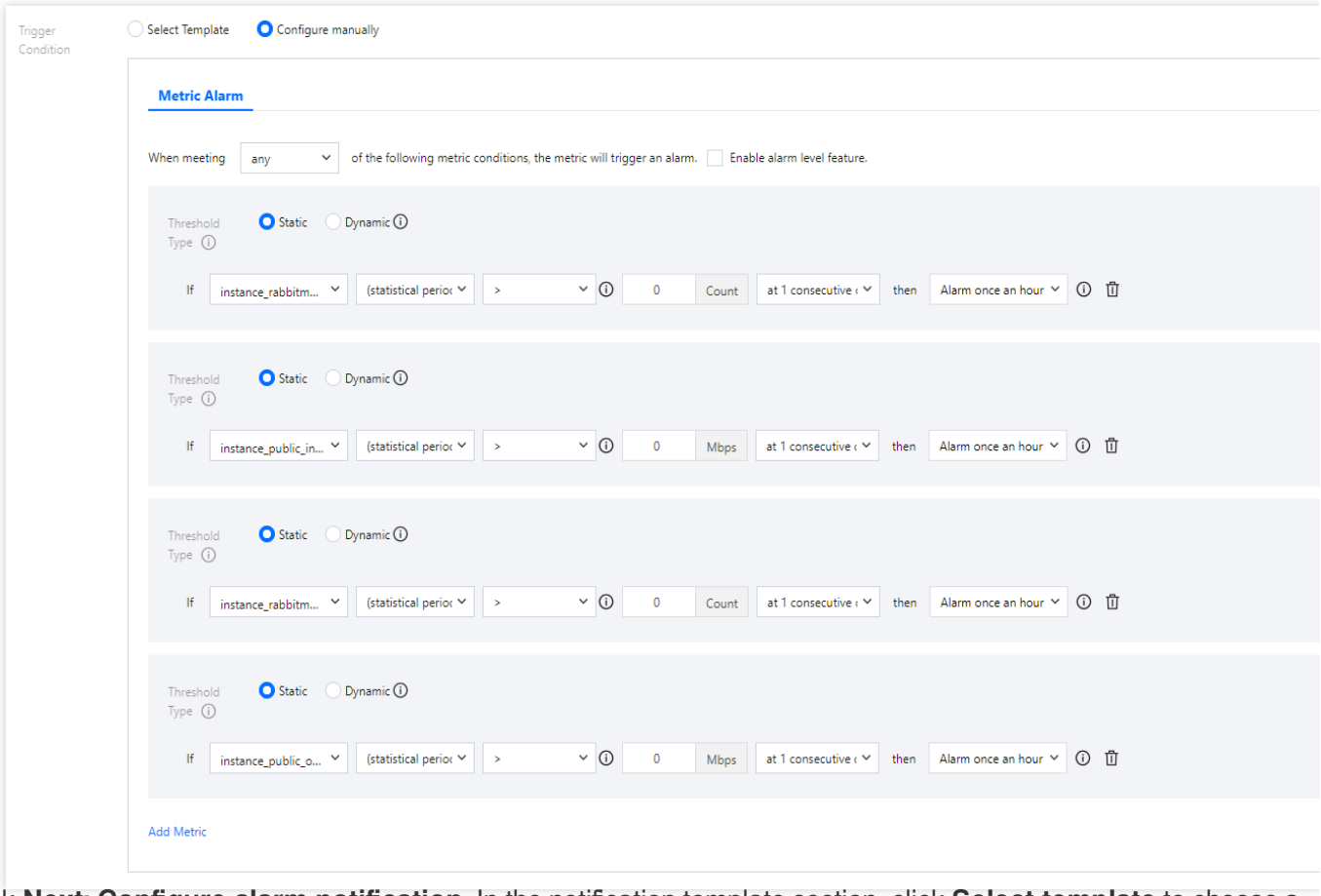
You can directly use the pre-configured alarm templates for TDMQ for RabbitMQ. Follow these steps:

Step 1: For **Policy Type**, select **TDMQ** / **RabbitMQ Exclusive Edition** / **Cloud Data Disks**.

Step 2: For **Alert Object**, select the RabbitMQ resource for which you want to configure the alarm policy.

Step 3: For **Trigger Conditions**, check **Use predefined trigger conditions**. The predefined alearm trigger conditions will appear as shown below:

Step 4: Adjust the specific trigger conditions according to your business needs.



3. Click **Next: Configure alarm notification**. In the notification template section, click **Select template** to choose a template. You can also click **Create new template** to create a notification template, setting the alarm recipients and notification channels.

Note:

For more information on alarms, see TCOP Alarm Policies.

4. Click **Complete** to finish the configuration.

Creating Trigger Condition Template

1. Log in to the TCOP console.

2. In the left sidebar, click **Alarm Configuration** to enter the alarm configuration list page , and then click Trigger Condition Template.

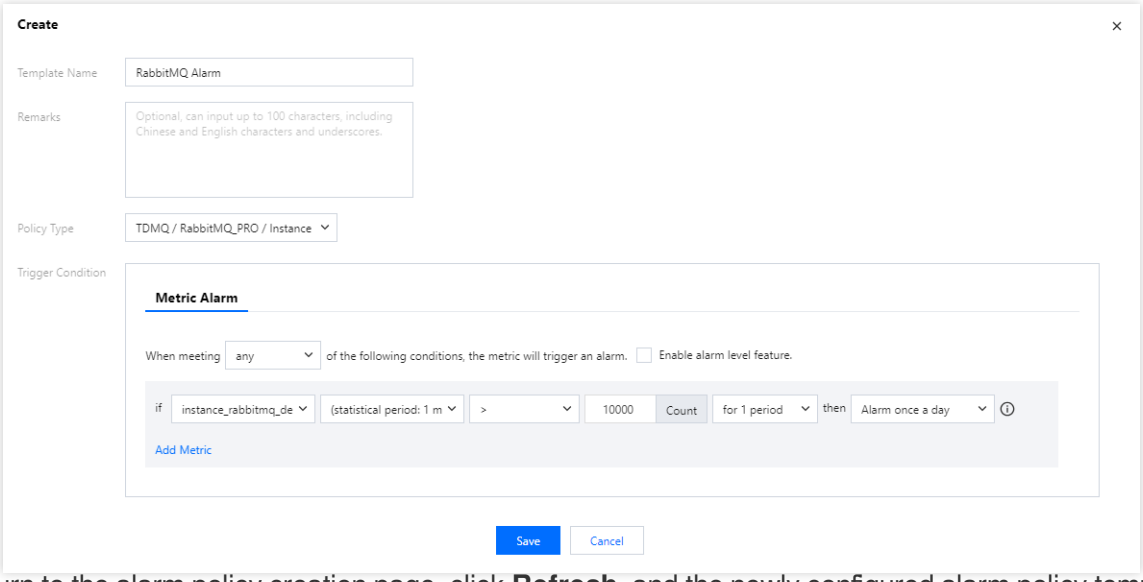
3. In the Trigger Condition Template tab, click **Create Trigger Condition Template**.

4. On the template creation page, configure the policy type.

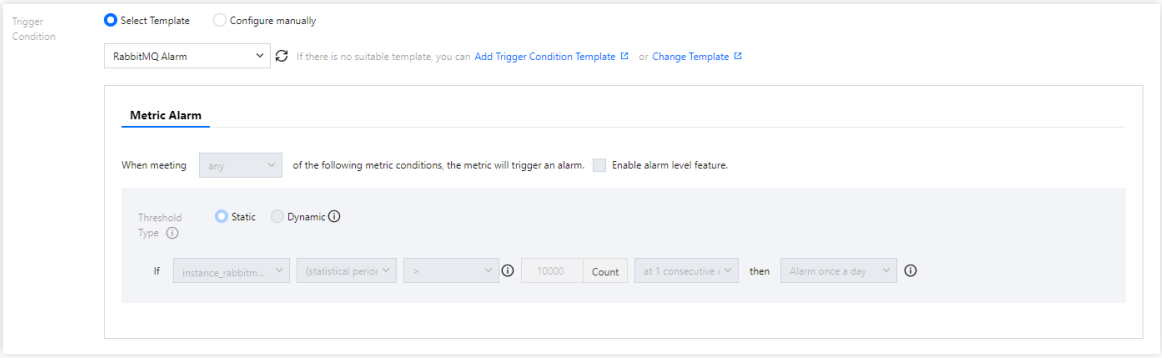
Policy Type: Select the policy type under **TDMQ** / **RabbitMQ Exclusive Edition** category.

Use Preset Trigger Conditions: Check this option to display the system-recommended alarm policies.

5. After confirmation, click **Save**.



6. Return to the alarm policy creation page, click **Refresh**, and the newly configured alarm policy template will appear.



Alarm Configuration Recommendations

This section introduces key metrics you should focus on during the use of TDMQ for RabbitMQ and provides recommendations for configuring alarms accordingly:

Metric	Dimension	Recommended Alarm Configuration	Detailed Description
Disk utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the disk utilization exceeds 80% for 5 consecutive	High disk utilization can result in insufficient disk space on the node to store the messages assigned to it, preventing messages from being written to disk. It is recommended to clear data

		data points, and raise the alarm once every 30 minutes.	or scale out the cluster when the average disk utilization exceeds 80%.
Memory utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the memory utilization exceeds 50% for 5 consecutive data points, and raise the alarm once every 30 minutes.	High memory utilization can block message production. It is recommended to speed up consumption, apply flow control to production, or scale out the cluster when memory utilization exceeds 50%.
CPU utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the CPU utilization exceeds 70% for 5 consecutive data points, and raise the alarm once every 30 minutes.	High CPU utilization can affect the message production speed. It is recommended to scale out promptly when CPU utilization exceeds 70%.
Number of heaped messages	Node	Set a calculation granularity of 5 minutes, trigger the alarm when the number of heaped messages exceeds the expected business threshold for 5 consecutive data points, and raise the alarm once every 30 minutes.	Too many heaped messages will cause the disk usage of the Broker node to rise rapidly, preventing it from receiving more messages. Scale-out is required.
Node availability (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the node availability value equals to 1 for 3 consecutive data points, and raise the alarm once every 15 minutes.	Node availability issues or downtime can lead to message loss, especially when persistence or image queues are not enabled. This will also increase the load on other nodes, potentially reducing overall cluster performance. It is recommended to check the cause in combination with other metrics and alarm information.

Note:

Metric: For example, number of connections. If the calculation granularity is set to 1 minute, an alarm will be triggered if the average production latency exceeds the threshold for N consecutive data points within 1 minute.

Alarm frequency: For example, alarm once every 30 minutes means that within a 30-minute period, even if the metric exceeds the threshold during multiple consecutive calculation cycles, only one alarm will be triggered. No further alarms will be triggered during that 30-minute window. If the metric continues to exceed the threshold after the 30-minute period, a new alarm will be triggered.

Configuring Alarms

Last updated：2024-10-18 17:31:02

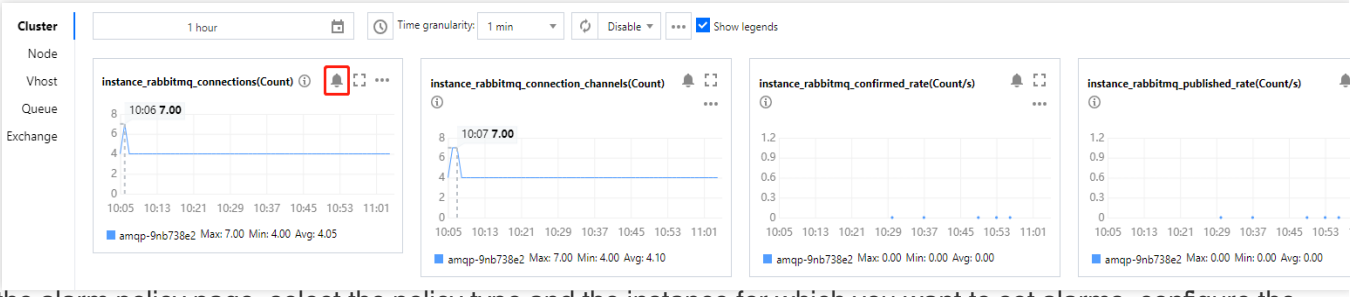
Overview

TCOP products provide monitoring features to all users by default, with no need for manual activation. The platform starts collecting monitoring data only after the user starts using a specific Tencent Cloud product.
TDMQ for RabbitMQ supports monitoring the resources you create under your account, including clusters, nodes, Vhosts, queues, and exchanges, helping you stay informed about resource status in real-time. You can configure alarm rules for monitoring metrics, and when a metric reaches the specified threshold, TCOP can notify you via email, SMS, WeChat, or phone, enabling you to respond to exceptions promptly.

Configuring Alarm Rules

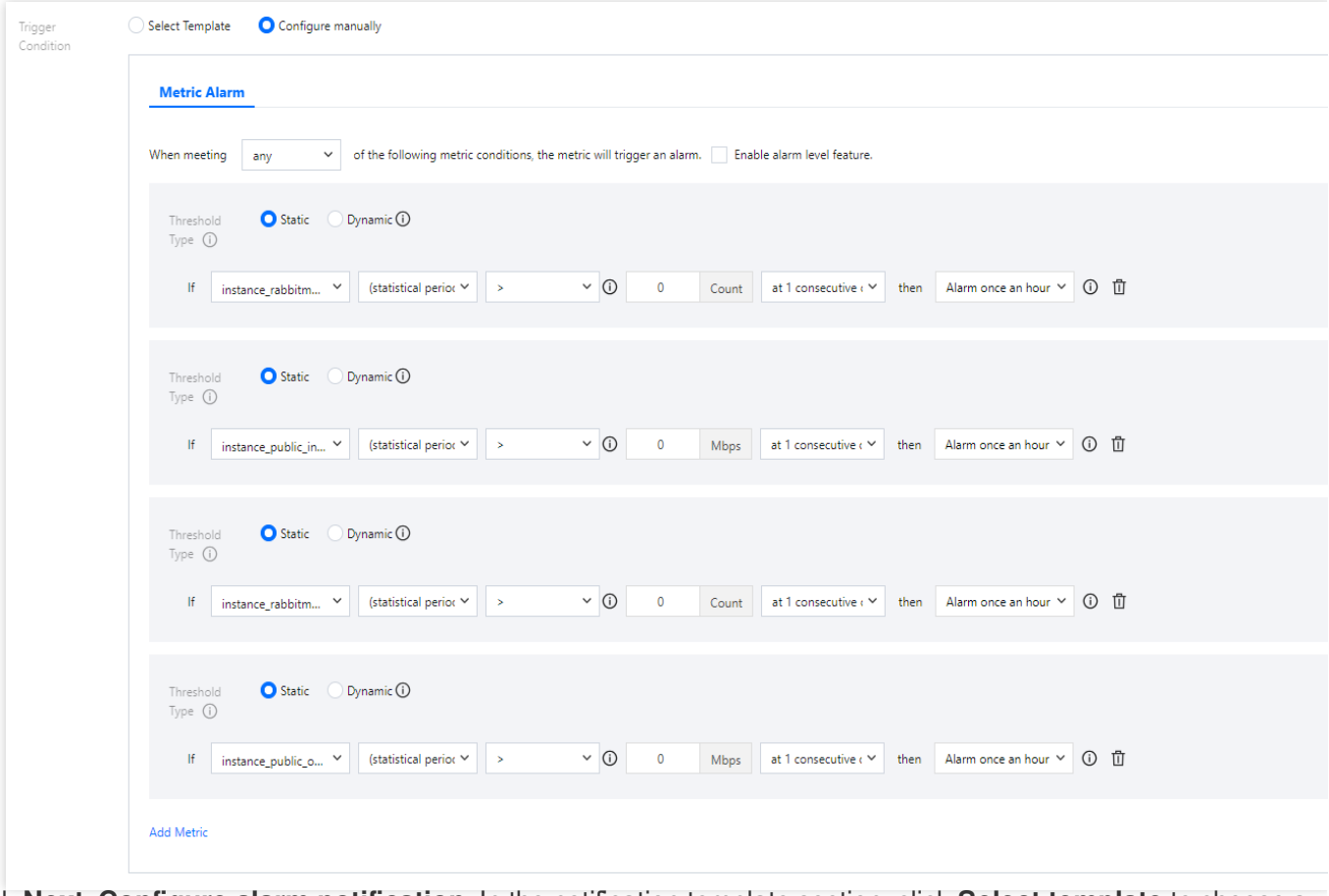
Creating Alarm Rules

You can configure alarm rules for monitoring metrics, and when a metric reaches the specified threshold, TCOP can notify you via email, SMS, WeChat, or phone, helping you promptly address any abnormal situations.
1. On the monitoring page of the cluster, click the alarm button shown below to jump to the TCOP console to configure alarm policies.



2. On the alarm policy page, select the policy type and the instance for which you want to set alarms, configure the alarm rules, and set the notification template.
Monitoring Type: Select **Cloud Product Monitoring**.
Policy Type: Select **TDMQ** / **RabbitMQ Exclusive Edition**.
Alarm Object: Select the RabbitMQ resource for which you want to configure the alarm policy.
Trigger Condition: Support **Select template** and **Manual configuration**. By default, manual configuration is selected. For manual configuration instructions, see the following guidelines. To create a template, see Create trigger condition template.
You can directly use the pre-configured alarm templates for TDMQ for RabbitMQ. Follow these steps:

Step 1: For **Policy Type**, select **TDMQ** / **RabbitMQ Exclusive Edition** / **Cloud Data Disks**.
Step 2: For **Alert Object**, select the RabbitMQ resource for which you want to configure the alarm policy.
Step 3: For **Trigger Conditions**, check **Use predefined trigger conditions**. The predefined alearm trigger conditions will appear as shown below:
Step 4: Adjust the specific trigger conditions according to your business needs.



3. Click **Next: Configure alarm notification**. In the notification template section, click **Select template** to choose a template. You can also click **Create new template** to create a notification template, setting the alarm recipients and notification channels.

Note:

For more information on alarms, see TCOP Alarm Policies.

4. Click **Complete** to finish the configuration.

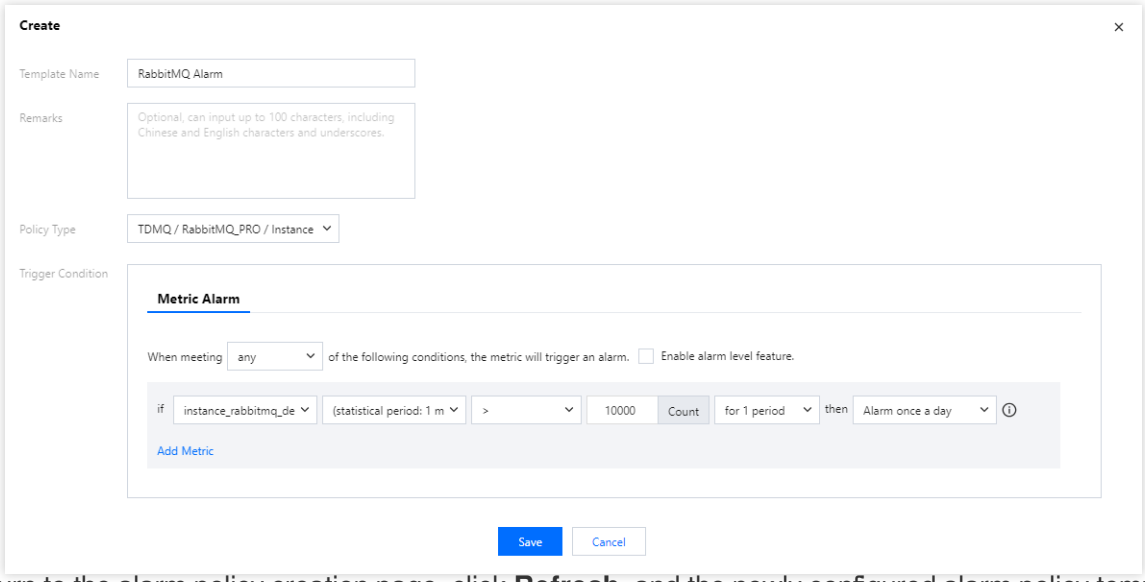
Creating Trigger Condition Template

1. Log in to the TCOP console.
2. In the left sidebar, click **Alarm Configuration** to enter the alarm configuration list page , and then click Trigger Condition Template.
3. In the Trigger Condition Template tab, click **Create Trigger Condition Template**.
4. On the template creation page, configure the policy type.

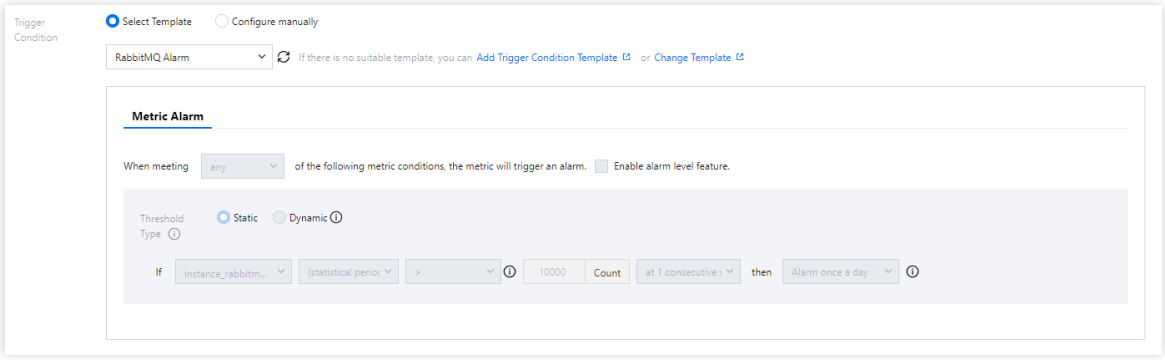
Policy Type: Select the policy type under **TDMQ** / **RabbitMQ Exclusive Edition** category.

Use Preset Trigger Conditions: Check this option to display the system-recommended alarm policies.

5. After confirmation, click **Save**.



6. Return to the alarm policy creation page, click **Refresh**, and the newly configured alarm policy template will appear.



Alarm Configuration Recommendations

This section introduces key metrics you should focus on during the use of TDMQ for RabbitMQ and provides recommendations for configuring alarms accordingly:

Metric	Dimension	Recommended Alarm Configuration	Detailed Description
Disk utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the disk utilization exceeds 80% for 5 consecutive data points, and raise the alarm once every 30 minutes.	High disk utilization can result in insufficient disk space on the node to store the messages assigned to it, preventing messages from being written to disk. It is recommended to clear data or scale out the cluster when the average disk utilization exceeds 80%.

Memory utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the memory utilization exceeds 50% for 5 consecutive data points, and raise the alarm once every 30 minutes.	High memory utilization can block message production. It is recommended to speed up consumption, apply flow control to production, or scale out the cluster when memory utilization exceeds 50%.
CPU utilization (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the CPU utilization exceeds 70% for 5 consecutive data points, and raise the alarm once every 30 minutes.	High CPU utilization can affect the message production speed. It is recommended to scale out promptly when CPU utilization exceeds 70%.
Number of heaped messages	Node	Set a calculation granularity of 5 minutes, trigger the alarm when the number of heaped messages exceeds the expected business threshold for 5 consecutive data points, and raise the alarm once every 30 minutes.	Too many heaped messages will cause the disk usage of the Broker node to rise rapidly, preventing it from receiving more messages. Scale-out is required.
Node availability (%)	Node	Set a calculation granularity of 1 minute, trigger the alarm when the node availability value equals to 1 for 3 consecutive data points, and raise the alarm once every 15 minutes.	Node availability issues or downtime can lead to message loss, especially when persistence or image queues are not enabled. This will also increase the load on other nodes, potentially reducing overall cluster performance. It is recommended to check the cause in combination with other metrics and alarm information.

Note:

Metric: For example, number of connections. If the calculation granularity is set to 1 minute, an alarm will be triggered if the average production latency exceeds the threshold for N consecutive data points within 1 minute.

Alarm frequency: For example, alarm once every 30 minutes means that within a 30-minute period, even if the metric exceeds the threshold during multiple consecutive calculation cycles, only one alarm will be triggered. No further alarms will be triggered during that 30-minute window. If the metric continues to exceed the threshold after the 30-minute period, a new alarm will be triggered.

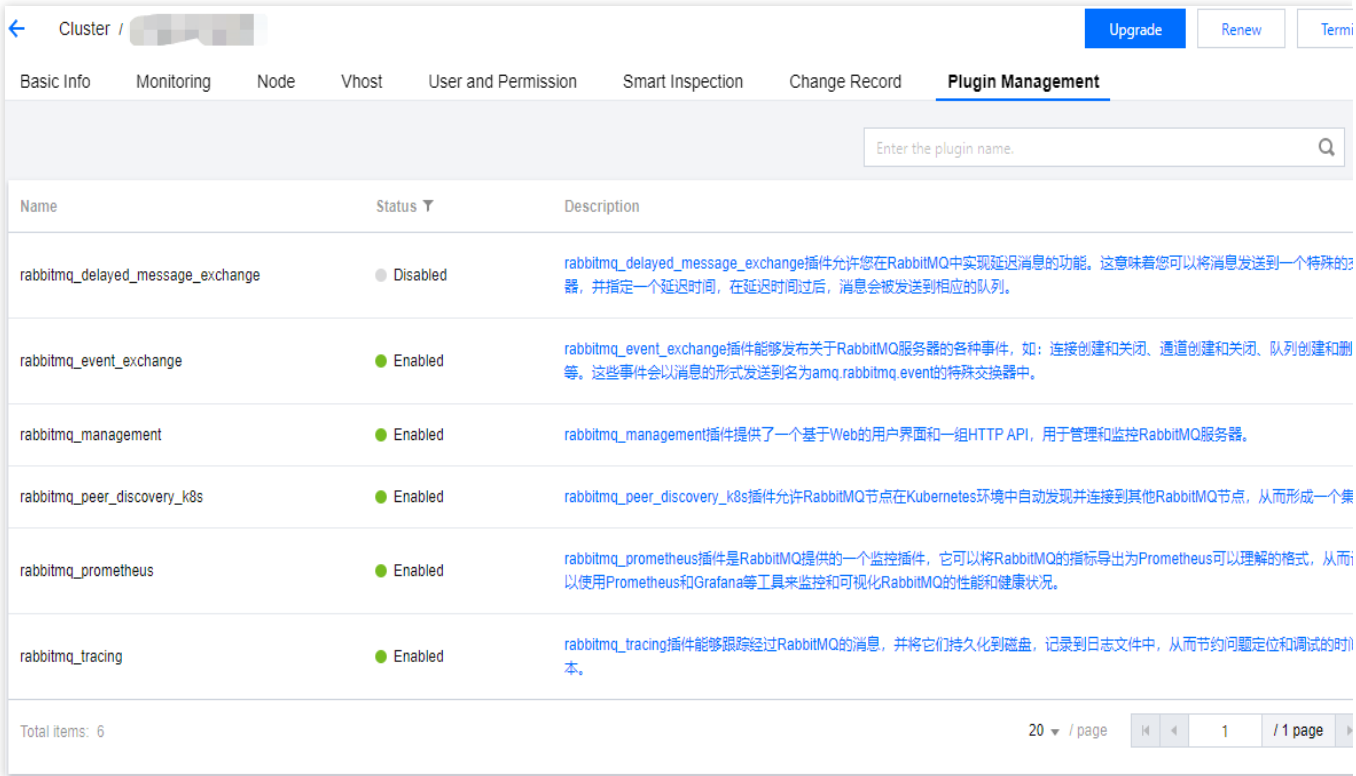
Plugin Management

Last updated：2024-06-26 15:56:25

The **Plugin Management** module allows you to view and manage plugins supported by a TDMQ for RabbitMQ cluster.
This document describes how to view supported plugins in the TDMQ for RabbitMQ console.

Viewing Plug-ins

1. Log in to the RabbitMQ console.
2. In the left sidebar, select **Cluster**, select an appropriate region, and click the ID of the target cluster to enter the cluster detail page.
3. At the top of the cluster detail page, select the **Plugin Management** tab to enter the plugin management page.



The following table describes plugins that are enabled by default after a TDMQ for RabbitMQ cluster is created.

Plugin	Description
rabbitmq_event_exchange	The rabbitmq_event_exchange plugin can publish various events about the RabbitMQ server, such as connection creation and closure, channel creation and closure, and queue creation and deletion. These events are sent to a special exchange named **amq.rabbitmq.event** using messages.

rabbitmq_management	The rabbitmq_management plugin provides web user interfaces and a set of HTTP APIs for managing and monitoring the RabbitMQ server.
rabbitmq_peer_discovery_k8s	The rabbitmq_peer_discovery_k8s plugin allows RabbitMQ nodes to automatically discover and connect to other RabbitMQ nodes in a Kubernetes environment to form a cluster.
rabbitmq_prometheus	The rabbitmq_prometheus plugin is a monitoring plugin provided by RabbitMQ. It can export RabbitMQ metrics in a format understandable by Prometheus. Therefore, you can use tools, such as Prometheus and Grafana to monitor and visualize the RabbitMQ performance and health status.
rabbitmq_tracing	The rabbitmq_tracing plugin can trace messages passing through RabbitMQ and persist them to a disk or record them in log files, thereby reducing time in problem identification and debugging.

Note:

TDMQ for RabbitMQ has the rabbitmq_delayed_message_exchange delayed message plugin **Disabled by Default** , due to the following risks and limitations of the plugin:

1. The current plugin's design is not suitable for scenarios with a high volume of delayed messages (unscheduled messages reaching hundreds of thousands or even millions). Please carefully evaluate the message throughput in a production environment to avoid unexpected long delay, message loss, and other issues.
2. Delayed messages only have one persistent copy on each node. If a node cannot function properly (for example, heaped messages cause continuous OOM leading to restarts and inability to recover), the delayed messages on that node cannot be consumed by the consumer side.
3. The delayed switch does not support setting **mandatory** . Producers cannot be notified about messages that could not be routed through **basic.return** events. Thus, before they send delayed messages, it is crucial to ensure that the corresponding switches, queues, and routing relationships exist.

For the original text on risks and limitations, please see the Official Usage Restriction Statement of the RabbitMQ Delayed Message Plugin.

Additionally, you can see the documentation TDMQ for RabbitMQ > Development Guide > Delayed Messages for two implementations of delayed messages described therein.

Message Query

Last updated：2025-04-29 10:56:29

Overview

If there are abnormal or missing messages, you can use the message query feature in the TDMQ for RabbitMQ console to analyze and locate the problem promptly.
This document describes how to query messages in the TDMQ for RabbitMQ console.

Prerequisites

You have toggled on **Trace Plugin** for the vhost where you want to query messages.

Implementation Principle Overview

After enabling the Trace plugin for the Vhost, the service component will consume the corresponding RabbitMQ cluster's track messages. After processing, it can achieve the feature of querying message trajectories in the console.

Use Limits

Overview of the implementation principle of message query: After the Trace Plugin of the VHost is enabled, the service component will consume the trace messages of the corresponding RabbitMQ cluster. Through a series of processing, it can achieve the feature of querying message traces on the console.
Based on the principles mentioned, message trace relies on service components consuming trace messages. Since service components are underlying public services, they cannot guarantee that trace messages from high-traffic RabbitMQ clusters can be consumed in a timely manner; heap of trace messages can cause issues such as high memory load within the cluster, affecting the stability of the RabbitMQ cluster.
Therefore, it is not recommended to enable the Trace Plugin in production environments, especially in overall clusters (including all VHosts), where the scenario involves sending **TPS over 10,000** . The Trace Plugin is advised to be used in small-traffic verification/troubleshooting scenarios.

Directions

1. Log in to the RabbitMQ console.
2. On the left sidebar, click **Message Query** and select the region and time range for the query.
3. Select the cluster, vhost, and queue to be queried. You can enter the routing key, user, message headers, and message body to narrow down the query.
4. Click **Query**, and all results will be displayed in the list below.

Note：
To ensure the stability of the cluster, the console restricts the quantity and dimensions of message queries. Users can query messages under specific queues, with up to 10,000 results returned. These results are based on filtering by queue and Routing Key (adding a Routing Key filter is optional). Users can also add filters for user, headers, and body to further refine their search, but the query will be conducted within the aforesaid limit of up to 10,000 messages. Therefore, it is recommended to use the message query feature in small-traffic verification/troubleshooting scenarios.
5. Click **View Message Details** in the **Operation** column of the target message to view its details and content (message body).

Access Management
Granting Sub-Account Access Privileges

Last updated：2025-04-29 14:50:44

Basic Concepts of CAM

The root account authorizes sub-accounts by associating policies. The policy setting can be specific to the level of **[API, Resource, User/User Group, Allow/Deny, and Condition]**.

Account

Root account: Possess all resources on Tencent Cloud and can have any access to any of its resources.
Sub-account: Including sub-users and collaborators.
Sub-user: It is created and fully owned by a root account that created the sub-user.
Collaborator: Originally has the identity of a root account. After being added as a collaborator of the current root account, it becomes one of the sub-accounts, able to switch back to the root account identity.
Identity credentials: Include two kinds: **login credentials** which refer to user login name and password, and **access certificates** which refer to cloud API keys (SecretId and SecretKey).

Resource and Permission

Resource: An object operated in cloud services. In TDMQ for RabbitMQ, resources include clusters, Vhosts, Exchanges, Queues, routeRelations, etc.
Permission: It is an authorization that allows or forbids users to perform certain operations. By default, **a root account has access to all its resources**, while **a sub-account doesn't have access to any resources under the root account**.
Policy: It is a syntax rule that defines and describes one or more permissions. The **root account** performs authorization by **associating policies** with users/user groups.
Click to view more CAM documentation >>

References

Target	Link
Understand the Relationship Between Policy and User	Policy Management
Understand the Basic Policy Structure	Policy Syntax

Learn about which products support CAM.	Products supported by CAM

Using RabbitMQ with a Sub-Account

To ensure that sub-accounts can use RabbitMQ smoothly, the root account needs to authorize sub-accounts.
Log in to the CAM console as the root account. Find the corresponding sub-account in the sub-account list and click Grant in the Action column.
RabbitMQ provides two kinds of preset policies for sub-accounts: QcloudTDMQReadOnlyAccess and QcloudTDMQFullAccess. The former can only view relevant information on the console, while the latter can perform read-write and other related operations on the product console.

Except for the above preset policies, for convenience, the root account also needs to grant appropriate API call permissions of other cloud products to the sub-account according to your actual needs. The active RabbitMQ involves the corresponding API access permissions of the following cloud products:

Cloud Product	API Name	API functions	Role in RabbitMQ
Tencent Cloud Observability Platform (Monitor)	GetMonitorData	query metric monitoring data	View corresponding monitoring metrics in console display
Tencent Cloud Observability Platform (Monitor)	DescribeDashboardMetricData	query metric monitoring data	View corresponding monitoring metrics in console display
resource tag (Tags)	DescribeResourceTagsByResourceIds	Query resource tags	View resource tags of the cluster

To grant the above permissions to the sub-account, the root account also needs to perform the **create custom policy** operation on the **Policies** page of the CAM console. Click **by policy syntax** to create a new one, then select **blank template** and input the following policy syntax:

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "action": [
        "monitor:GetMonitorData",
        "monitor:DescribeDashboardMetricData",
        "tag:DescribeResourceTagsByResourceIds"
      ],
```

```
            "resource": [
              "*"
            ]
          }
        ]
    }
```

After the policy is created, in the operation list, just associate the created policy with the sub-account, as shown below:

Grant Sub-Account Operation-Level Permission

Last updated：2025-04-29 14:51:47

Application Scenario

This document introduces you to use the Tencent Cloud root account to perform operation-level authorization for sub-accounts. Based on your actual needs, you can grant different read/write permissions to sub-accounts.

Directions

Grant Full Read and Write Permissions

Notes:

After granting the sub-account full read and write permissions, the sub-account will have the full read and write capability for all resources under the root account.

1. Log in using the root account to the cloud access management console.
2. In the left sidebar, click Policy to enter the policy management list page.
3. In the search box on the right, enter **QcloudTDMQFullAccess** to search.

4. In the search results, click the **Associated Users/Groups** of **QcloudTDMQFullAccess** and select the sub-account to be authorized.

5. Click **Confirm** to complete authorization. This policy will appear in the user's policy list.

Grant Read-Only Permission

Notes:

After granting read-only permission to the sub-account, the sub-account will have the read-only capability for all resources under the root account.

1. Log in using the root account to the cloud access management console.
2. In the left sidebar, click Policy to enter the policy management list page.
3. In the search box on the right, enter **QcloudTDMQReadOnlyAccess** to search.

4. In the search results, click the **Associated Users/Groups** of **QcloudTDMQReadOnlyAccess** and select the sub-account to be authorized.

5. Click **Confirm** to complete authorization. This policy will appear in the user's policy list.

Grant Sub-Account Resource-Level Permission

Last updated：2025-04-29 14:52:34

Application Scenario

This task guides you to use the root account to perform resource-level authorization for the sub-account. The sub-account that obtains permission can gain the capability to control a specific resource.

Prerequisites

Have a Tencent Cloud root account and have already activated Tencent Cloud CAM.
At least one sub-account exists under the root account, and access authorization has been completed for the sub-account.
Have at least one RabbitMQ instance.

Directions

You can use the policy feature in the cloud access management console to authorize the RabbitMQ resources owned by the root account to the sub-account. The detailed operation of authorizing RabbitMQ **resources to the sub-account** is as follows. This example takes authorizing a cluster resource to the sub-account as an example, and the operation steps for other types of resources are similar.

Step 1: Obtain the "Cluster ID" of the RabbitMQ Cluster

1. Use **root account** to log in to TDMQ for RabbitMQ console, click Cluster List, and select the corresponding region.
2. In the **cluster list**, obtain the necessary "Cluster ID".

Step 2: Create an Authorization Policy

1. Enter the CAM console, and click Policies in the left sidebar.
2. Click Create Custom Policy and select Create by Policy Generator.
3. In the Visual Strategy Generator, keep the **effect** as **permission**, enter TDMQ in **service** to filter, and select **TDMQ (tdmq) message queue** from the results.

4. Select all operations in the operation. You can also select the operation type according to your own needs.
Notes:
Some APIs do not support resource authentication temporarily. The display on the console page shall prevail. For the list of APIs that support resource-level authorization, refer to the list of interfaces that support resource-level authorization in the appendix of this document.

5. Select a specific resource from resources, find Add Custom Resource Six-Segment Style. In the pop-up sidebar dialog box, fill in the cluster prefix and resource ID. For the retrieval process, see **Step 1**.

Parameter	Description
Effect	Select **Allow**
Service	Select **Message Queue (TDMQ) (tdmq)**
Action	Select **read operation**
Resource	Select **specific resource**, click **add custom resource six-segment style**. Region: Select the region where the resource is located. Account: system auto-fill resource prefix: cluster fill in the ID of the cluster you want to authorize
Condition	Allow access to the specified action only when the request comes from the specified IP range.

6. Click Next and fill in the policy name as needed.
7. Click **Select User** or **Select User Group** to choose the user or user group to whom you want to grant resource permissions.

8. Click **Complete**. The sub-account with granted resource permissions will then have the ability to access related resources.

Granting Sub-Accounts Tag-Level Permissions

Last updated：2025-04-29 14:53:27

Application Scenario

This task guides you to authorize sub-accounts to access resources under a specific tag using the root account and tag-based authentication. The authorized sub-account can gain control capability over resources with the corresponding tag.

Prerequisites

Have a Tencent Cloud root account and have already activated Tencent Cloud CAM.
At least one sub-account exists under the root account, and access authorization has been completed for the sub-account.
Have at least one RabbitMQ cluster resource instance.
Have at least one **tag**. If you don't have one, you can go to Tag Console > **Tag List** to create one.

Directions

You can use the policy feature in the cloud access management console to grant sub-accounts read/write permissions to RabbitMQ resources owned by the root account and already bound with tags via tag-based authorization. The detailed steps for granting resource permissions to sub-accounts by tags are as follows.

Step 1: Bind a Tag to the Resource

1. Use **root account** to log in to TDMQ for RabbitMQ console and open the cluster management page.
2. Check the target cluster, click **Edit Resource Tags** in the upper left corner, and bind resource tags to the cluster.

Step 2: Authorize by Tag

1. Enter the CAM console, and click **Policies** in the left sidebar.
2. Click Create Custom Policy, and select Tag-based Authorization.
3. In the Visual Strategy Generator, enter tdmq in Service to filter. Select Tencent Distributed Message Queue (tdmq) from the results. In Operations, choose All Operations. You can also select corresponding operations as needed.

4. In the tag location selection, select the tag key and tag value for cluster resource binding. Multiple tags are in an OR relationship, and only need to meet one of them.
5. In the condition key selection position, select the appropriate condition key. The condition keys resource_tag and request_tag allow multiple selections or you can select just one.

6. Click next and fill in the policy name as needed.
7. Click **Select User** or **Select User Group** to choose the user or user group to whom you want to grant resource permissions.

8. Click **Complete**. Authorized sub-accounts can then control resources under specified tags based on policy control.

Manage Resource Tags in a Unified Way

You can also manage resource tags in a unified way in the Tag Management Console. The detailed operations are as follows:
1. **Log in to Tencent Cloud.**
2. Select Resource Tag in the left sidebar. Select query conditions as needed, and select TDMQ > Cluster in Resource Type.
3. Click to query resource.
4. Check the needed resources in the results, click Edit Tag, and then you can bulk bind or unbind tags.

Tag Management

Last updated：2025-04-29 15:38:56

Overview

Tag is a marker provided by Tencent Cloud to identify cloud resources. It is a Key-Value pair. Tags can help you classify and manage TDMQ for RabbitMQ resources from various dimensionalities, such as business, purpose, and owner.

Notes:

Tencent Cloud will not use the tags you set. Tags are only used for your management of TDMQ for RabbitMQ resources.

Use Limits

Quantity Limit

One resource can bind up to 50 tags.

Naming Restrictions

Tag Key	Tag Value
Tag keys starting with `qcs:`, `project:`, or `project` are system-reserved tag keys, and creation of system-reserved tag keys is prohibited.	-
In UTF-8, a tag key must contain at least 1 and no more than 127 Unicode characters.	In UTF-8, a tag value must contain at least 1 and no more than 255 Unicode characters.
Support characters encoded in UTF-8 format, spaces, digits, and special characters. Does not support names starting or ending with a space. Support in English: + - = . _ : / @ ( ) [ ] , ; > Support in Chinese mode: + - = / @ ( ) [ ] :	
Case-sensitive.	

Usage Examples

Case Description

Case: A certain company has 6 TDMQ for RabbitMQ edition clusters on Tencent Cloud. The information of the use departments, business scopes and persons in charge of these 6 clusters is as follows:

Queue ID	Department	Business Scope	Owner
amqp-78383dp8p8w1	E-commerce	Marketing campaigns	Zhang San
amqp-78383dp8p8w2	E-commerce	Marketing campaigns	Wang Wu
amqp-78383dp8p8w3	Games	Game A	Li Si
amqp-78383dp8p8w4	Games	Game B	Wang Wu
amqp-78383dp8p8w5	Entertainment	Post-production	Wang Wu
amqp-78383dp8p8w6	Entertainment	Post-production	Zhang San

For the amqp-78383dp8p8w1 instance, you can add the following three sets of tags:

Tag Key	Tag Value
dept	ecommerce
business	mkt
owner	zhangsan

Similarly, you can also set appropriate tags for other resources based on their department, business scope, and owner information.

Set Tag in TDMQ for RabbitMQ Console

As detailed above, after designing the tag key and tag value, you can log in to the TDMQ for RabbitMQ console to set the tags.

1. Log in to the TDMQ for RabbitMQ console.

2. On the cluster management list page, select the region, check the clusters for which you want to edit tags, and click **Edit Resource Tag** at the top of the page.

3. In the pop-up "edit tag" window, set the tag.

For example, add three sets of tags to the amqp-78383dp8p8w1 cluster.

Notes:

If the existing tags do not meet your requirements, please go to Tag Management to create a tag.

4. Click Confirm. A prompt indicating successful modification will appear, and the bound tags can be viewed in the resource tag column of the cluster.

Filtering Resources by Tag Key

You can filter out clusters bound to a specific tag in the following steps:

1. Select **tag** in the search box at the top-right corner of the page.
2. Select the tag you want to search for in the window that pops up after **Tag:**, and click **Confirm** to perform the search.
3. For example: Select `Tag: owner:zhangsan` to filter out the cluster bound with the tag key `owner:zhangsan` .

Editing a Tag

1. On the cluster management list page, after selecting a region, check the needed cluster, and click Edit Resource Tag at the top of the page.

Notes:

Batch editing of tags is supported for up to 20 resources.

2. In the pop-up "edit tag" window, add, modify, or delete tags based on actual needs.

Migration to Cloud
Migrating RabbitMQ to Cloud

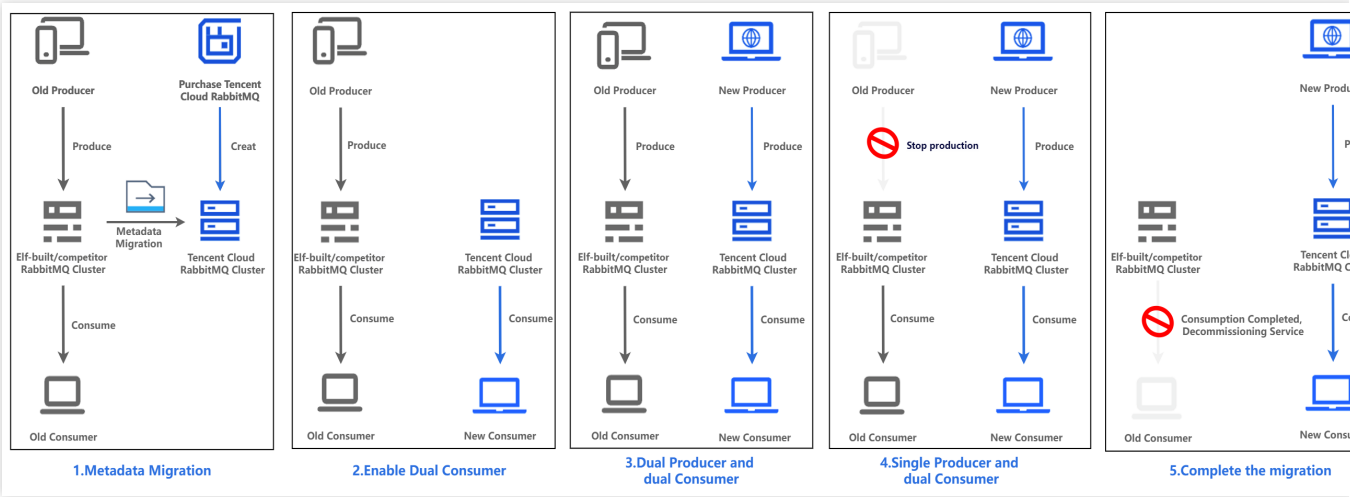
Last updated：2024-08-07 14:24:45

Overview

This document provides an overall introduction to a feasible scheme for migrating self-built/competitor RabbitMQ clusters to TDMQ for RabbitMQ clusters.

Scheme Description: Dual-Producer Dual-Consumer

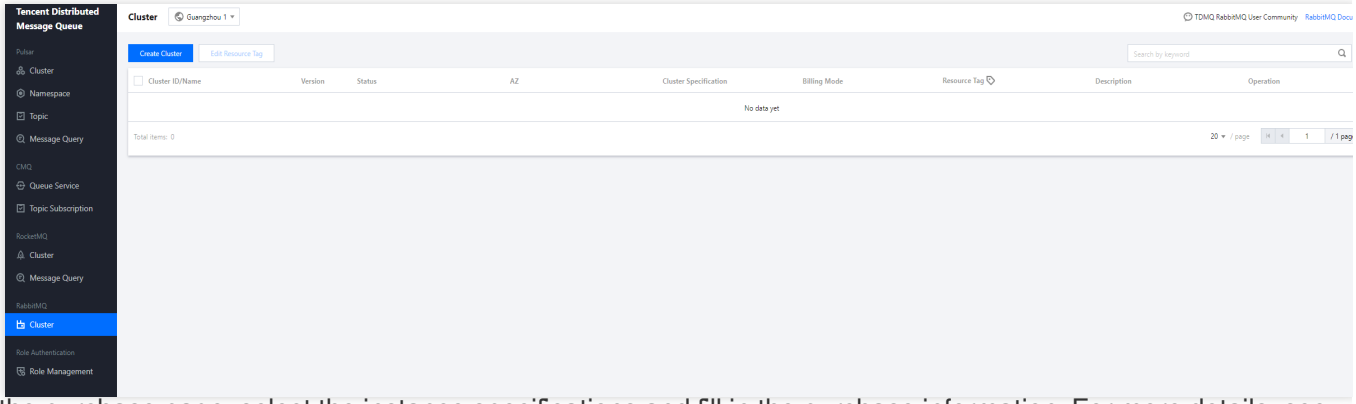
This scheme is simple, clear, and easy to operate, with no data heap, ensuring timely message consumption.
Directions are as follows:
1. Purchase a RabbitMQ cluster on Tencent Cloud. In the initial status, producer and consumer operations are performed on the old cluster. Once the new cluster is ready, complete the metadata migration of RabbitMQ.
2. Deploy a new consumer side to the new cluster and the dual-producer mode is enabled.
3. Deploy a new producer side and gradually switch the traffic to the new cluster to enable the dual-producer dual-consumer mode.
4. Stop the production traffic of the old cluster, temporarily retain the consumer side of the old cluster, and enter the single-producer dual-consumer mode.
5. After the data on the old cluster is consumed and confirming that there are no heaped messages, disable the old consumption service to complete the migration process.


Old Producer
Purchase Tencent Cloud RabbitMQ
Produce
Creat
Metadata Migration
Elf-built/competitor RabbitMQ Cluster
Tencent Cloud RabbitMQ Cluster
Consume
Old Consumer
1.Metadata Migration
Old Producer
Produce
Elf-built/competitor RabbitMQ Cluster
Tencent Cloud RabbitMQ Cluster
Consume
Consume
Old Consumer
New Consumer
2.Enable Dual Consumer
Old Producer
New Producer
Produce
Produce
Elf-built/competitor RabbitMQ Cluster
Tencent Cloud RabbitMQ Cluster
Consume
Consume
Old Consumer
New Consumer
3.Dual Producer and dual Consumer
Old Producer
New Producer
Stop production
Produce
Elf-built/competitor RabbitMQ Cluster
Tencent Cloud RabbitMQ Cluster
Consume
Consume
Old Consumer
New Consumer
4.Single Producer and dual Consumer
New Produ
Elf-built/competitor RabbitMQ Cluster
Consumption Completed, Decommissioning Service
Old Consumer
New Consumer
5.Complete the migration

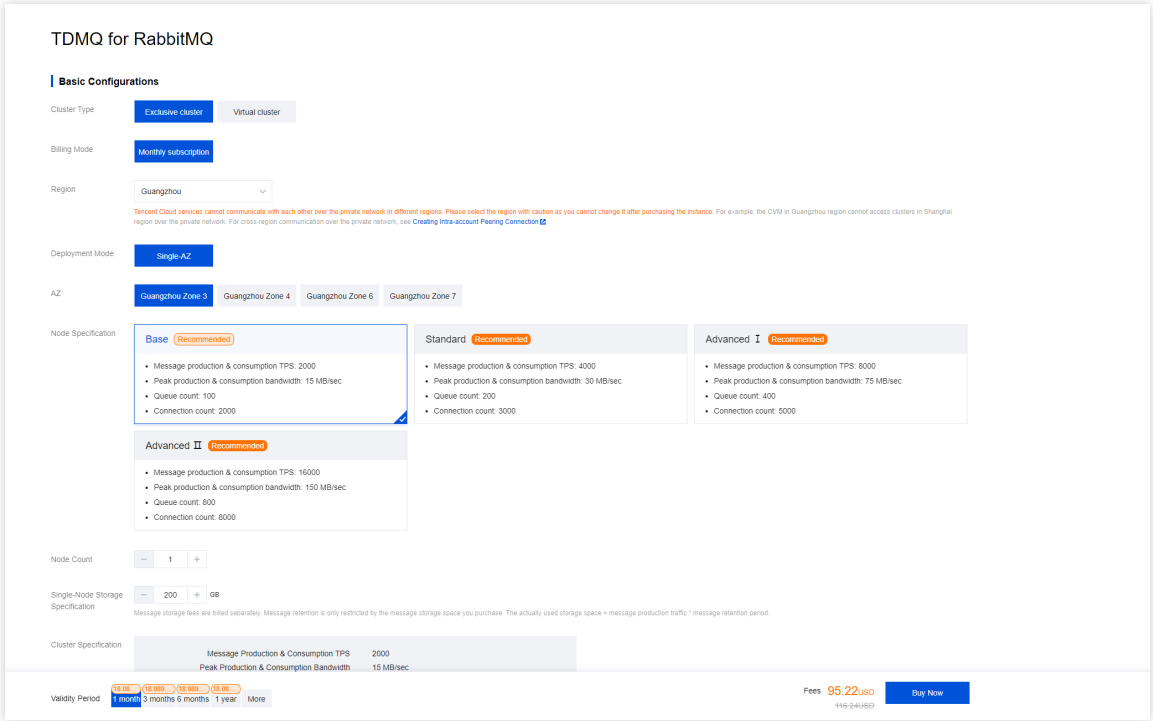
Step 1. Purchasing a TDMQ Instance

Last updated：2024-10-18 17:37:45

1. Log in to the TDMQ console.
2. In the left sidebar, choose **Cluster Management** > **Cluster List** , and then click **Create Cluster** to enter the purchase page.



3. On the purchase page, select the instance specifications and fill in the purchase information. For more details, see Purchase Method.



4. Select **I have read and agree to TDMQ for RabbitMQ Terms of Service** and click **Buy Now**.
5. On the order payment page, click **Pay** and wait 3–5 minutes. Then, you can see the created cluster on the **Cluster** page.

Step 2. Migrating Metadata to Cloud

Last updated：2024-10-18 17:37:45

Overview

This document describes how to migrate metadata from an open-source RabbitMQ cluster to a TDMQ for RabbitMQ cluster.

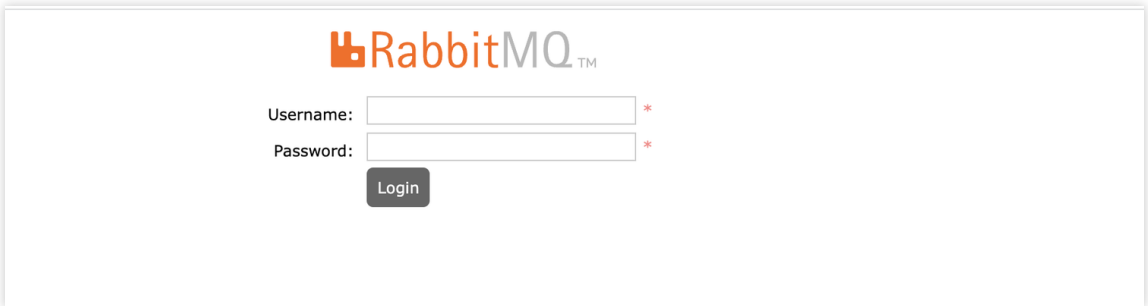
Prerequisites

You have exported the metadata file from open-source RabbitMQ.

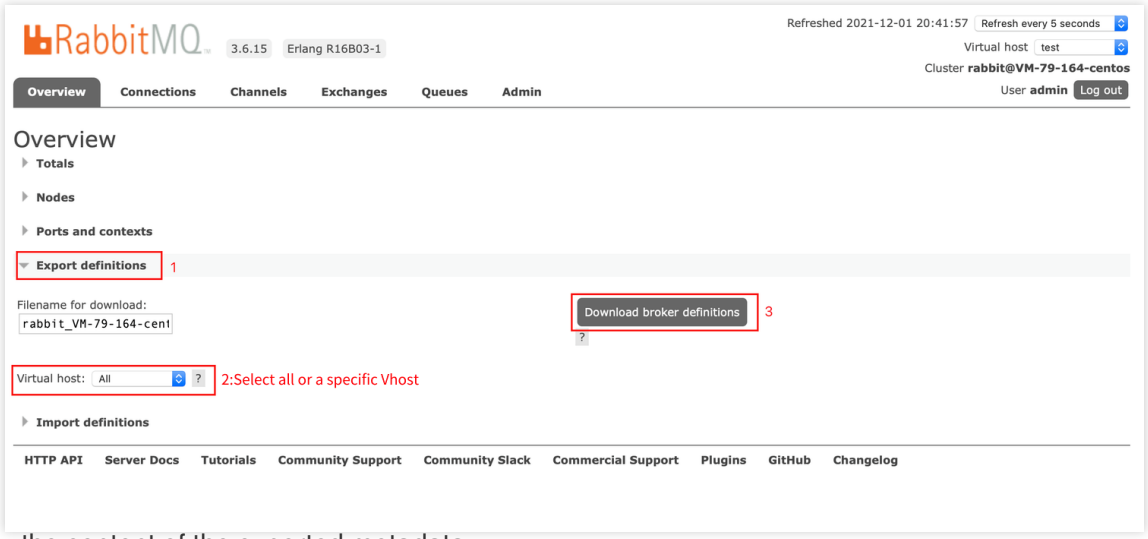
Directions

Exporting the metadata from a self-built RabbitMQ cluster

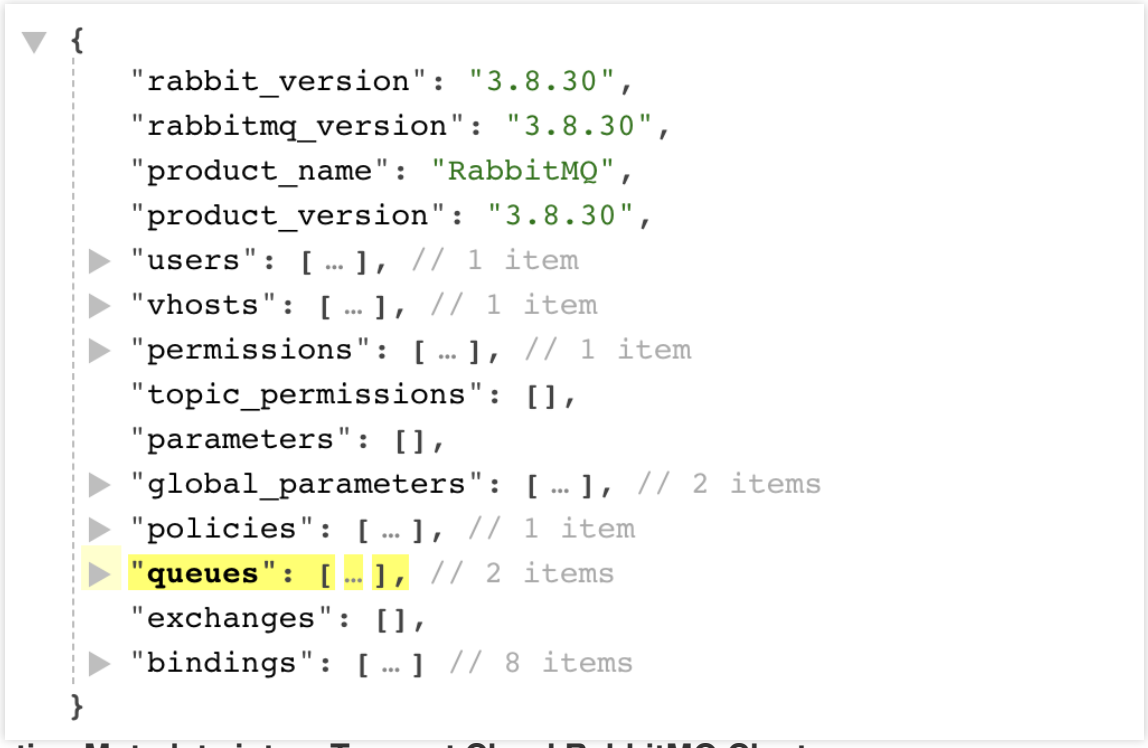
1. Log in to the RabbitMQ console.
2. In the left sidebar, choose **Cluster Management** > **Cluster List**. After the region is selected, click the ID of the target cluster to enter the basic information page of the cluster.
3. In the **Web console access address** module, view the username and password, and click the public network access address.
4. Use the username and password from the previous step to log in to the self-built open-source RabbitMQ console.



5. On the **Overview** tab page, click **Export definitions**, and enter the filename for download. Select All or a specific vhost for the Virtual host field and click **Download broker definitions** on the right to export the metadata file of all Vhosts or the specified vhost.

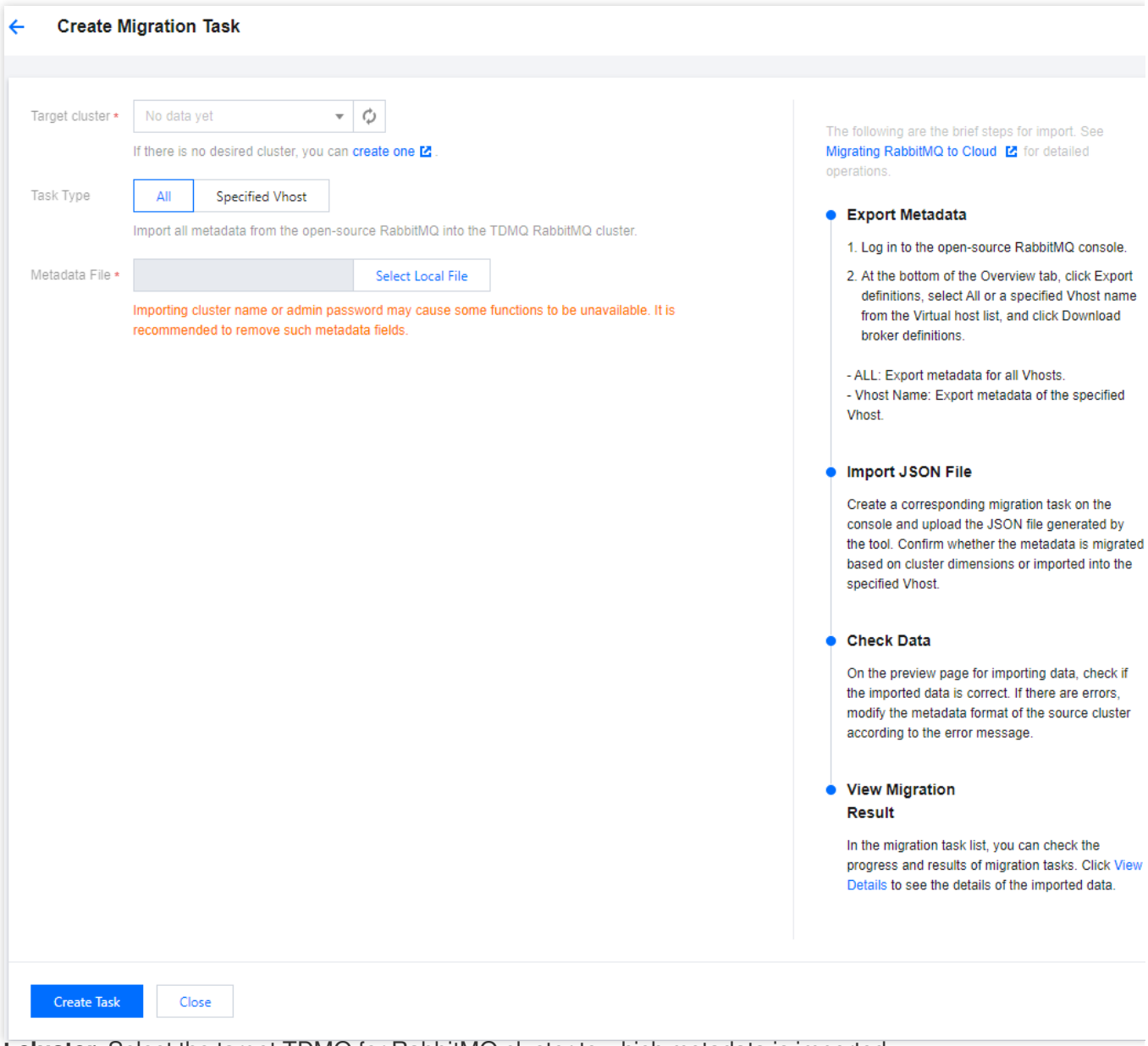


6. View the content of the exported metadata.



Importing Metadata into a Tencent Cloud RabbitMQ Cluster

1. Log in to the Tencent Cloud console, and click **Migrate to Cloud** .
2. On the **Cloud Migration Task List** page, click **Create Task** .



Target cluster: Select the target TDMQ for RabbitMQ cluster to which metadata is imported.

Task Type

All: Import all metadata from the open-source RabbitMQ cluster into the TDMQ for RabbitMQ cluster.

Specified Vhost: Import specified Vhost metadata from the open-source RabbitMQ cluster into the specified Vhost of the TDMQ for RabbitMQ cluster.

Metadata File: Select a local metadata file.

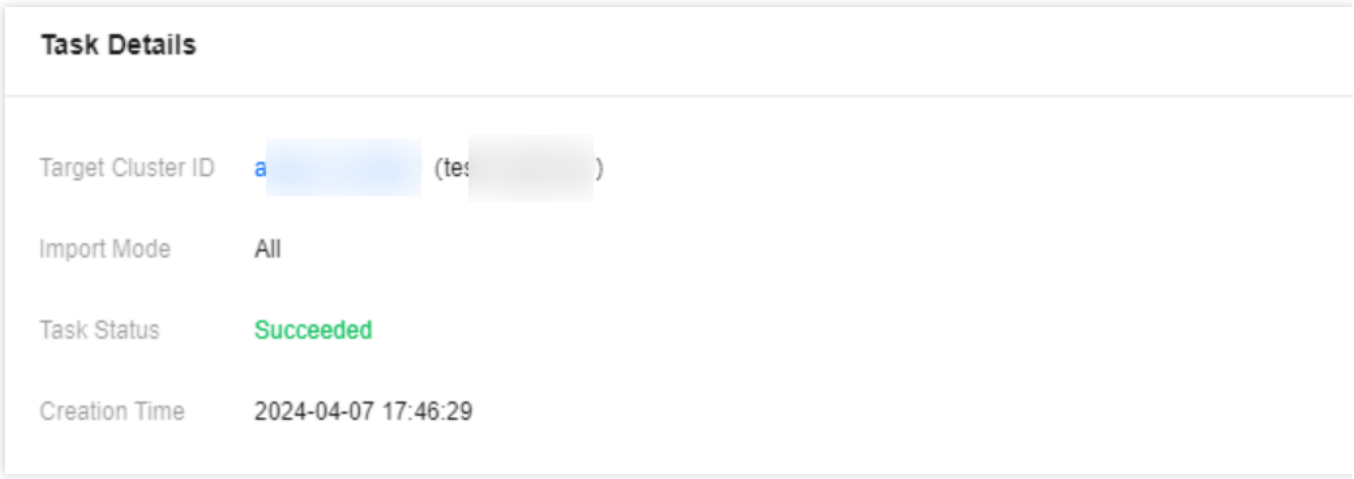
Note:

Importing the cluster name or admin password may disable some features. It is recommended that you remove such metadata fields.

3. Check the data to import on the preview page to ensure its accuracy. If there are any errors, modify the source cluster metadata format based on the error message.

4. Click **Create Task** . A task record is generated on the **Cloud Migration Task List** page.

5. Click **View Details** to view details of this migration task.


Task Details
Target Cluster ID
Import Mode
All
Task Status
Succeeded
Creation Time
2024-04-07 17:46:29

Step 3. Migrating Data to Cloud

Last updated：2024-08-07 14:25:27

Overview

This document mainly introduces how to use the dual-producer dual-consumer scheme to switch the services of a self-built RabbitMQ cluster to TDMQ for RabbitMQ.

Scheme: Dual-Producer Dual-Consumer Mode

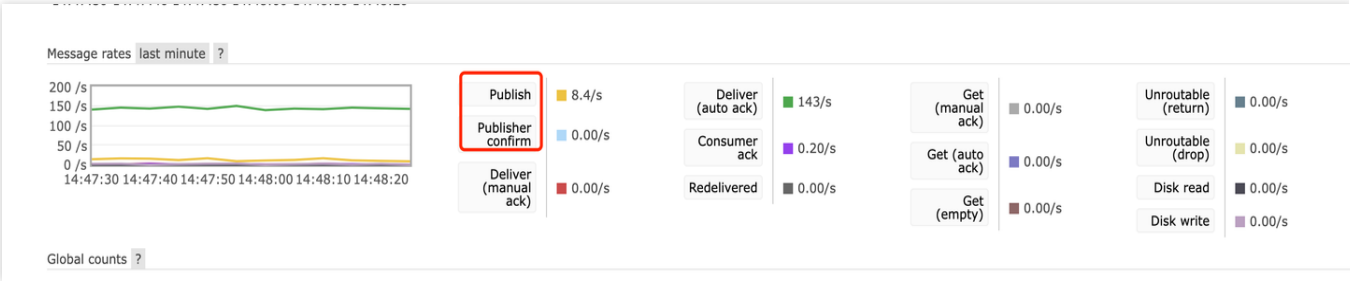
Prerequisites

1. Purchased a RabbitMQ Instance on Cloud
2. Migrated the metadata of the self-built RabbitMQ cluster to Tencent Cloud RabbitMQ.

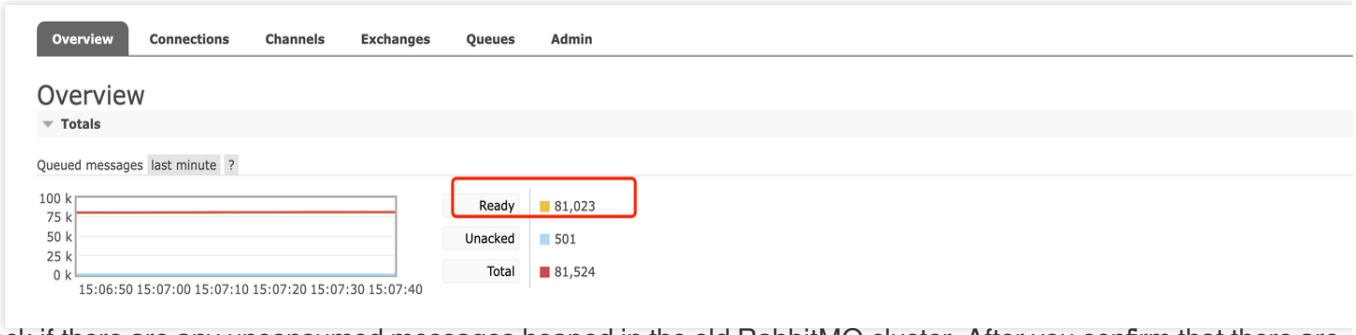
Directions

1. Switch the access information for some nodes in the consumer cluster and transfer these consumers to the new RabbitMQ cluster. These consumers will consume messages from the new RabbitMQ cluster, while the remaining consumers will continue to consume messages from the old RabbitMQ cluster.
2. Switch the access information for some nodes in the producer cluster and transfer these producers to the new RabbitMQ. These producers will send messages to the new RabbitMQ cluster, while the remaining producers will continue to send messages to the old RabbitMQ cluster. To prevent duplicate or lost messages, you can implement idempotent logic for message consumption in advance.
3. Transfer the remaining producers to the new RabbitMQ cluster. Then, all messages will be sent to the new RabbitMQ cluster.

Tips 1: You can confirm that the production traffic of the self-built RabbitMQ cluster has stopped in the community management console.

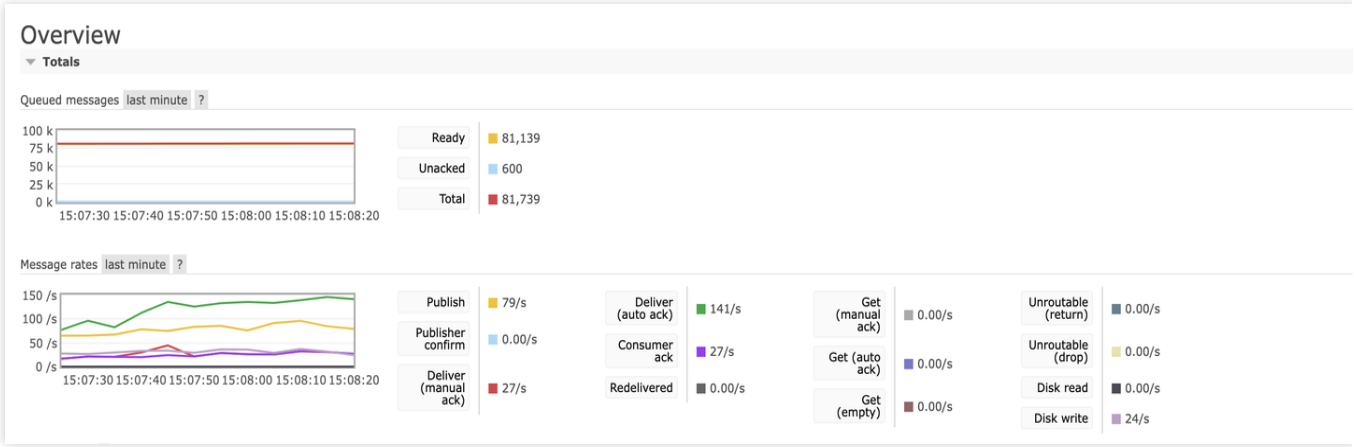


Tips 2 : You can confirm that the heaped messages in the self-built RabbitMQ cluster are decreasing in the community management console.



4. Check if there are any unconsumed messages heaped in the old RabbitMQ cluster. After you confirm that there are no heaped or unhandled messages in the old RabbitMQ cluster, transfer the remaining consumers to the new RabbitMQ cluster to complete the migration of the entire data flow.

Tips: Confirm the message production and consumption in the cloud RabbitMQ cluster, and ensure there is no message heap.



Note:

Follow the above steps strictly. If you switch producers first and then switch consumers, message loss may occur. Before you switch the remaining consumers, ensure that all messages in the old RabbitMQ cluster have been consumed to avoid any missed consumption.

Possible Issues

Order Issue

Due to the cluster switch, the order of messages cannot be guaranteed during the switch process. There may be partial disorder during the switch.

Message Duplication

In theory, Message will not duplicate, but in extreme cases it can occur. For example, during the switch process, a consumer has consumed a message but has not sent an ACK to the server (the old RocketMQ cluster). This can

cause the message to enter the retry queue, leading to duplicate consumption. Implementing idempotent logic for the messages can avoid this issue.

Consumption Delay

During the consumer switch process, the reallocation of partitions requires rebalance between queues and consumer clients, which may cause short consumption delays. No additional operations are needed in this situation, and operations will resume once the switch is completed.