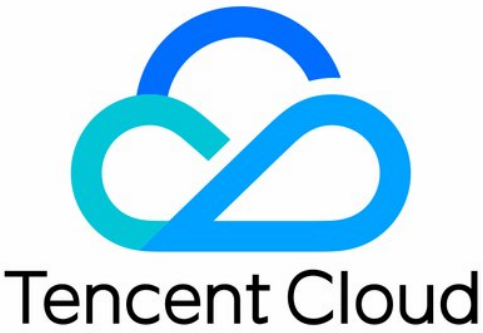
TencentDB for MongoDB

Operation Guide

Product Documentation



Contents

Operation Guide
Access Management
Overview

Last updated：2024-01-15 14:40:06

Cloud Access Management (CAM) is a web-based Tencent Cloud service that helps you securely manage and control access permissions to your Tencent Cloud resources. Using CAM, you can create, manage, and terminate users (groups), and control the Tencent Cloud resources that can be used by the specified user through identity and policy management.

Background

If you have multiple users managing different Tencent Cloud services such as CVM, VPC, and TencentDB, and they all share your Tencent Cloud account access key, you may face the following problems:
Your key will be easily compromised because it is shared by several users.
You cannot restrict the access from other users and your service will be vulnerable to the security risks caused by their maloperations.

Basic Concepts

Root account
When you sign up for a Tencent Cloud account, the system creates a root account identity for you to log in to Tencent Cloud services. Tencent Cloud records your usage and bills you based on the root account. The root account has full access to the resources under it by default and can create sub-accounts and set permissions for them.
Sub-account
A sub-account is created by and belongs to the root account. Every sub-account has a definite ID and identity credential.
Identity credential
An identity credential includes a **login credential** and an **access certificate**. The former refers to a user's login name and password. The latter refers to Tencent Cloud API keys (SecretId and SecretKey).
Resource
A resource is an object manipulated in Tencent Cloud services, such as a TencentDB for MongoDB instance.
Permission
It is an authorization that allows or forbids users to perform certain operations. By default, the **root account** has full

access to all resources under the account, while a **sub-account** does not have access to any resources under its root account.

Policy

It is a syntax rule that defines and describes one or more permissions. By default, a sub-account has no access to Tencent Cloud services or resources. To grant a sub-account such access, you need to create a CAM policy.

References

For more information on access management, see CAM Overview.

Authorization Policy Syntax

Last updated：2024-01-15 14:40:06

A policy is a syntactic specification of a user permission set, which accurately describes the authorized resource set, operation set, and authorization conditions.

CAM Policy Syntax

```
{
        "version":"2.0",
        "statement":
        [
            {
                "effect":"effect",
                "action":["action"],
                "resource":["resource"],
                 "condition": {"key":{"value"}}
            }
        ]
}
```

The following table describes policy statements.

Parameter	Subparameter	Required	Description
version	N/A	Yes	Currently, only the value `2.0` is allowed.
statement	effect	Yes	It describes the result of a statement. The result can be "allow" or an "explicit deny".
	action	Yes	It describes the allowed or denied operation which can be an API or a feature set (a set of specific APIs prefixed with `permid`).
	resource	Yes	It describes the details of authorization. All resources can be described in the six-segment format. Each service has its own resources and detailed resource definition.
	condition	Yes	It describes the condition for the policy to take effect. A condition consists of operator, action key, and action value. A condition value may contain information such as time and IP address. A condition value may be the time, IP address, etc. Some services allow you to specify additional values in a condition.

Note:

The **statement** element describes the details of one or more permissions. This element contains a permission or permission set of other elements such as `effect`, `action`, `resource`, and `condition`. One policy has only one `statement`.

Defining an action

In a CAM policy statement, you can specify any API operation from any service that supports CAM. APIs prefixed with `mongodb:` should be used for TencentDB for MongoDB, such as `mongodb:BackupDBInstance` or `mongodb:CreateAccountUser`.

To specify multiple operations in a single statement, separate them by comma:

```
"action":["mongodb:action1","mongodb:action2"]
```

You can also specify multiple operations by using a wildcard. For example, you can specify all operations beginning with "Describe" in the name as shown below:

```
"action":["mongodb:Describe*"]
```

If you want to specify all operations in MongoDB, use a wildcard "*" as shown below:

```
"action"：["mongodb:*"]
```

Defining a resource

Each CAM policy statement has its own applicable resources. Resource paths are generally in the following format:

```
qcs:project_id:service_type:region:account:resource
```

Project_id describes the project information, which is only used to enable compatibility with legacy CAM logic and can be left empty.

service_type describes the product abbreviation, such as `mongodb`.

region describes the region information, such as `bj`.

account describes the root account of the resource owner, such as `uin/12345678`.

resource describes the detailed resource information of each product, such as `instance/instance_id` or `instance/*`.

You can set `resource` to an instance ID (cmgo-aw6g1g0z) in a statement as shown below:

```
"resource":[ "qcs::mongodb:bj:uin/12345678:instance/cmgo-aw6g1g0z"]
```

You can also use the wildcard "*" to specify all instances that belong to a specific account as shown below:

```
"resource":[ "qcs::mongodb:bj:uin/12345678:instance/*"]
```

If you want to specify all resources or if a specific API operation does not support resource-level permission control, you can use the wildcard "*" in the `resource` element as shown below:

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

If you want to specify multiple resources in a single command, separate them by comma. In the following example, two resources are specified:

```
"resource":["resource1","resource2"]
```

The table below describes the resources that can be used by MongoDB and the corresponding resource description methods, where words prefixed with `$` are placeholders, `region` refers to a region, and `account` refers to an account ID.

Resource Type	Resource Description Method in Authorization Policy
Instance	`qcs::mongodb:$region:$account:instance/*` `qcs::mongodb:$region:$account:instance/$instanceId`
VPC	`qcs::vpc:$region:$account:vpc/$vpcId`
Security Group	`qcs::cvm:$region:$account:sg/$sgId`

Default Permission Policy of TencentDB for MongoDB

TencentDB for MongoDB supports the following system permission policies.

Policy Name	**Note**
`QcloudMongoDBFullAccess`	TencentDB for MongoDB management permission. A Tencent Cloud sub-account granted with this permission has the same permissions as the root account, including all permissions of console and API operations.
`QcloudMongoDBReadOnlyAccess`	Read-only permission. A Tencent Cloud sub-account granted with this permission has only the read-only permission of all resources under the Tencent Cloud root account but not operation permissions of the console and APIs.

The system permission policy `QcloudMongoDFullAccess` is as follows:

```
{
```

```
    "version": "2.0",
    "statement": [
        {
            "action": [
                "monitor:GetMonitorData",
                "monitor:DescribeBaseMetrics",
                "mongodb:*"
            ],
            "resource": "*",
            "effect": "allow"
        }
    ]
}
```

The system permission policy `QcloudMongoDBReadOnlyAccess` is as follows:

```
{
    "version": "2.0",
    "statement": [
        {
            "action": [
                "monitor:GetMonitorData",
                "monitor:DescribeBaseMetrics",
                "mongodb:Describe*"
            ],
            "resource": "*",
            "effect": "allow"
        }
    ]
}
```

Custom Permission Policy of TencentDB for MongoDB

Currently, TencentDB for MongoDB supports custom policies for the following resource-level permissions.

Note:

TencentDB API operations not listed here do not support resource-level permissions. You can still authorize a user to perform such a TencentDB API operation, but you must specify * as the resource element of the policy statement.

Action Name	Permission Description	Resource Description
BackupDBInstance	Backs up a database instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
CreateAccountUser	Creates an account	qcs::mongodb:$region:$account:instance/*

		qcs::mongodb:$region:$account:instance/$instanceId
CreateDBInstanceHour	Creates a pay-as-you-go TencentDB for MongoDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DeleteAccountUser	Deletes an account	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeAccountUsers	Queries the user information of an account	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeBackupAccess	Gets the permission to download an instance backup	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeBackupRules	Gets the backup rules of a TencentDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeClientConnections	Gets the number of client connections	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeDBBackups	Queries the list of backups of an instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeDBInstances	Queries the list of database instances	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeInstanceDB	Queries the collection/database information of an instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeSlowLog	Gets the slow log information	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeSlowLogPattern	Gets the slow log statistics	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
DescribeSpecInfo	Queries purchasable instance specifications	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
ExchangeInstance	Replaces the original instance with a temp instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId

IsolateDBInstance	Isolates a TencentDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
ModifyDBInstanceSpec	Adjusts the configurations of a TencentDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
OfflineIsolatedDBInstance	Deactivates an isolated TencentDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
RemoveCloneInstance	Deletes a temp instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
RenameInstance	Renames an instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
RenewInstance	Renews a TencentDB instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
ResizeOplog	Adjusts the oplog size of an instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
RestartInstance	Restarts an instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
RestoreDBInstance	Restores a database instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetAccountUserPrivilege	Sets user permissions	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetAutoRenew	Sets auto-renewal	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetBackupRules	Sets backup rules	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetInstanceFormal	Promotes a temp instance to the primary instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetInstanceMaintenance	Sets the instance maintenance time	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
SetPassword	Sets password	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId

SetReadOnlyToNormal	Promotes a read-only instance to the primary instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
TerminateDBInstanceHour	Terminates a pay-as-you-go instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId
UpgradeDBInstanceHour	Upgrades a pay-as-you-go instance	qcs::mongodb:$region:$account:instance/* qcs::mongodb:$region:$account:instance/$instanceId

Custom permission policy example

If you want to grant an account the `CreateDBInstance` and `CreateAccountUser` permissions on the "cmgo-aw6g****" instance, you can create a policy as follows:

```
{
    "version": "2.0",
    "statement": [
        {
            "effect": "allow",
            "action": [
                "mongodb:CreateDBInstance",
                "mongodb:CreateAccountUser"
            ],
            "resource": [
                "qcs::mongodb::uin/100001540306:instanceId/cmgo-aw6g****"
            ],
            "condition": {
                "ip_equal": {
                    "qcs:ip": [
                        "10.0.0.4"
                    ]
                }
            }
        }
    ]
}
```

Creating a custom permission policy

You can create a custom policy on the Policies page in the CAM console. For detailed directions, see Creating Custom Policy.

Authorization Permission Policy

Last updated：2024-01-15 14:40:06

Permissions of Tencent Cloud root accounts and sub-accounts are separated. You can grant sub-accounts different permissions as needed, which avoids security risks caused by exposure of your Tencent Cloud account key.

Granting a sub-account a permission policy

Background

Company A activates the TencentDB for MongoDB service and wants its team members to manipulate the involved resources. For security or trust considerations, it doesn't want to directly disclose its Tencent Cloud account key to the team members; instead, it wants to create corresponding sub-accounts for them. The sub-accounts can manipulate resources only with authorization by its root account and separate usage calculation and billing are not required, as all fees are charged to its Tencent Cloud account. It also wants to be able to revoke or delete the operation permissions of sub-accounts at any time.

Directions

Step 1. Create a sub-account user

You can create a sub-account user through the console or an API.
Log in to the CAM console and enter the User List page to create a user. For detailed directions, see Creating Sub-User.
Create a sub-user and configure permissions for them by calling the `AddUser` API with an access key. For more information, see AddUser.

(Optional) Step 2. Create a custom permission policy

1. On the Policies page in the CAM console, search for a target policy by policy name in the search box in the top-right corner.
2. If the permission policy does not exist, you need to customize one. For detailed directions, see Creating Custom Policy.

Step 3. Assign the permission policy to the sub-account

On the Policies page in the CAM console, find the target permission policy and associate it with the target sub-account. For detailed directions, see Authorization Management.
On the User List page in the CAM console, find the target sub-account and associate them with the target policy. For detailed directions, see Authorization Management.

References

Logging in to the console

You can let your team members use a sub-account to log in to the Tencent Cloud console and access TencentDB for MongoDB. For detailed directions, see Logging in to Console with Sub-account.

Modifying a sub-account

You can view and modify the information of a sub-account as instructed in User Information.

Deleting a sub-account

You can revoke or delete the operation permissions of a sub-account as instructed in Deleting Sub-Users.

Granting a permission policy across Tencent Cloud accounts

Background

Company A activates TencentDB for MongoDB and wants company B to have part of the permissions of its TencentDB for MongoDB operations, such as instance read/write and slow query operation. Company B wants to have a sub-account to take care of such businesses. In this case, company A can authorize the root account of company B to access TencentDB for MongoDB resources through a role. For the specific concept and use cases of role, see Role Overview.

Directions

Step 1. Company A creates a role for company B

1. Log in to the CAM console and go to the Roles page.
2. Click **Create Role**. In the **Select role entity** window, select **Tencent Cloud Account**
3. On the **Create Custom Role** page, create a role.

a. On the **Enter role entity info** page, select **Other root account** as **Tencent Cloud account**, enter the root account of company B as **Account ID**, set other parameters as prompted, and click **Next**.

b. On the **Configure Role Policy** page, select the target policy and click **Next**.

c. On the **Review** page, enter a role name such as `DevOpsRole` in the **Role Name** box, review the selected policy, and click **Complete**.

Step 2. Company B grants a sub-account the permission to assume the role

1. On the Policies page in the CAM console, click **Create Custom Policy**.
2. In the **Select Policy Creation Method** window, select **Create by Policy Syntax**.
3. On the **Create by Policy Syntax** page, create a policy.

a. In the **Select a template type** section, select **Blank Template** and click **Next**.

b. On the **Edit Policy** page, enter a policy name such as `sts:AssumeRole` in the **Policy Name** input box.

c. In **Policy Content**, set the policy content according to the policy syntax and click **Complete**. Below are examples:

```
{
 "version": "2.0",
 "statement": [
 {
     "effect": "allow",
     "action": ["name/sts:AssumeRole"],
     "resource": ["qcs::cam::uin/12345:RoleName/DevOpsRole"]
 }
 ]
}
```

4. Return to the **Policies** page, find the created custom policy, and click **Associate Users/Groups** in the **Operation** column.

5. Associate the custom policy with the sub-account of company B and click **OK**.

Step 3. Company B uses the sub-account to access Tencent Cloud resources through the role

1. Log in to the console with the sub-account of company B and select **Switch Role** in the profile photo drop-down list.

2. On the role switch page, enter the root account of company A and role name to switch to the role of company A.

References

You can modify a role as instructed in Modifying Role.

You can delete a role as instructed in Deleting a Role.

For more information on how to use CAM, see Overview.

Instance Management
Viewing Instance Details

Last updated：2024-10-10 16:44:27

Overview

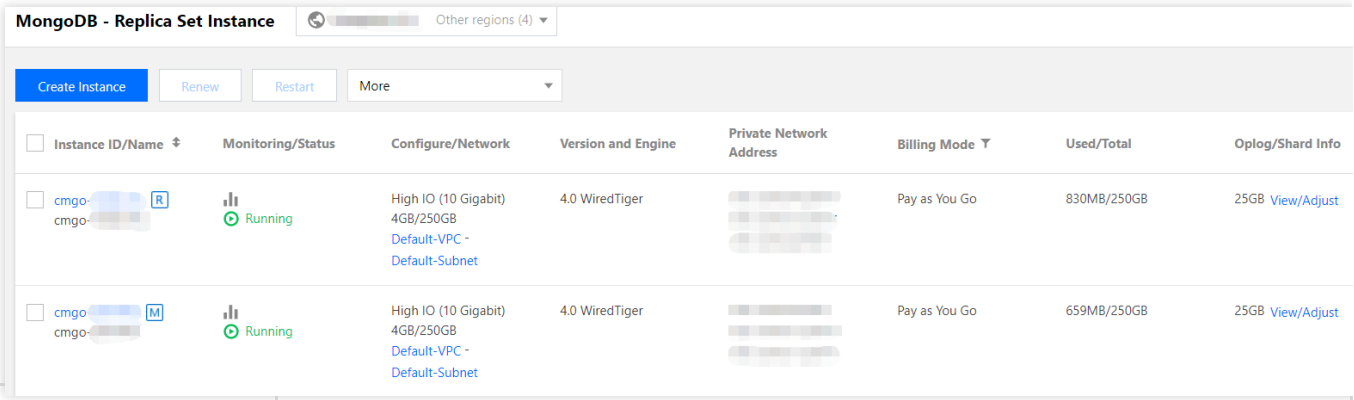
After purchasing a TencentDB for MongoDB instance, you can quickly view its details in the console, such as the status, capacity usage, primary/secondary nodes in the cluster, and network status. You can also perform Ops and management operations efficiently.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The instance hasn't been terminated or isolated into the recycle bin. For more information, see Recycle Bin.

Viewing the Instance List

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Shard Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
In the search box in the top-right corner, you can search for the target instance by instance ID, instance name, private IP, or tag key.
If you can't find the target instance in the instance list, select **Recycle Bin** on the left sidebar to check whether it is isolated there due to overdue payments. For more information, see Recycle Bin.
5. View the target instance information, such as the status, specification, and storage engine.



Parameter	Description
Instance ID/Name	Instance ID: The instance's unique ID. Name: The instance name set during instance creation. You can hover over the instance name and click to rename the instance for easier identification and management.
Monitoring/Status	Monitoring: You can click to enter the monitoring panel and view the instance monitoring data. For more information, see Viewing Monitoring Data. Status: Instance status. The normal status is **Running**. If a task is being executed in the instance, the task name, such as **Changing configuration**, will be displayed here.
Specification/Network	Specification Type: Supports High IO 10G Type and Cloud Edition. For more information, see Product Specification Types. Specification: The specification of each instance node. Replica set: Memory/Disk capacity. Sharded cluster instance: Memory/Disk capacity × shard quantity. Network: The network information of the instance.
Version and Engine	Supports database versions of 6.0, 5.0, 4.4, 4.2, 4.0, 3.6, and 3.2. The 3.2 version is no longer available for sale. Storage Engine: It is WiredTiger by default.
Private Network Address	The private IPv4 address and port of all primary and secondary mongod nodes in the TencentDB instance. A TencentDB instance can be accessed only over the private network. When using MongoDB Shell for access, you need to configure the private IP address and port. For detailed directions, see Connecting to TencentDB for MongoDB Instance.

Billing Mode	Instance billing mode, which is pay-as-you-go. For more information, see Billing Overview.
Used/Total	The used/total disk capacity of the instance. This parameter helps you quickly check the disk utilization of the current instance.
Oplog/Shard Info	You can click **View/Adjust** to view the disk capacity reserved for oplog or adjust it based on your business needs. For detailed directions, see Adjusting Oplog Capacity.
Project	Instance project. You can view the information of all instances in this project. You can also move the instance to another project as instructed in Specifying Project for Instance.
Operation	Select **Adjust Specification** to adjust the instance's memory and disk capacity as instructed in Adjusting Instance Specification. Select **Adjust Specification** > **Node Management** to manage the mongod and mongos nodes of the instance as instructed in Viewing Node Information. Select **More** > **Security Group** to change security group inbound rules. Select **More** > **Restart** to restart the instance as instructed in Restarting Instance. Select **More** > **Manage** to view the instance details. Select **More** > **Edit Tag** to edit the instance tag keys and values as instructed in Editing Instance Tag.

Viewing Instance Details

In the **Instance ID/Name** column of the target instance, click the instance ID to enter the **Instance Details** page.

Section	Parameter	Description
Basic Info	Instance Name	Custom instance name.
	Instance ID	The instance's unique ID.
	Instance Status	The instance's current status. The normal status is **Running**.
	Region	Instance region and AZ. You can click **Modify AZs** to switch to another AZ in the same region. For more information, see Changing Instance AZ.
	Project	The project to which the instance belongs. You can click **Switch to Another Project** to assign the instance to another project as instructed in Specifying Project for Instance.
Specification Info	Instance Type	You can set the instance cluster architecture type as **Replica Set** or **Sharded Cluster**. For more information, see System Architecture.

	Model Type	Product specification types include High IO 10G Type and Cloud Edition. For more information, see Specification Type.
	Version and Engine	The version and the storage engine of the instance. Version upgrade is supported. For detailed directions, see Version Upgrade.
	Mongod Node Specification	The specification of a single mongod node, including the CPU core quantity, memory, disk capacity, and node quantity. For the detailed specifications supported by replica set and sharded cluster instances, see Product Specifications.
	Mongos Node Specification	The specification of a single mongos node, including the CPU core quantity, memory, and node quantity. For the detailed specifications supported by replica set and sharded cluster instances, see Product Specifications.
	Disk Capacity	The total disk capacity of the instance.
Configuration Info	Billing Mode	Billing mode of the instance: Pay-as-you-go.
	Creation Time	Creation time of the instance.
	Maintenance Period	Instance maintenance period. To ensure the stability of the database, the backend system will periodically perform maintenance operations on the instance. You can click **Modify** to adjust the maintenance period as instructed in Setting Instance Maintenance Period. We recommend you schedule maintenance during off-peak hours.
	Auth-Free Access	You can view whether auth-free access to databases is enabled. If the status is **Not enabled yet**, you can click **Enable** to enable this feature as instructed in Accessing Instance Without Authentication.
	Tag	Tags associated with the instance. You can change them as instructed in Editing Instance Tag.
Network Configuration	Network	Instance VPC name. You can click **Switch Network** to switch the VPC and subnet as instructed in Switching Instance Network. If needed, you can also create a VPC as instructed in Creating VPC.
	Subnet	AZ-specific subnet in the instance VPC. A VPC allows for subnets in different AZs, which can communicate with each other over the private network by default. After you modify the instance AZ, we recommend that you also switch the subnet to reduce the access latency.
	Connection Type	The node type for database access. Access read/write primary node: Access the database through the primary node of the instance, which allows both read and write operations.

		Access read-only node only: Access the database exclusively through read-only nodes. If no read-only nodes are configured when creating an instance, they will not be displayed. Access secondary node only: Access the database exclusively through a secondary node. Access secondary node and read-only node: Access the database through a secondary node preferentially. If all secondary nodes are unavailable, the database will be accessed through a read-only node.
	Access address (connection string)	The URI encoded connection string of each connection type. You can directly copy a string to access the database as instructed in Connecting to TencentDB for MongoDB Instance.

More operations

Renaming an instance

1. In the Instance List, hover over the name of the target instance and click

on the right.

2. In the instance name input box, enter a new name, which must meet the following requirements:

Minimum of 1 character, maximum of 60 characters.

A combination of letters, digits, underscores, and hyphens.

A letter, digit, or special symbol is counted as one character.

Setting fields in the instance list

1. Click

in the top-right corner of the instance list.

2. On the **Display Settings** page, select the fields to be displayed.

3. Click **OK**, and you can see the reset fields in the instance list.

Exporting the instance list

You can click

in the top-right corner of the instance list to export the entire list.

Related APIs

API	Description
DescribeDBInstances	Queries the list of TencentDB instances
RenameInstance	Renames an instance

Adjusting Instance Specification

Last updated：2024-04-02 16:55:54

Overview

If the specification of your purchased TencentDB for MongoDB instance doesn't meet your current business requirements, whether it's below or above, you can easily adjust it according to your actual business conditions (at the initial stage, at the rapid development stage, during peak hours, or during off-peak hours), so as to better meet your needs such as full utilization of resources and real-time cost optimization.

Adjusting the specification

The memory and CPU cores of mongod and mongos nodes are in fixed combinations, and the disk capacity has a corresponding value range. For example, if the specification of a mongod node is 2-core 4 GB MEM, then the disk capacity can range from 100 to 500 GB.
Mongod replica secondary nodes: You have the option to choose between 3 nodes (one-primary-two-secondary), five (one-primary-four-secondary), or seven (one-primary-six-secondary) nodes. Currently, you cannot customize the number of secondary nodes.
The numbers of mongos nodes supported by single-AZ deployed and multi-AZ deployed instances are different. A single-AZ deployed instance can contain 3–32 nodes, while a multi-AZ deployed instance can contain 6–32 nodes.
You can select a specification based on your business conditions as described in Product Specifications first. Then, select an adjustment type and make adjustments based on the following table.

Specification Adjustment Type	Specification Description
Adjust the mongod node specification	You can adjust the memory, disk capacity, and oplog capacity of a mongod node.
Adjust the replica node quantity	You can add or remove secondary nodes in both replica set and sharded cluster. You have the option to choose between 3 nodes (one-primary-two-secondary), five (one-primary-four-secondary), or seven (one-primary-six-secondary) nodes. Currently, you cannot customize the number of secondary nodes.
Adjust the shard quantity	You can add or remove mongod shards in a sharded cluster.

Adjust the mongos node specification	You can adjust the CPU cores and memory of mongos nodes in a sharded cluster.
Adjust the mongos node quantity	You can add mongos nodes in a sharded cluster.
Add read-only nodes	You can add 0 to 5 read-only nodes.

Switching Instance Network

Last updated：2024-06-24 17:45:05

You can directly switch the network of a TencentDB for MongoDB instance in the console to adjust the network status promptly.

Overview

Tencent Cloud supports classic network and VPC as described in Overview, which are capable of offering a diversity of smooth services. On this basis, we provide more flexible services as shown below to help you manage network connectivity with ease.
Switch from classic network to VPC: A single TencentDB source instance can be switched from classic network to VPC.
Switch from VPC A to VPC B: A single TencentDB source instance can be switched from VPC A to VPC B.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support instance network switch.

Billing Overview

Switching the instance network doesn't incur additional fees.

Note

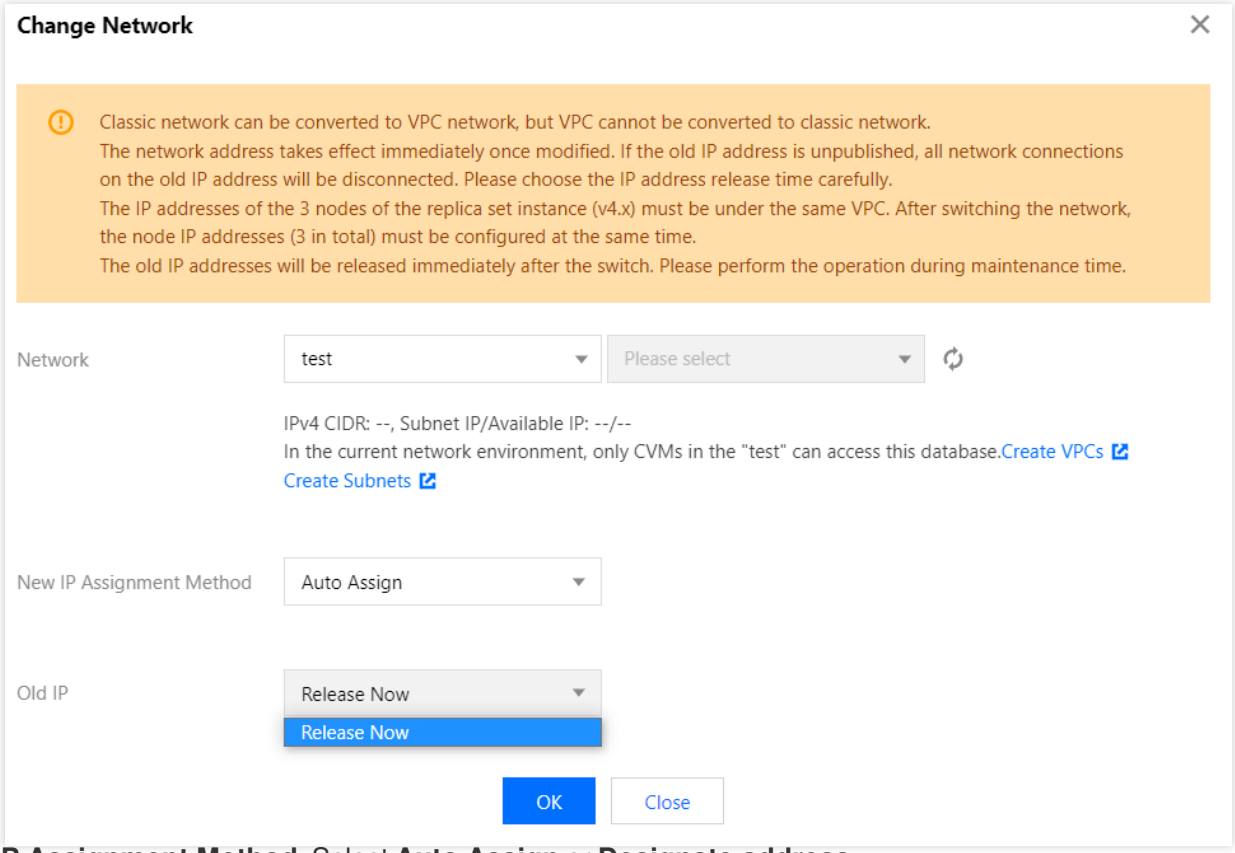
Switching the network may cause the change of instance's private IP. The original IP will become invalid after the reclamation time has elapsed. You need to change the instance IP on the client promptly.
The switch from classic network to VPC is irreversible. After the switch to a VPC, the TencentDB instance cannot communicate with Tencent Cloud services in another VPC or classic network.
The network switch of a primary instance doesn't apply to its associated read-only instances or disaster recovery instances, you need to manually switch the network for these instances.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB replica set or sharded cluster instance is in **Running** status.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page and click **Change Network** on the right of **Network**.
6. In the **Change Network** pop-up window, select a VPC and subnet in the current region in the drop-down list next to **Network**.
   If existing networks can't meet your requirements, you can click **Create VPCs** or **Create Subnets** to create a network and select it.



New IP Assignment Method: Select **Auto Assign** or **Designate address**.

Auto Assign: The system will automatically assign an available IP address based on the currently selected network environment.

Designate Address: You can enter a specific IP address in the **New IPv4 Address** input box. The specified IP address must be unoccupied and within the IP range of the specified VPC.

Note:

You can only select a VPC in the region of the instance.

We recommend that the VPC where the CVM instance resides should be selected; otherwise, the CVM instance will not be able to access TencentDB for MongoDB over the private network, unless a CCN is created between the two VPCs.

Old IP: For a replica set instance, the old IP address can be released immediately. For a sharded cluster instance, you need to select the release time of the old IP address in the drop-down list, which can be **Release Now**, **Release after 1 day**, **Release after 2 days**, **Release after 3 days**, or **Release after 7 days**. The IP address will be released after the retention time has elapsed.

Note:

When the delayed release is selected, there will be a transition period for IP address conversion, which is called "delay time". During the delay time, the old IP address can still be connected to, and the new IP address also takes effect. When the delay time is over, the old IP address will be reclaimed, the relevant cleanup task will be started to clear the configurations and records related to it, and all network connections to it will be closed immediately. You should choose the release time carefully.

7. After confirming the network switch, click **OK**. Return to the instance details page where you can view the new network of the instance.

Accessing Instance Without Authentication

Last updated：2024-10-10 16:46:37

Overview

Auth-free access allows you to access your TencentDB instance quickly and efficiently, but it also exposes your instance to security threats. We recommend that you enable this feature only in test or maintenance scenarios and disable it during routine business operations.

Version Description

Currently, TencentDB for MongoDB versions 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support auth-free access.

Note

Upgrading to the auth-free access version involves kernel upgrade and momentary disconnections.
Enabling auth-free access will require the instance to **restart**. It is advisable to enable this feature during off-peak hours.

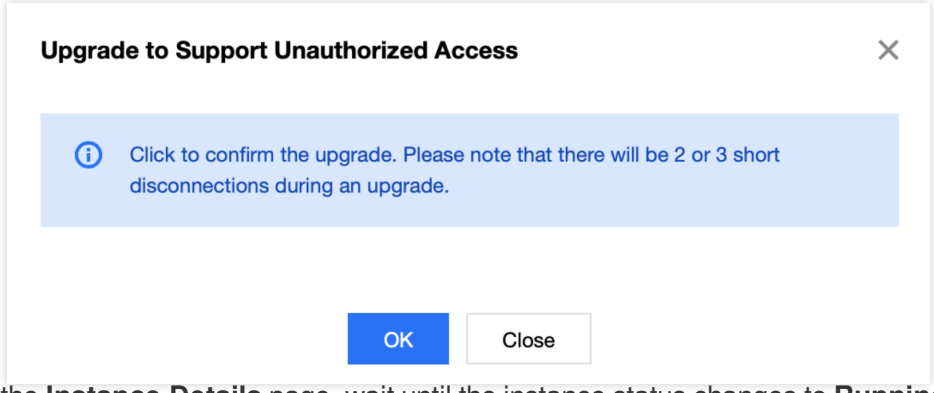
Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB instance is in **Running** status.
On the **Instance Details** page, the status of **Auth-Free Access** is **Not enabled yet**.

Enabling Auth-Free Access

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.

5. Click the instance ID to enter the **Instance Details** page.
6. On the **Instance Details** page, click **Enable** next to **Auth-Free Access**.
7. In the **Enable Auth-Free Access** pop-up window, confirm the impact of enabling this feature and click **OK**.



8. On the **Instance Details** page, wait until the instance status changes to **Running**. Once completed, Then, you can connect to the database using the private IPv4 address and port without the need for a username and password.

Disabling Auth-Free Access

On the **Instance Details** page, click **Disable** next to **Auth-Free Access** to disable this feature.

Relevant Operations

You can access TencentDB for MongoDB databases using MongoDB shell or drivers in various programming languages as instructed in Connecting to TencentDB for MongoDB Instance.

Changing Instance AZ

Last updated：2024-04-02 16:42:38

Overview

During regular maintenance, you can place your TencentDB for MongoDB and CVM instances in the same AZ to lower the network latency. TencentDB for MongoDB multi-AZ instances support free AZ switch, and single-AZ instances can be upgraded to multi-AZ instances.

Billing Overview

Changes to the AZ doesn't affect instance billing.

Note

Changes to the AZ may lead to primary-secondary switch and a momentary disconnection of about 10 seconds. It is advisable to make these changes during off-peak hours of your business.

Instructions

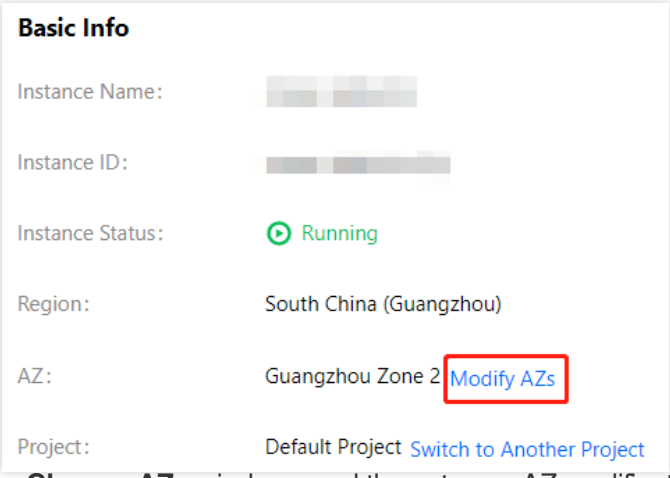
All attributes, specifications, and connection addresses of the instance will stay unchanged after AZ modification. However, the private IP of the database will change after network switch, so you need to reconnect to the instance.

Prerequisites

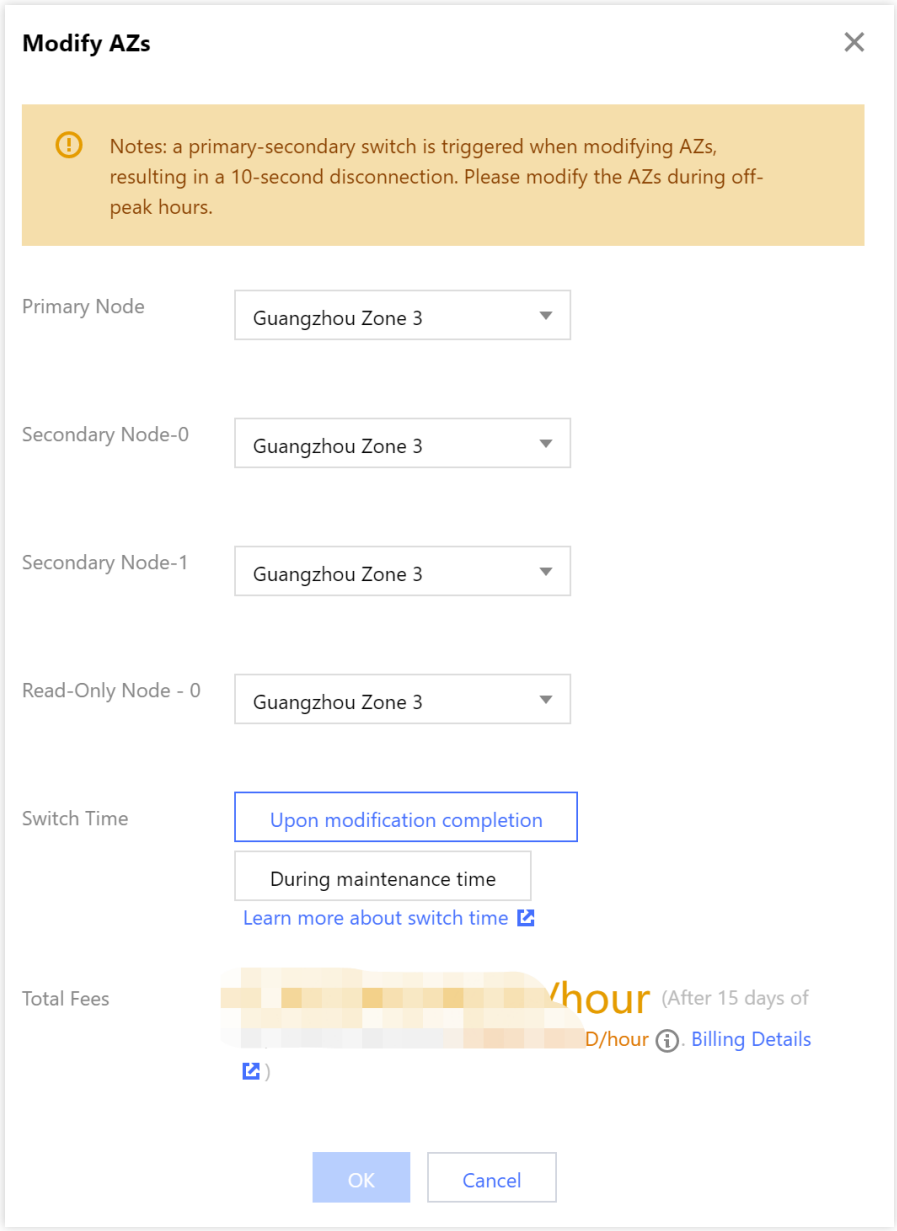
You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The instance is in **Running** status.
The target AZ and the current instance AZ are in the same region.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the instance ID to enter the **Instance Details** page.
6. On the **Instance Details** page, click **Change AZs next** to Region.



7. In the **Change AZs** window, read the notes on AZ modification. For a multi-AZ deployed instance, configure different AZs for the primary and secondary nodes respectively. For a single-AZ deployed instance, configure the same AZ for both the primary and secondary nodes.



Note:

Changes to the AZ may lead to primary-secondary switch and a momentary disconnection of about 10 seconds. It is advisable to make these changes during off-peak hours of your business.

8. Select the time for executing the AZ switch task next to **Switch Time**. You can click **Learn more about switch time** and change the instance maintenance period to the off-peak hours of your business. For detailed directions, see Setting Instance Maintenance Period.

Note:

If you have selected **During maintenance period**, do not select **Upon modification completion** to immediately execute the task before the configured maintenance period; otherwise, a program exception will occur. An initiated task cannot be stopped manually. To stop it, submit a ticket for application.

Upon modification completion: The task will be executed immediately after the configuration is completed.

During maintenance period: The task will be executed during the maintenance period.

9. Confirm **Total Fees**, click **OK** to enter the order page, check the order, click **Submit Order**, and make the payment to complete the operation.The instance status becomes **Changing AZ**. Wait for the task to be completed, and you can see that the AZ has been changed.

Subsequent operations

After changing the AZ, switch the VPC subnet to avoid a high access latency. For detailed directions, see Switching Instance Network.

Setting Instance Maintenance Period

Last updated：2024-10-10 16:48:06

TencentDB for MongoDB allows you to adjust the instance maintenance period in the console to meet the changes in your business requirements.

Overview

The maintenance period is crucial for ensuring the stability of your TencentDB for MongoDB instance. During this period, the backend system performs necessary maintenance operations on your instance. To minimize the potential impact on your business, we recommend you set an acceptable maintenance period, preferably during off-peak hours. This will help mitigate any disruptions to your business operations.
In addition, we also recommend you perform operations involving data migration, such as adjusting the memory specification or AZ of mongod or mongos nodes, during the maintenance period. Taking the database instance AZ change as an example, as the full and incremental data needs to be synced from the original AZ to the new AZ, data migration will be involved. After the AZ is changed, a momentary disconnection from the database may occur. When the task is initiated, the **Switch Time** can be selected as **During maintenance period**, so that the instance data migration will be started during the next **maintenance** period after the data sync is completed.
Note:
Before maintenance is carried out for TencentDB for MongoDB, notifications will be sent to the contacts configured in your Tencent Cloud account through SMS and email.

Version Description

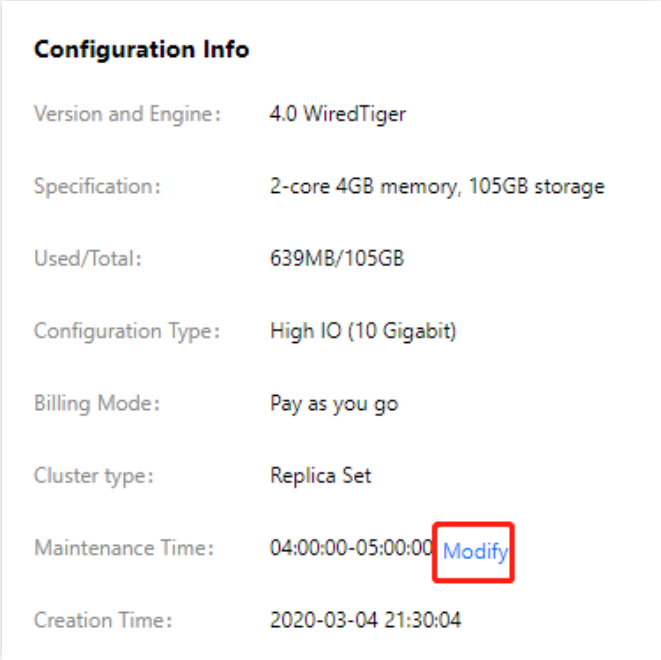
Currently, TencentDB for MongoDB versions 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support maintenance period configuration.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB replica set or sharded cluster instance is in **Running** status.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. On the **Instance Details** page, click **Change** on the right of **Maintenance Period**.
7. In the **Change Maintenance Period** window, set **Start Time** and **Duration**.



8. Click **OK**. You can view the newly set maintenance period on the **Instance Details** page.

Specifying Project for Instance

Last updated：2024-01-15 14:40:06

TencentDB for MongoDB allows you to assign an instance to a new project in the console to meet your requirements in ever-changing business scenarios.

Overview

A project is a set of applications or services that share resources. Each project is unique, with its own applications, services, and resources isolated from and unaffected by those in other projects.
You can specify an appropriate project for your database instances to facilitate collaboration. In this way, you can easily manage your instances globally and stay on top of the operational conditions of the entire project.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, and 5.0 support instance restart.

Billing Overview

Changing the instance project doesn't incur additional fees.

Note

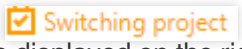
Assigning and reassigning TencentDB instances among projects will not affect the services provided by the instances.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
You have specified a project for the instance. The **Default Project** is used by default.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. In the **Basic Info** section, click **Switch to Another Project** on the right of **Project**.
7. On the **Assign to Project** page, select the target project.
8. Click **OK**. In the **Basic Info** section,



will be displayed on the right of **Instance Status**.
9. Wait for the task to be completed. On the right of **Project**, you can see the project to which the instance is reassigned.

You can filter instances by **Project** in the instance list to view the running status of each instance in the entire project.

API

API Name	Description
AssignProject	Specifies the project to which an instance belongs

Editing Instance Tag

Last updated：2024-10-10 16:49:52

TencentDB for MongoDB allows you to edit instance tags in the console for easier instance management.

Overview

A tag consists of a tag key and value. It can be used to tag TencentDB for MongoDB instances. If you have multiple types of resources under your Tencent Cloud account which are correlated in many ways, and your resources are growing and becoming increasingly difficult to manage, you can use tags to group and categorize resources that serve the same purpose or are associated with each other. In this way, when performing daily Ops or troubleshooting, you can quickly search for resources and perform batch operations for more efficient Ops.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support tag management.

Billing Overview

Tag management is a free service provided by Tencent Cloud for your Tencent Cloud account. You can simply go to the console to use this service.

Note

A tag consists of 1 tag key and 1 tag value (tagKey:tagValue).
Up to 50 tags can be bound to an instance.
For each instance, a tag key can correspond to only one tag value.

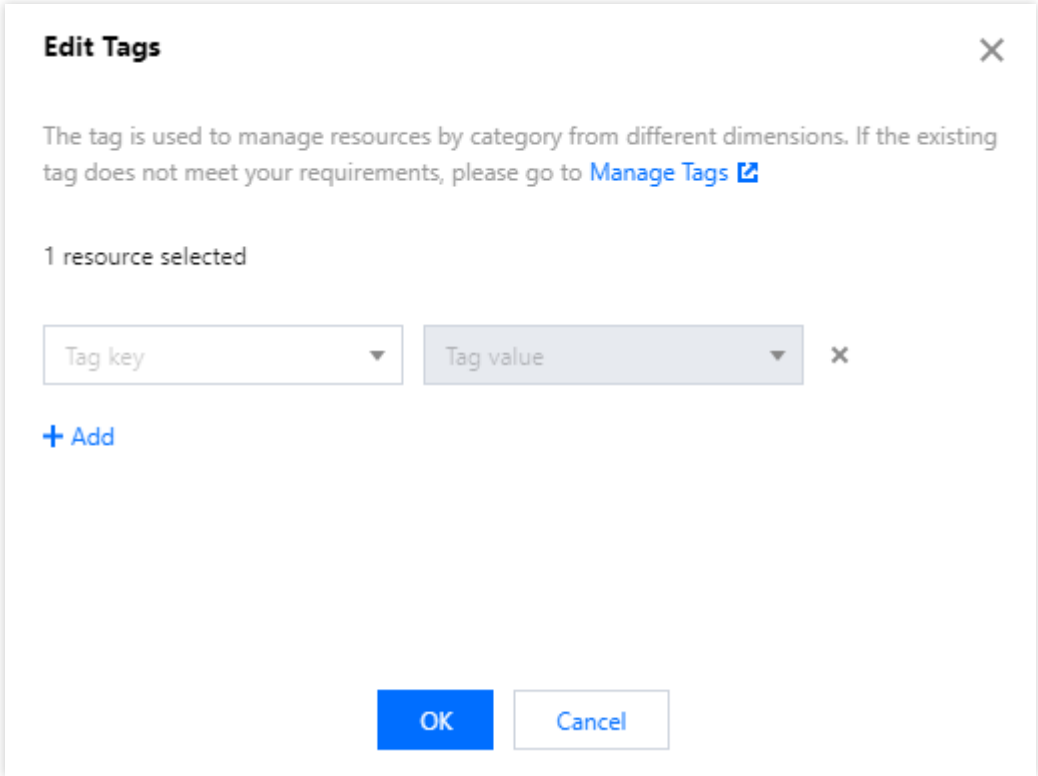
Prerequisite

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Enter the **Edit Tag** page in any of the following ways:
In the **Operation** column of the target instance, select **More** > **Edit Tag**.
Click the target instance ID and click

on the right of **Tag** in the **Configuration** section on the **Instance Details** page.
6. On the **Edit Tag** page, select an appropriate tag key from the **Tag Key** drop-down list and select the tag value from the **Tag Value** drop-down list.



7. (Optional) If existing tags don't meet your business requirements, perform the following operations:
7.1 In the top-right corner of the current page, click **Manage Tags**.
7.2 On the **Manage Tags** page, click **Create Tag**.
7.3 On the **Create Tag** page, carefully read the notes on tag configuration.
7.4 Set a new tag key in the **Tag Key** input box and enter the tag value in the **Tag Value** input box. The requirements for the tag key are as follows:

- It can contain 1–63 characters.
- It can contain letters and digits.
- It can contain the following special symbols: plus sign, equal sign, underscore, hyphen, dot, colon, slash, @, parentheses, and brackets.

7.5 Click **OK** to complete the creation.
7.6 Go back to the **Edit Tag** page of the database instance. Click **Reload** in the **Tag Key** drop-down list, select the created tag key, and select the tag value.
8. Click **OK**.

References

For more information on tag management, see Tag.

Restarting Instance

Last updated：2024-10-10 16:50:43

If the number of connections to an instance reaches the threshold or the instance has performance problems, you need to restart it manually. This document describes how to restart a replica set or sharded cluster instance.

Overview

When the system becomes completely unavailable due to a high load, you can restart it to resume node operations. Due to the architecture of the TencentDB for MongoDB instance, instance restart divides into mongos restart and mongod restart.
mongos is a routing service configured for MongoDB sharding. It processes query requests from the application layer and determines the location of data in a sharded cluster
mongod is the primary daemon process for the MongoDB system. It handles data requests, manages data access, and performs background management operations.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support instance restart.
The architecture of MongoDB **4.0** replica set instance is simplified, where the mongos component is removed; therefore, instance restart doesn't involve mongos restart.

Note

During restart, there may be one or two momentary disconnections for around 10s each. We recommend that you configure an automatic reconnection feature for your application.
You cannot cancel an ongoing restart operation; therefore, proceed with caution.

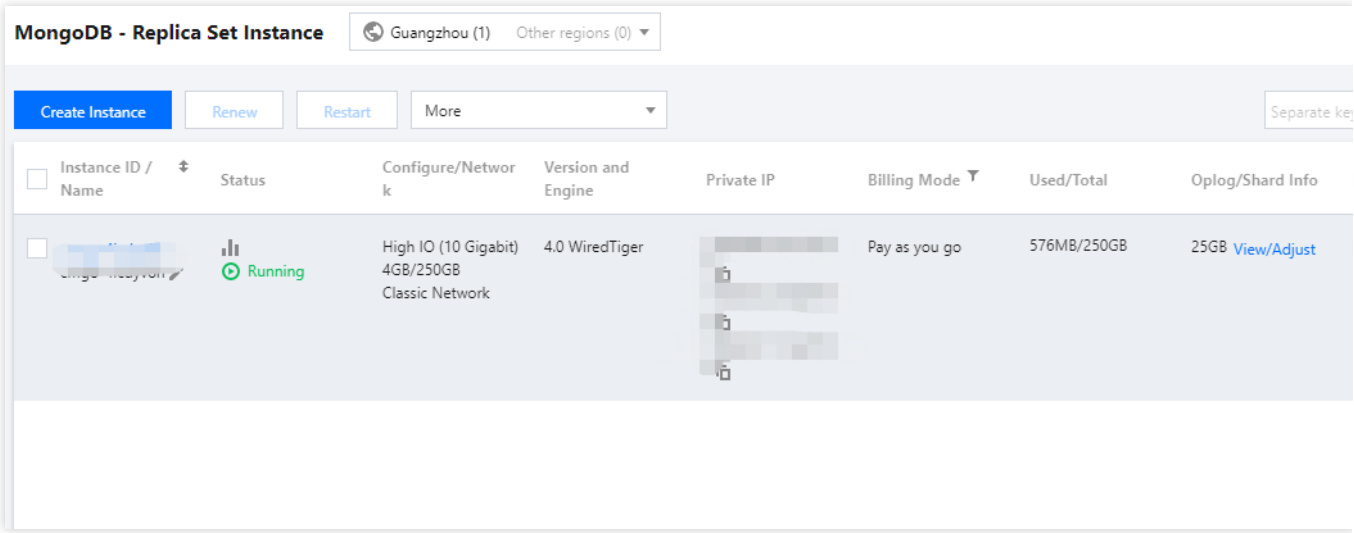
Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB instance is in **Running** status.

Directions

Restarting one instance

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. On the row of the target instance, click **More** > **Restart** in the **Operation** column.



6. In the **Restart MongoDB** window, click **View Details** to confirm the information of the instance to be restarted.
7. Select the components to be restarted and click **OK**.
8. In the instance list, you can see that the instance enters the **Restarting** status. Wait for the task to be completed.

Restarting multiple instances

1. In the instance list, select the instances to be restarted.
2. Above the instance list, click **Restart**.
3. In the **Restart MongoDB** window, click **View Details** to confirm the information of all the instances to be restarted.
4. Select the components to be restarted and click **OK**.

Terminating Instance

Last updated：2024-10-10 16:52:49

If you no longer need a TencentDB for MongoDB instance, you can directly terminate and return it in the console.

Overview

You can terminate an instance if you no longer need it. The terminated instance will be put into the recycle bin. For instances in the recycle bin, you can restore or release them as needed based on different scenarios.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support instance termination.

Billing Overview

When an instance is returned and its status has changed to **Isolated**, it will no longer generate fees.
For instances that met the 5-day no-questions-asked refund policy, the payment will be returned to your Tencent Cloud account.
For normal instances, the payment will be returned to your Tencent Cloud account by the proportion of the cash and gift cards paid for the purchase.
After a pay-as-you-go instance is returned, it will be moved to the recycle bin and retained there for three days. During the retention period, the instance cannot be accessed, but it can be restored after top-up.

Note

After an instance is completely terminated, its data will be cleared and cannot be recovered. You need to back up the data in advance.

Note

When an instance is terminated, its read-only instances will not be terminated simultaneously.

After an instance is terminated, it will be moved to the recycle bin. During the retention period, the instance cannot be accessed. To use the instance again, you can restore it from the recycle bin. For detailed directions, see Recycle Bin. When the instance is terminated, its IP resources will be released simultaneously. The DR instance will be disconnected from the synchronous connection and automatically upgraded to the master instance.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB instance is in **Running** status.

Directions

Pay-as-you-go instance

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Operation** column of the target instance, select **More** > **Terminate**.
6. In the pop-up window, read the prompt message carefully, confirm the instance to be terminated, and click **OK**.

Recycle Bin

Terminated instances will be put into the recycle bin and can be restored during the retention period. For more information, see Recycle Bin.

Adjusting Oplog Capacity

Last updated：2024-10-10 16:58:18

Overview

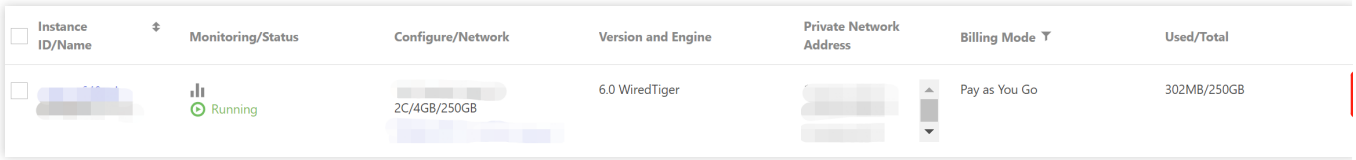
Oplog is an important component in MongoDB used to log the operations in the database. The oplog capacity should be at least 10% of the node capacity, as the oplog records all database operations, including insertions, updates, and deletions. If the oplog capacity is too small, the oplog may be overwritten, affecting the rollback feature of MongoDB. When you purchase an instance, the oplog capacity is 10% of the instance capacity by default and can be expanded to 90% of the instance capacity as needed; however, it cannot be shrunk currently.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
If your instance is pay-as-you-go, make sure that your Tencent Cloud account balance is sufficient.
The instance and its associated instances are in **Running** status and are not executing any tasks.

Directions

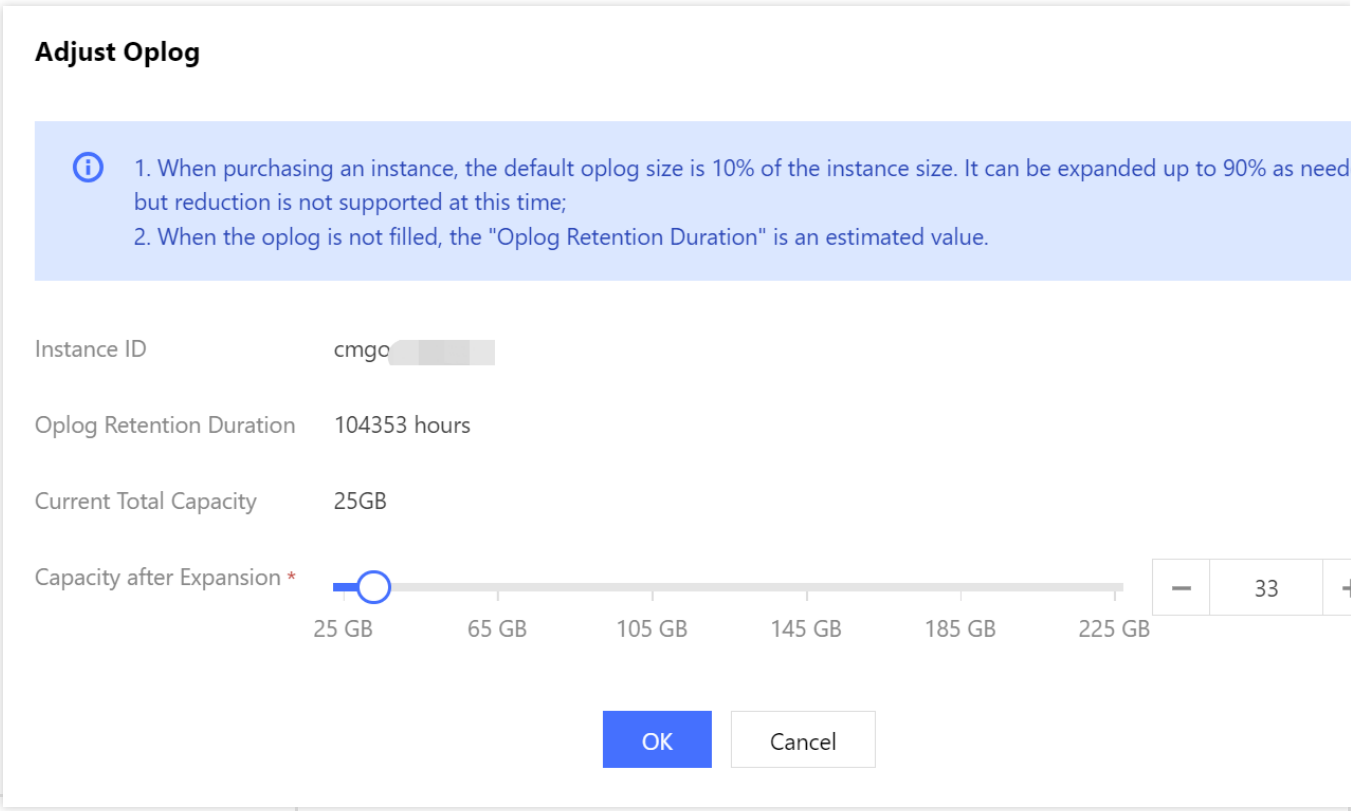
1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Shard Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Oplog Info** column, click **View/Adjust**.



6. In the **Adjust Oplog** pop-up window, confirm the instance information and evaluate the target oplog capacity based on the current capacity.

Note:

The Oplog capacity should be at least 10% of the node capacity. When the size of the Oplog file reaches its maximum capacity, MongoDB starts overwriting previous operation records from the beginning of the file. If the Oplog is too small, it can lead to rapid data overwrite, posing Ops risks.



Interface Parameters	Description
Oplog retention duration	The time it takes for the Oplog storage to be filled. If the Oplog is not full, the retention duration is an estimate.
Current total capacity	The storage capacity of the Oplog for each Mongod node.
Capacity After Scale-out	Use the slider to set the Oplog capacity after scale-out.

7. Click **OK**, and then in the left sidebar, choose **Task Management**. Wait for the task to complete the scale-out.

Node Management
Node Overview

Last updated：2024-06-26 16:34:04

The replica set architecture of TencentDB for MongoDB achieves high availability and read/write separation by deploying multiple types of nodes. Each replica set instance consists of one primary node, one or multiple secondary nodes, and one hidden node.
The sharded cluster architecture implements the horizontal capacity expansion of data based on the replica set architecture by combining multiple replica sets, each of which is a shard.
Each node is as described below:

Node	Feature	Description
Primary node	It is responsible for handling read/write operations.	There can be only one primary node in each replica set instance.
Secondary node	It replicates the data of the primary node by periodically polling the oplogs of the primary node with data consistency guaranteed. When the original primary node fails, a new primary node will be elected from multiple secondary nodes to ensure the high availability.	When a client connects to a secondary node, it has read-only access and cannot write data. A secondary node can be expanded as described in Adding Secondary Node. A secondary node can be promoted to primary node as described in Promoting Secondary Node to Primary Node.
Hidden node	A secondary node will be designated as the hidden node by default for each newly purchased instance. It serves as an invisible replica node to back up data. When a secondary or a read-only node fails, this hidden node and the faulty secondary node can be switched to new secondary nodes, thereby achieving high availability.	There can be only one hidden node in each replica set instance. It is not possible to delete a secondary node that has been designated as a hidden node. A hidden node is not included in the "Primary Node's Replica List" and will not be elected as the primary node. However, it can participate in the voting process to elect the primary node.
Read-only node	If the read-only replica feature is enabled, the system will set one or more secondary nodes as read-only nodes. Read-only nodes are primarily used in scenarios with a large volume of read requests. They synchronize data from the primary or secondary nodes through the operation log. The system	Read-only nodes do not participate in the election of the primary node. There can be multiple read-only nodes in each replica set instance. For more information, see Adding Read-Only Node.

	automatically routes read requests to the read-only nodes to reduce the access pressure on the primary node.	

Viewing Node Information

Last updated：2024-04-02 16:14:18

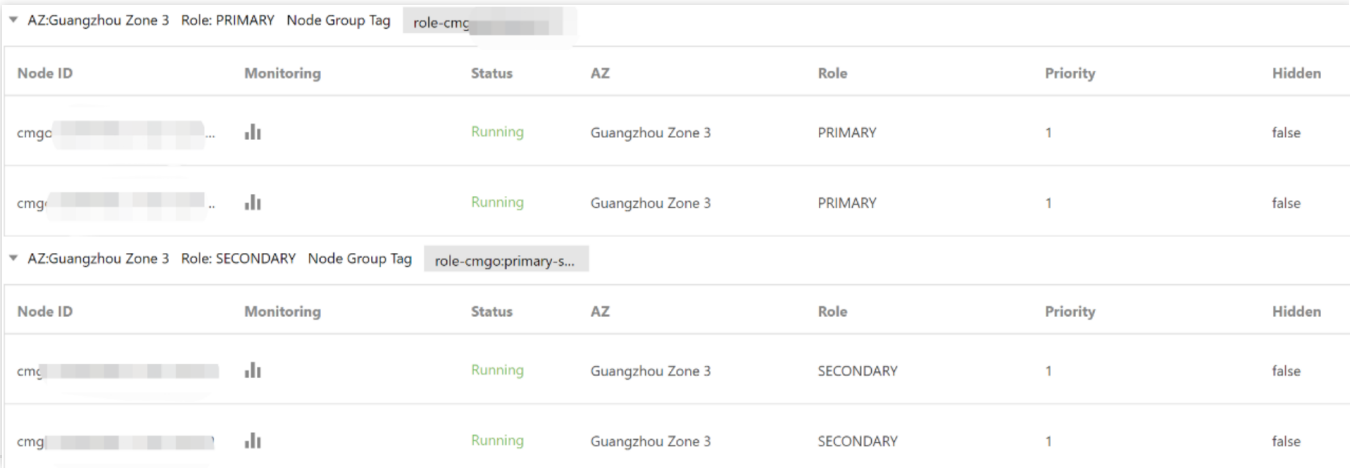
Overview

TencentDB for MongoDB allows you to view the instance node information, including node ID, role, running status, and used capacity. In addition, it supports node management operations, such as adjusting node specification, promoting secondary node to primary node, enabling read-only replica, and configuring primary/secondary failover. You can use node management to efficiently manage instance nodes and locate node exceptions.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. View the mongod and mongos node information.

Mongod Node



Parameter	Description
Node ID	Mongod node ID.
Monitoring	Click

	, and you can view the monitoring metrics of the node on the monitoring panel on the right. For more information, see Viewing Monitoring Data.
Status	Status of the current node.
AZ	AZ of the current node.
Role	Role of the current node. `PRIMARY` : Primary node. `SECONDARY` : Secondary node. `READONLY` : Read-only node.
Priority	The priority of a node for being elected as the primary node. The greater the value, the higher the priority.
Hidden	Whether the node is hidden. Default value: `false` .
Primary/Secondary Delay (second)	The latency in syncing data from the primary node to the secondary node in seconds.
Used Disk Space	The storage usage of the node disk.

Mongos Node

Parameter	Description
Node ID	Mongos node ID.
Monitoring	Click , and you can view the monitoring metrics of the node on the monitoring panel on the right. For more information, see Viewing Monitoring Data.
Status	Status of the node.
AZ	AZ of the mongos node.

Adjusting Mongod Node Specification

Last updated：2024-05-06 11:52:34

Overview

If your purchased TencentDB for MongoDB instance is over-provisioned or under-provisioned, your business needs cannot be best met, and you can quickly adjust its specifications as needed (at the start, during rapid development, or during peak/off-peak hours), so you can get the most out of your resources and reduce unnecessary costs in real time. You can change the computing specification and disk capacity for mongod nodes. In order to choose the specifications that best suit your needs, we recommend that you first check TencentDB Product Specifications before making a choice.

Billing Overview

The instance will be billed by the new specifications after its specifications are changed. Make sure that your Tencent Cloud account balance is sufficient to cover the updated billing. For more information, see Specification Adjustment Billing.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
If your instance is pay-as-you-go, make sure that your Tencent Cloud account balance is sufficient.
The instance and its associated instances are in **Running** status and and there are currently no tasks being executed.

How Expansion Works and Impacts

If there are sufficient physical machine resources where the node resides, the storage space will be expanded in this node. This expansion process does not require any migration or switching between nodes, ensuring uninterrupted connections..
If there are insufficient physical machine resources where the node resides, expanding the storage space will require cross-node data migration. After the expansion task is initiated, the system will create nodes and sync data as required by new specifications, and perform the switch as scheduled. As a momentary disconnection of about 10s will occur during the switch, we recommend that you include the reconnection mechanism in your business code and

perform the adjustment during off-peak hours. The higher volume of data, the longer time it takes to complete the configuration adjustment.

Note：

After the scaling operation is executed, the system will re-trigger a new automatic backup task.

Notes

During adjustment, there may be one or two momentary disconnections for about 10s each. We recommend that you configure an automatic reconnection feature for your application.

During adjustment, if you set the write concern level to write majority, there may be a short request delay; therefore, it is advisable to set an appropriate business timeout period.

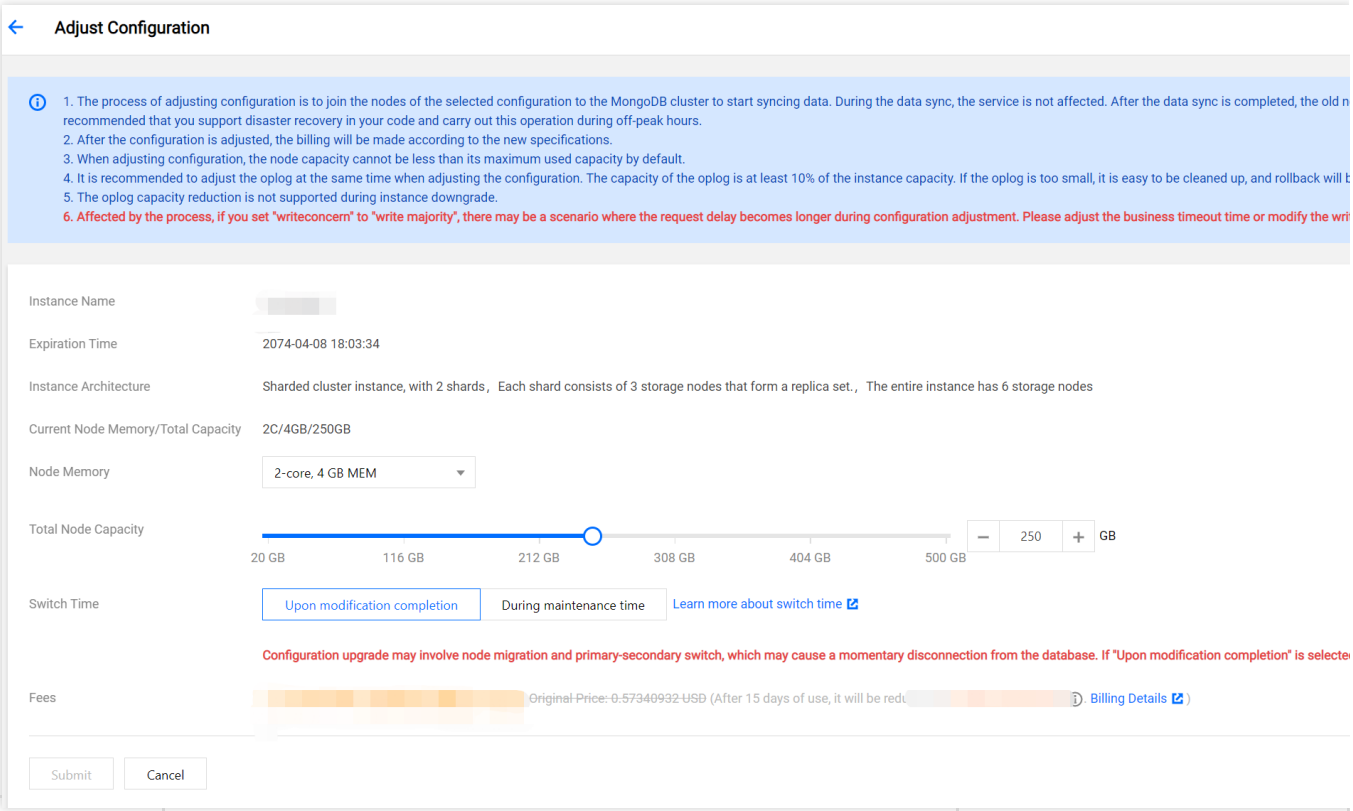
The name, private network address, and port of the instance remain unchanged after specification adjustment.

A started specification adjustment task cannot be canceled.

After the instance is upgraded, we recommend that you adjust the Oplog capacity to avoid having a small Oplog size. This is important to prevent data overwriting, which could potentially impact the rollback feature of MongoDB. For more information, see Adjusting Oplog Capacity.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Operation** column of the target instance, select **Adjust Configuration** in the drop-down list.
6. On the **Adjust Configuration** page, you can adjust the node memory and total node capacity.



Parameter	Description	Example
Instance Name	The name of the instance that requires specification adjustment.	test-4dot2-XXX
Instance Architecture	The description of the instance's cluster architecture. For more information, see System Architecture.	A sharded cluster instance has two shards, and each shard consists of three storage nodes to form a replica set. Hence, the entire instance has six storage nodes in total.
Node Memory/Total Capacity	The mongod node memory and total capacity per node in the current instance. For a sharded cluster, the total node capacity is the node capacity per shard. For how to query the number of CPU cores of an instance, see the mongod specifications in Product Specifications.	4 GB/100 GB MEM
Node Memory	Select the new memory per mongod node in the drop-down list. For how to choose a specification, see the mongod specifications in Product Specifications.	8 GB MEM
Total Node Capacity	Adjust the total disk capacity per mongod node on the slider, which is the same as the total capacity of the current node by	500 GB MEM

	default. For how to choose a specification, see the mongod specifications in Product Specifications.	
Switch Time	If you select **Upon modification completion**, the instance specification adjustment task will be executed immediately. Instance memory and disk adjustment may involve node migration or primary-secondary switch. As the switch time is uncontrollable, disconnection or restart may occur. If you select **During maintenance period**, the task will be executed during the maintenance period. For more information, see Setting Instance Maintenance Period.	During maintenance period
Fees	Pay-as-you-go: Hourly unit price after instance specification adjustment. You can click **Billing Details** to view the billable items and billing formula and confirm the fees.	177.991 USD

7. After confirming that everything is correct, click **Submit**.

Related APIs

API Name	Description
ModifyDBInstanceSpec	Adjusts the specifications of a TencentDB instance

Adding Secondary Node

Last updated：2024-04-02 15:54:16

Overview

All secondary nodes of an instance contribute to the system's high availability. When the primary node fails, each secondary node may be elected as the new primary node to execute data write requests. Therefore, the more the replicas, the higher the availability. In scenarios with a high number of concurrent requests with more reads and less writes, if read/write separation is enabled, you can add secondary nodes to improve the read performance and greatly relieve the read pressure on the primary node.
A TencentDB for MongoDB cluster can have three (one-primary-two-secondary), five (one-primary-four-secondary), or seven (one-primary-six-secondary) nodes in total. You can add secondary nodes appropriately based on the surge in the concurrent requests to your business and remove secondary nodes when the business load drops. This helps you better utilize resources and reduce unnecessary costs in real time.

Billing

The instance will be billed by the new specifications after its specifications are changed. Make sure that your Tencent Cloud account balance is sufficient to cover the updated billing. For more information, see Specification Adjustment Billing.

Notes

After new nodes are added to the cluster, data sync will start without affecting the business.
Be sure to plan for disaster recovery. We recommend you initiate a specification adjustment task during the maintenance period. For more information, see Setting Instance Maintenance Period.
Do not adjust the node quantity and node computing/storage specifications at the same time.
After the node quantity is adjusted, billing will start based on the new specification.
The name, private network address, and port of the instance remain unchanged after node quantity adjustment.
A started specification adjustment task cannot be canceled.

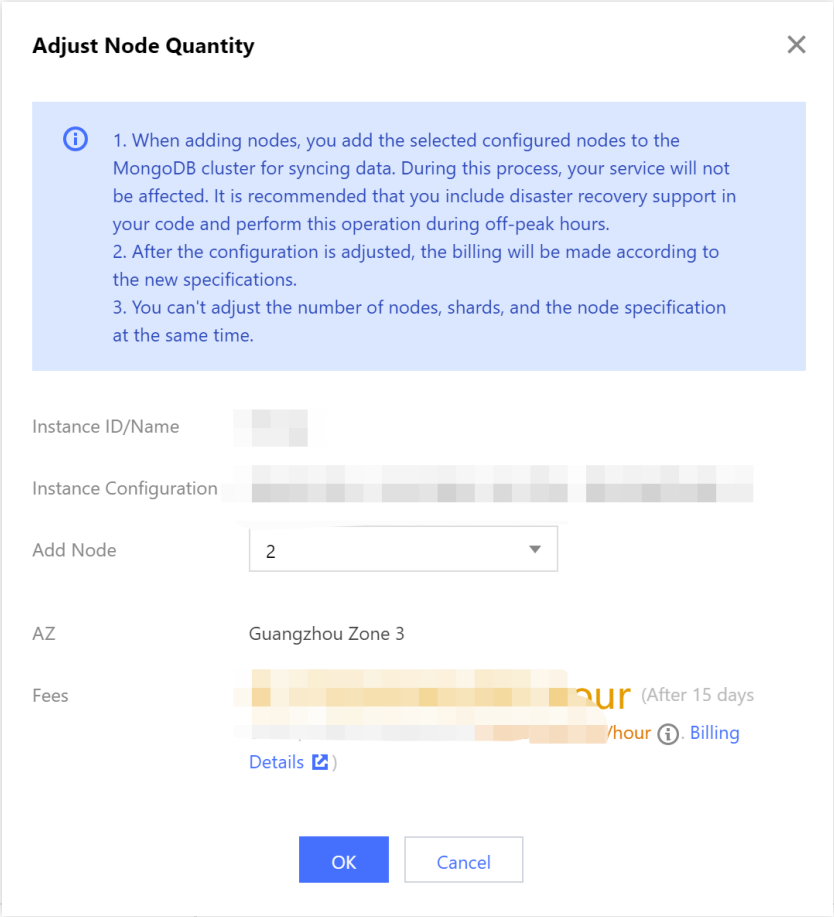
Prerequisites

You have created a TencentDB for MongoDB instance.

If your instance is pay-as-you-go, make sure that your Tencent Cloud account balance is sufficient.
The instance and its associated instances are in **Running** status and and there are currently no tasks being executed.

Adding a secondary node (replica set)

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance**.
3. Above the instance list on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Mongod Node** tab on the **Node Management** tab, click **Add Secondary Node**.
7. In the Adjust Node Quantity pop-up window, read the notes on node quantity adjustment, and proceed to confirm and configure the parameters as detailed below:



Parameter	Description
Instance ID/Name	The name of the instance that requires node quantity adjustment.
Instance	Check the current specification of the instance, including the CPU core quantity, memory,

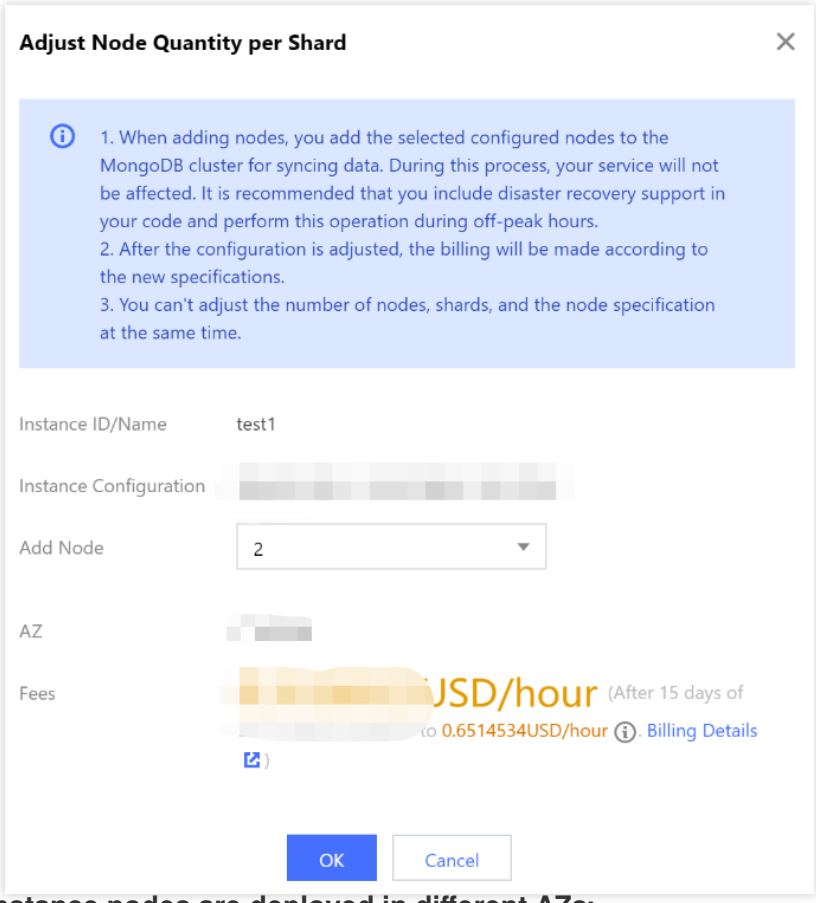
Specification	disk capacity, and node quantity. The node quantity includes all primary and secondary nodes. You should determine the number of nodes to be added based on the current specification.
Add Node	Select the number of secondary nodes to be added from the drop-down list.
Deployment AZ	This parameter will be displayed if the instance nodes are in the same AZ. It indicates the AZ where all instance nodes are deployed.
Secondary Node-n	This parameter will be displayed if the instance nodes are in different AZs. It indicates the AZs of different nodes and ranges from 0 to 6. Select the AZs for the new secondary nodes from the drop-down list.
Fees	Pay-as-you-go: Hourly unit price after instance speciation adjustment. You can click Billing Details to view the billable items and billing formula and confirm the fees.For more information, see Specification Adjustment Billing.

8. Confirm the fees and click **OK**.
9. On the left sidebar, select **Task Management**, and you can view the ongoing task. Wait until **Task Progress** becomes **100%** and **Task Status** becomes **Completed**.

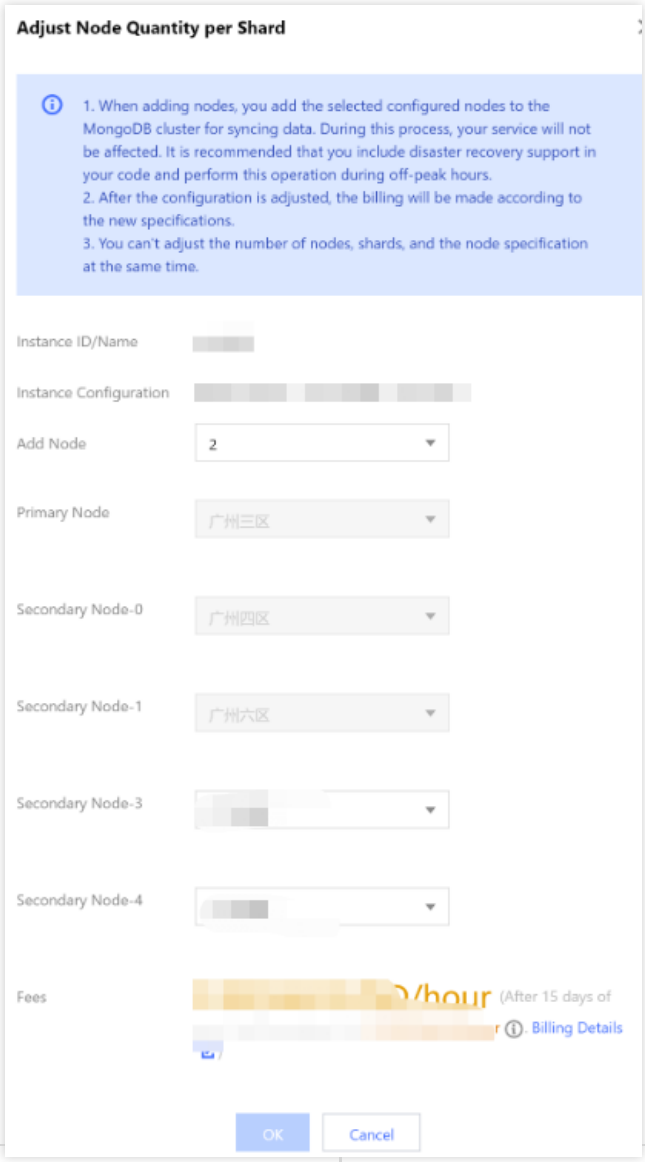
Increasing the node quantity per shard (for sharded cluster instance)

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Sharded Cluster Instance**.
3. Above the instance list on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Mongod Node** tab on the **Node Management** tab, click **Add Secondary Node**.
7. In the Adjust Node Quantity per Shard pop-up window, read the notes on node quantity adjustment, and proceed to confirm and configure the parameters as detailed below:

The instance nodes are deployed in the same AZ:



The instance nodes are deployed in different AZs:



Parameter	Description
Instance ID/Name	Confirm the name of the instance that requires node quantity per shard adjustment.
Instance Specification	Check the current specification of the instance, including the CPU core quantity, memory, disk capacity, and node quantity. The node quantity includes all primary and secondary nodes. The nodes are evenly distributed to shards; for example, if there are two shards and eight nodes, each shard will have four nodes. You should determine the number of nodes to be added based on the current specification.
Add Node	Select the number of secondary nodes to be added per shard from the drop-down list.
Deployment AZ	This parameter will be displayed if the instance nodes are in the same AZ. It indicates the AZ where all instance nodes are deployed.

Secondary Node-n	This parameter will be displayed if the instance shard nodes are in different AZs. It indicates the AZs of different nodes and ranges from 0 to 6. Select the AZs for the new secondary nodes from the drop-down list.
Fees	Pay-as-you-go: Hourly unit price after instance speciation adjustment. You can click Billing Details to view the billable items and billing formula and confirm the fees.

8. Conform the fees and click **OK**.
9. On the left sidebar, select **Task Management**, and you can view the ongoing task. Wait until **Task Progress** becomes **100%** and **Task Status** becomes **Completed**.

Related APIs

API	Description
ModifyDBInstanceSpec	Adjust the specification of a TencentDB instance

Deleting Secondary Node

Last updated：2024-04-02 15:44:10

Overview

Deleting secondary nodes can reduce the high availability of a cluster. When the business load is low, you can remove secondary nodes appropriately to avoid wasting resources.

Instructions

Deleting secondary nodes can reduce the high availability of a cluster. Therefore, proceed with caution. Make sure that after some secondary nodes are deleted, the cluster still has three (one-primary-two-secondary), five (one-primary-four-secondary), or seven (one-primary-six-secondary) nodes in total.
A hidden node cannot be deleted, as when a secondary node fails, the system will automatically switch it with the hidden node to guarantee the cluster's high availability.
The IP address of a deleted secondary node won't be retained, so connections to the secondary node will be closed.

Prerequisites

You have created a TencentDB for MongoDB instance.
If your instance is pay-as-you-go, make sure that your Tencent Cloud account balance is sufficient.
The instance and its associated instances are in **Running** status and and there are currently no tasks being executed.

Deleting a secondary node (replica set)

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance**.
3. Above the instance list on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Mongod Node** tab on the **Node Management** tab, select the target secondary node in the **Instance List** and select **Operation** > **Delete Secondary Node**.

Note:

In the node list, if the value of a node in the **Hidden** column is **true**, the node is hidden and cannot be deleted.

7. In the **Delete Secondary Node** pop-up window, read the notes on node quantity adjustment and confirm the instance name.

Parameter	Description
Instance ID/Name	The name of the instance that requires node quantity adjustment.
Instance Specification	Check the current specification of the instance, including the CPU core quantity, memory, disk capacity, and node quantity. The node quantity includes all primary and secondary nodes. The nodes are evenly distributed to shards; for example, if there are two shards and eight nodes, each shard will have four nodes. You should determine the number of nodes to be deleted based on the current specification.
Secondary Node Information	Confirm the information of the secondary nodes to be deleted, including the node ID, AZ, role, and tags.
Specification Change Fees	Fees after specification adjustment. In pay-as-you-go billing mode, fees are charged hourly by the new specification in three billing tiers.
Compare	You can compare the specifications and maximum numbers of connections before and after the specification adjustment of mongod secondary nodes in order to assess whether the new specification meets your needs.

Related APIs

API	Description
ModifyDBInstanceSpec	Adjust the specification of a TencentDB instance

Adding Read-Only Node

Last updated：2024-06-25 11:19:53

Overview

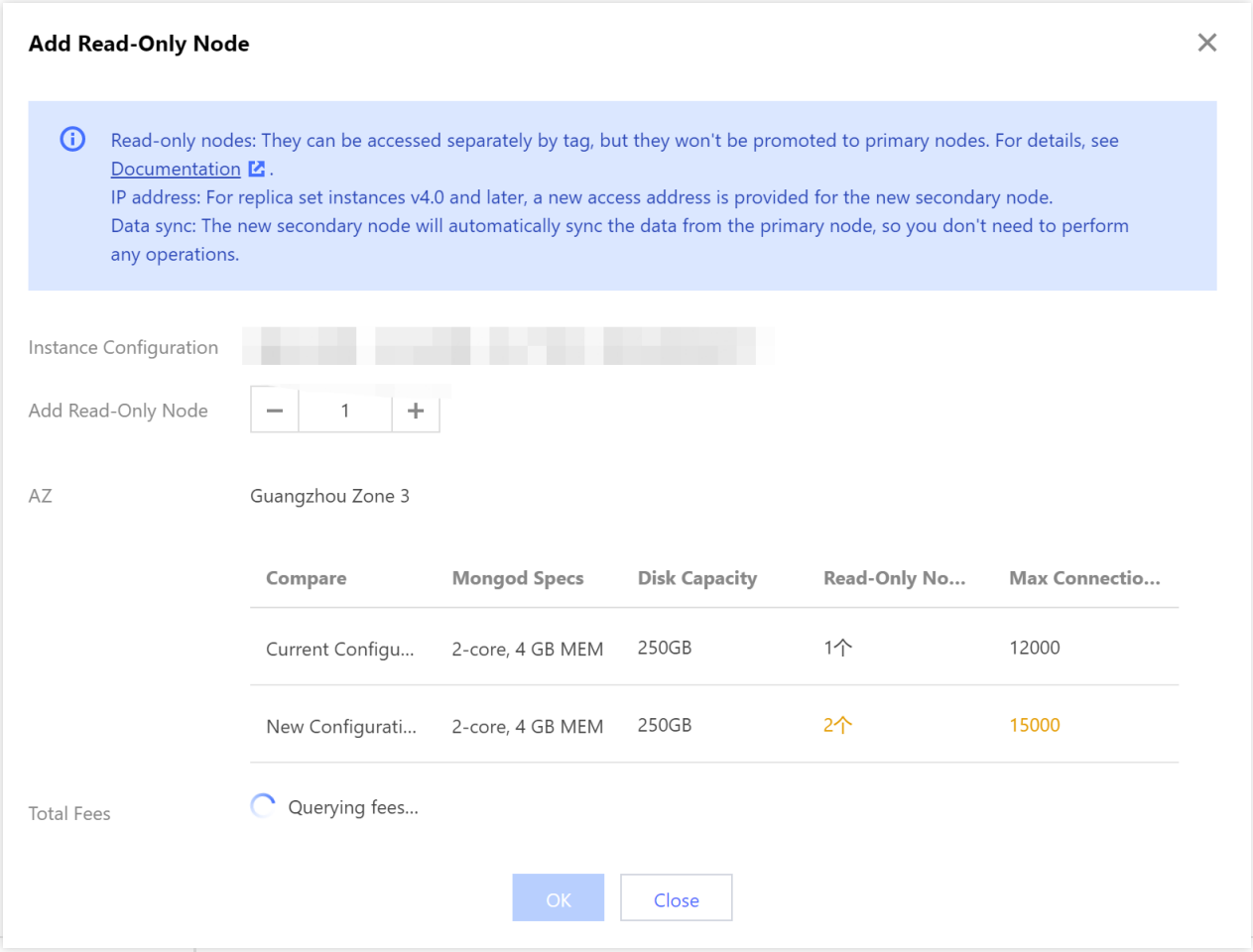
When your business has a massive number of read requests, the primary and secondary database nodes may struggle to handle them, causing high latency, slow response, and severely dropped throughput. To solve this problem, TencentDB for MongoDB provides read-only nodes with an independent connection address, allowing data to be synced from a primary or secondary node with the lowest latency through oplog. You can create one or multiple read-only nodes to implement read/write separation and relieve the access pressure on the primary and secondary nodes.
Two or more nodes can implement load balancing for read requests and guarantee a high availability; that is, when a read-only node fails, the system will automatically switch it with a hidden node. If automatic switch isn't performed, you can switch the node on the **Node Management** tab, and the connection address to the node will remain unchanged. You can directly get the connection string in the **Network** section on the **Instance Details** page.
If a read-only node isn't in the candidate list of the primary node, it won't be elected as the primary node or participate in the election.

Version Description

TencentDB for MongoDB versions 4.0, 4.2, 4.4, 5.0 and 6.0 support adding read-only nodes, while v3.6 doesn't.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Instance**. The directions for replica set instances and sharded instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Mongod Node** tab, click **Add Read-only Node**.



Parameter	Description
Instance Specification	Check the current specification of the instance, including the CPU core quantity, memory, disk capacity, total node quantity, and read-only node quantity to evaluate the number of read-only nodes to be added.
Add Read-Only Node	The number of new read-only nodes. Value range: 0–5.
AZ	The AZ where all read-only nodes are deployed. This parameter will be displayed if the instance nodes are in the same AZ.
Compare	Compare the specifications before and after adding the read-only node to evaluate whether the new specification meets your needs. The specification of a replica set instance includes mongod specification, disk capacity, number of read-only nodes, and maximum number of connections. The specification of a sharded cluster instance includes number of shards, mongod specification, disk capacity, number of read-only nodes, and maximum number of connections.
Total Fees	Pay-as-you-go: Hourly unit price after instance speciation adjustment. You can click **Billing Details** to view the billable items and billing formula and confirm the fees.

7. Click **OK**.
8. On the left sidebar, select **Task Management**. In the task list, find the instance by ID or name and wait until the **Task Status** for adding the read-only node is marked as **Completed**.

Adjusting Shard Quantity

Last updated：2024-04-02 15:33:32

Overview

You can adjust the shard quantity after purchasing a sharded cluster instance to adapt to your changing business scenarios.

Billing

The instance will be billed by the new specifications after its specifications are changed. Make sure that your Tencent Cloud account balance is sufficient to cover the updated billing. For more information, see Specification Adjustment Billing.

Note

After new nodes are added to the cluster, data sync will start without affecting the business.
Do not adjust the node quantity and node computing/storage specifications at the same time.
After the node quantity is adjusted, billing will start based on the new specification.
The name, private network address, and port of the instance remain unchanged after node quantity adjustment.
A started specification adjustment task cannot be canceled.

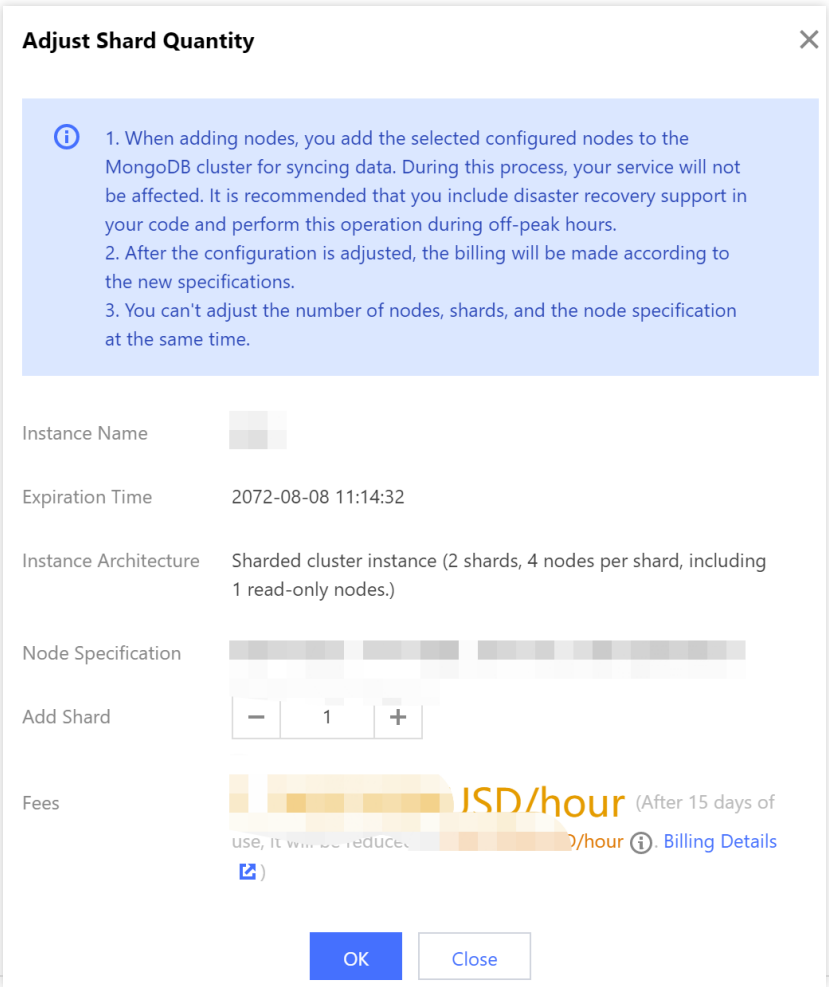
Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
If your instance is pay-as-you-go, make sure that your Tencent Cloud account balance is sufficient.
The sharded cluster instance and its associated instances are in **Running** status and there are currently no tasks being executed.

Directions

1. Log in to the TencentDB for MongoDB console.

2. In the **MongoDB** drop-down list on the left sidebar, select **Sharded Cluster Instance**.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Mongod Node** tab on the **Node Management** tab, click **Adjust Shard Quantity**.
7. In the **Adjust Shard Quantity** window, read the notes on shard quantity adjustment, specify the new number of shards as describe below, and confirm the fees.



Parameter	Description	Example
Instance Name	The name of the instance that requires node quantity adjustment.	test-4dot2-XXX
Expiration Time	The expiration time of the instance (for billing purpose).	2022-04-24 19:23:43
Instance Architecture	The description of the instance's cluster architecture. For more information, see System Architecture.	Sharded cluster instance with two shards and five storage nodes per shard
Node	The shard node specification information of the current sharded	2 cores, 4 GB memory,

Specification	cluster instance, including the number of CPU cores, memory, storage capacity, and node quantity.	250 GB storage, 10 nodes in total
Add Shard	Select the number of shards to be added to the instance. Value range: [1,36].	3
Fees	Pay-as-you-go: Hourly unit price after instance speciation adjustment. You can click **Billing Details** to view the billable items and billing formula and confirm the fees. For billing details after the configuration adjustment, see Configuration Adjustment Billing.	0.99 USD/hour

8. After confirming that everything is correct, click **OK**.

Related APIs

API Name	Description
ModifyDBInstanceSpec	Adjust the specification of a TencentDB instance

Adjusting Mongos Node Specification

Last updated：2024-04-02 15:19:28

Overview

Upgrading the computing specification of mongos nodes can increase the maximum number of connections to the database. You can adjust the mongos node specification appropriately based on the actual conditions of your business access.

Notes

Upgrading the CPU performance and memory capacity of mongos nodes may involve cross-node data migration and cause a momentary disconnection. Therefore, before performing this operation, make sure that your business has an automatic reconnection mechanism. We recommend that you complete this operation within the maintenance time during off-peak hours.

Version requirements

TencentDB for MongoDB 4.0, 4.2 and 4.4 support adjusting the mongos node specification.

Prerequisites

Instance type: Sharded cluster instance.
Instance status: Running.
The CPU performance and memory capacity of the mongos nodes are insufficient and need to be upgraded.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Sharded Cluster Instance**.
3. Above the **Instance List** on the right, select the region.
4. In the **Instance List**, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.

6. On the **Node Management** tab, click the **Mongos Node** tab.
7. On the **Mongos Node** tab, click **Change Mongos Specification**. In the pop-up window, configure the new specification.

Parameter	Description
Instance ID/Name	The unique ID and name of the instance.
AZ	The AZ where the instance resides.
Mongos Quantity	The current number of mongos nodes.
Mongos Specs	Select the new mongos node specification in the drop-down list, which can be 1-core 2 GB MEM, 2-core 4 GB MEM, 4-core 8 GB MEM, 8-core 16 GB MEM, or 16-core 32 GB MEM.
Switch Time	If you select Upon modification completion, the instance specification adjustment task will be executed immediately. Instance memory and capacity adjustment may involve node migration or primary-secondary switch. As the switch time point is uncontrollable, disconnection or restart may occur. If you select During maintenance period, the task will be executed during the maintenance period. For more information, see Setting Instance Maintenance Period.
Specification Change Fees	Fees after specification adjustment. In pay-as-you-go billing mode, fees are charged hourly by the new specification in three billing tiers.
Compare	You can compare the maximum number of connections before and after the mongos specification adjustment to evaluate whether the new specification meets your needs.

8. Click **OK**.

Adding Mongos Node

Last updated：2024-04-02 15:13:12

Overview

You can add more mongos nodes to increase the maximum number of connections to the database instance.

Version requirements

Currently, sharded cluster instances on v4.0, v4.2 and v4.4 support adding mongos nodes.

Instructions

After you add a mongos node, the system will automatically bind an IP address to it and enable the connection string for mongos access. Then, you can directly copy the connection string in the **Network** section on the **Instance Details** page.

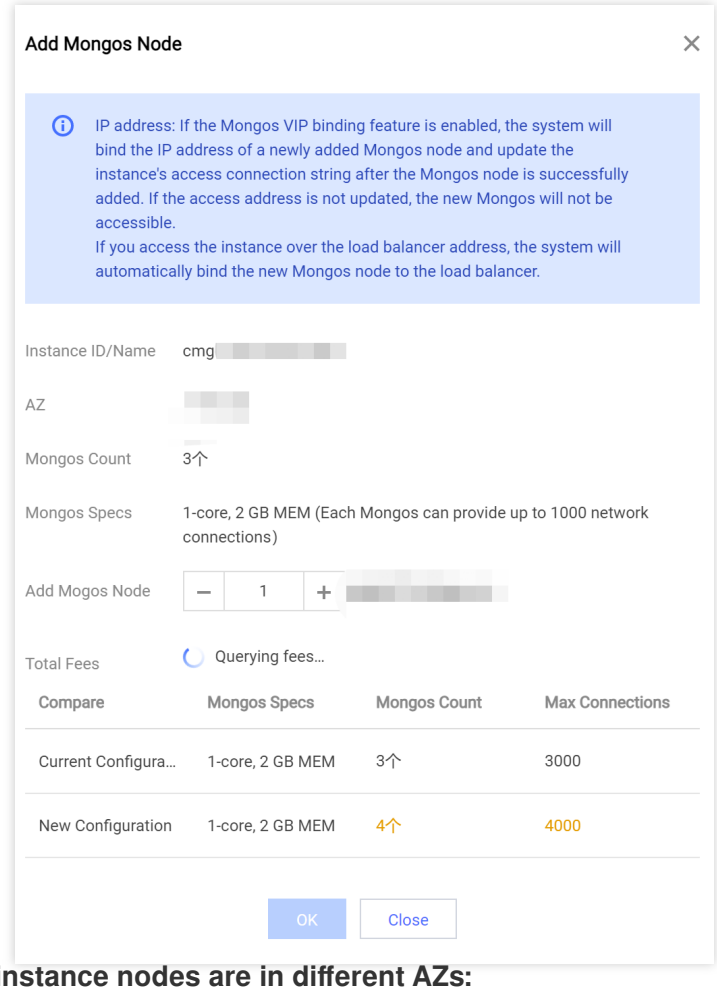
Prerequisites

Instance type: Sharded cluster instance.
Instance status: Running.
The CPU performance and memory capacity of the mongos nodes are insufficient and need to be upgraded.

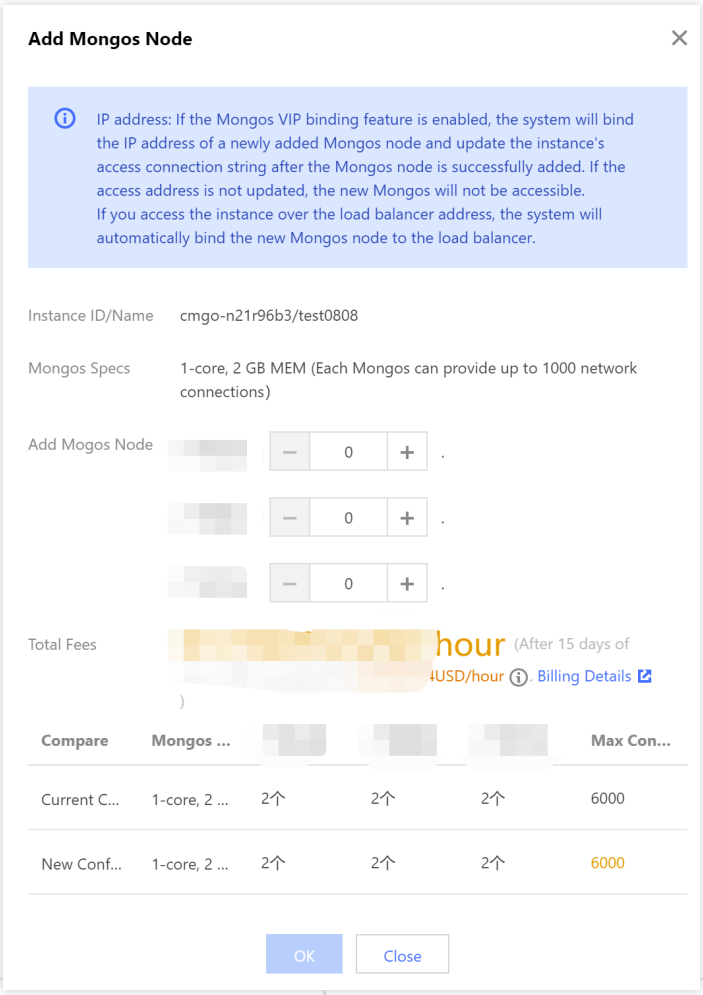
Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Sharded Cluster Instance**.
3. Above the **Instance List** on the right, select the region.
4. In the **Instance List**, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Node Management** tab, click the **Mongos Node** tab.
7. On the **Mongos Node** tab, click **Add Mongos Node**

The instance nodes are in the same AZ:



The instance nodes are in different AZs:



Parameter	Description
Instance ID/Name	The unique ID and name of the instance.
AZ	The AZ where the instance resides. This parameter will be displayed if the instance nodes are in the same AZ.
Mongos Quantity	The current number of mongos nodes configured for the instance. This parameter will be displayed if the instance nodes are in the same AZ.
Mongos Specs	Specification of mongos nodes, including the number of CPU cores, memory, and maximum number of connections.
Add Mongos Node	Select the number of mongos nodes to be added. An instance can have up to 48 mongos nodes.
Total Fees	Fees after specification adjustment. In pay-as-you-go billing mode, fees are charged hourly by the new specification in three billing tiers.
Compare	You can compare the specification, number of nodes in the AZ, and maximum numbers of connections before and after mongos nodes are added to assess

	whether the new specification meets your needs.

8. After confirming that everything is correct, click **OK**.

Enabling Mongos Access Address

Last updated：2024-04-02 15:11:11

Overview

After enabling the mongos access address of a sharded cluster instance, you can access the instance using this address. On the Instance Details page, you can see the mongos access connection string (for private network access).

Instructions

Under the current VIP of the instance, different vports will be bound to different mongos nodes.
After a mongos node fails, the system will bind its vPort to a new mongos process. The VIP and vPort will remain unchanged.
Enabling mongos access address won't affect the original CLB access address.

Version description

TencentDB for MongoDB v4.0 and later support enabling the mongos access address.

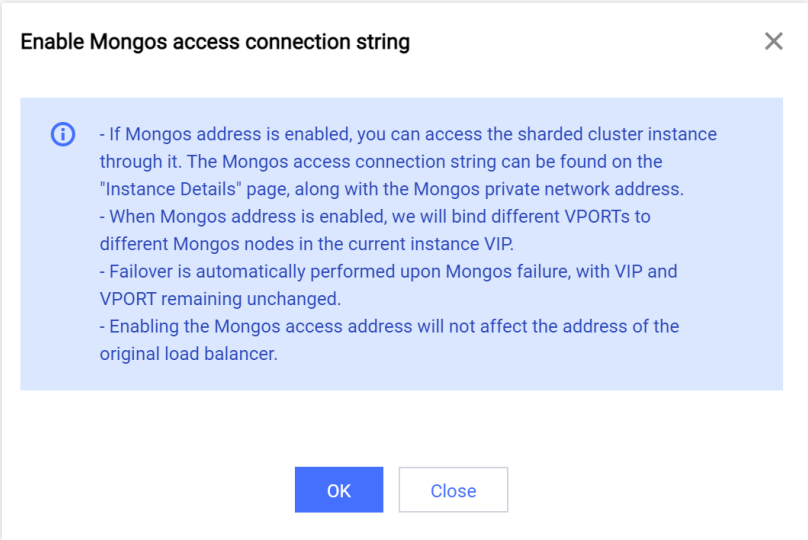
Prerequisites

Instance type: Sharded cluster instance.
Instance status: Running.

Directions

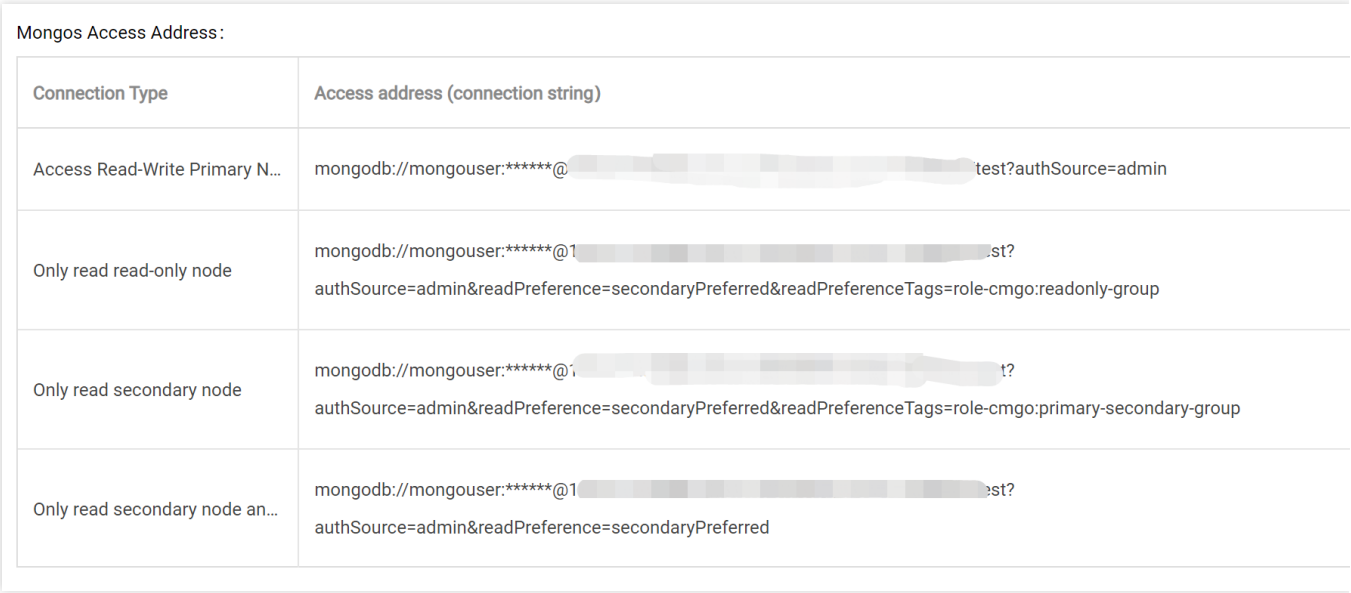
1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Sharded Cluster Instance**.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.
6. On the **Node Management** tab, click the **Mongos Node** tab.

7. On the **Mongos Node** tab, click **Enabling Mongos Access Address**.
8. In the pop-up window, confirm the impact of enabling the access connection string and click **OK**.



9. On the left sidebar, select **Task Management**. In the task list, find the instance with the **Task Type** being **Enabling Node Access Address** by ID and wait until the **Task Status** is marked as **Completed**.
10. On the left sidebar, select **Sharded Cluster Instance**. In the instance list, find the instance with the access address enabled, click its ID to enter the **Instance Details** page. In **Access Address** in the **Network Configuration** section, you can view the mongos access address. Hover over the connection string of the access address and click

to copy it for mongos node access.



Promoting Secondary Node to Primary Node

Last updated：2024-04-02 15:05:25

Overview

A TencentDB for MongoDB replica set instance can have one primary node and multiple secondary nodes. If you find that the primary node is abnormal, you can actively promote a secondary node to primary node to ensure normal business operations. In a sharded cluster instance, all shard nodes are grouped into one primary node group, and all secondary nodes are grouped into multiple secondary node groups. If some shard nodes in the primary node group are abnormal, you can actively promote all nodes in a secondary node group to the primary node group.

Version description

TencentDB for MongoDB v3.6 and later support promoting secondary node to primary node.

Notes

When a node is promoted to primary node, existing TCP connections to the database will be closed. Therefore, before performing this operation, make sure that your business has an automatic reconnection mechanism; otherwise, you need to manually reconnect to the database.

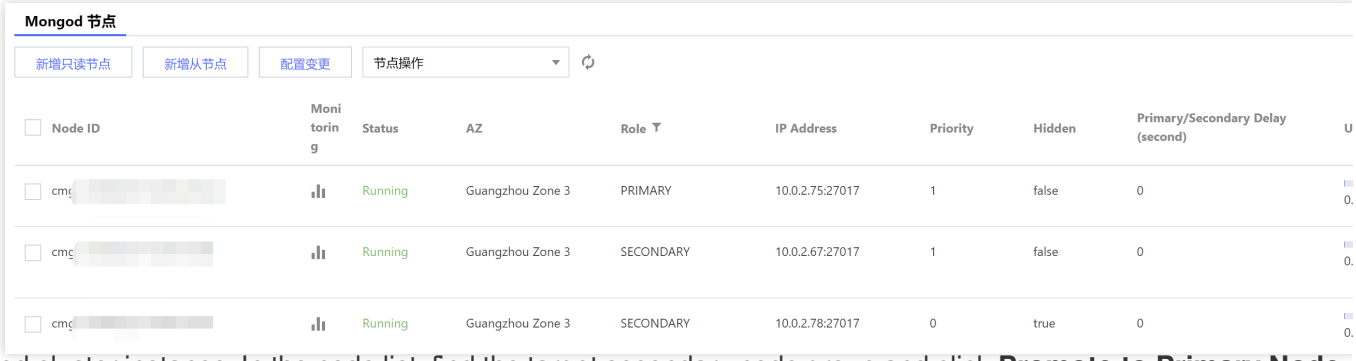
Prerequisites

The instance is in **Running** status.

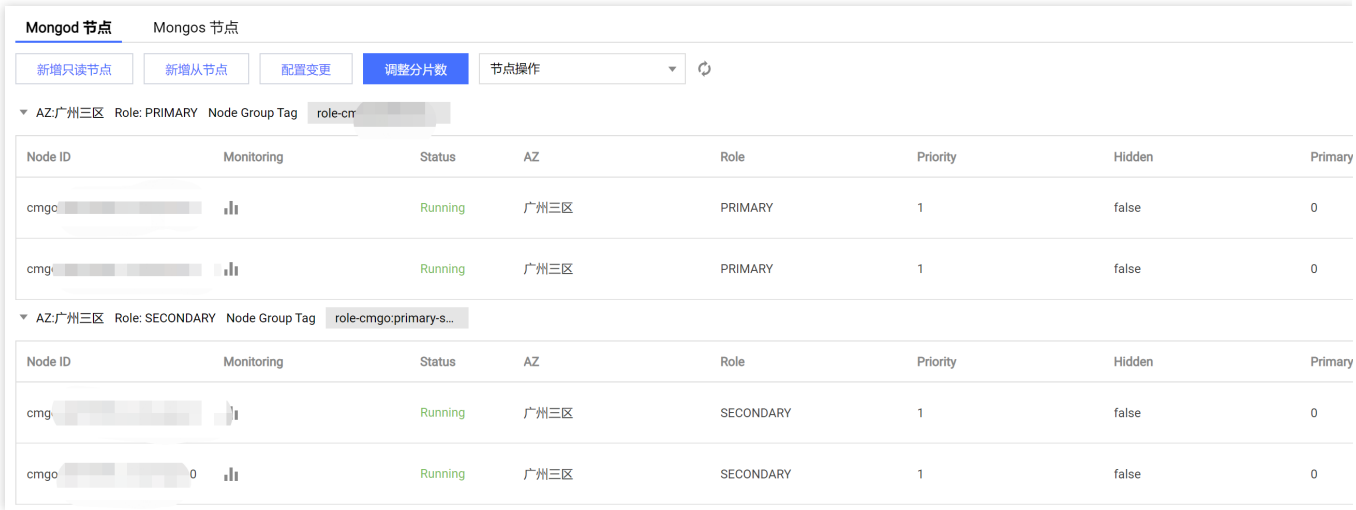
Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the **Instance ID** to enter the **Instance Details** page and click the **Node Management** tab.

6. On the **Mongod Node** tab on the **Node Management** tab, find the target secondary node in the node list.
Replica set instance: In the node list, find the target secondary node and click **Promote to Primary Node** in the **Operation** column.



Sharded cluster instance: In the node list, find the target secondary node group and click **Promote to Primary Node** in the top-right corner.



7. In the **Promote to Primary Node** window, select **Confirm the risk of the promotion to primary node** and click **OK**.

Note:

When a node is promoted to primary node, existing TCP connections to the database will be closed. Therefore, before performing this operation, make sure that your business has an automatic reconnection mechanism; otherwise, you need to manually reconnect to the database.

8. Return to the **Instance Details** page, you can see that **Instance Status** is **Promoting to Primary Node**. After this status disappears, the task is completed. On the **Node Management** tab, you can see that the **Role** of the original secondary node becomes **PRIMARY**.

Restarting a Node

Last updated：2024-10-10 11:47:15

Overview

In daily Ops, restarting nodes helps free up memory or clear caches, optimizing resource usage. When a node experiences a failure, restarting it can assist in restoring service.

Must-Knows

Restarting a node will disconnect existing connections, so ensure that your application has a reconnection mechanism in place.
Restarting the primary node will trigger a primary-secondary switch, and there is a risk of data loss if unsynchronized data has not been replicated to the secondary node. Assess the risks carefully.

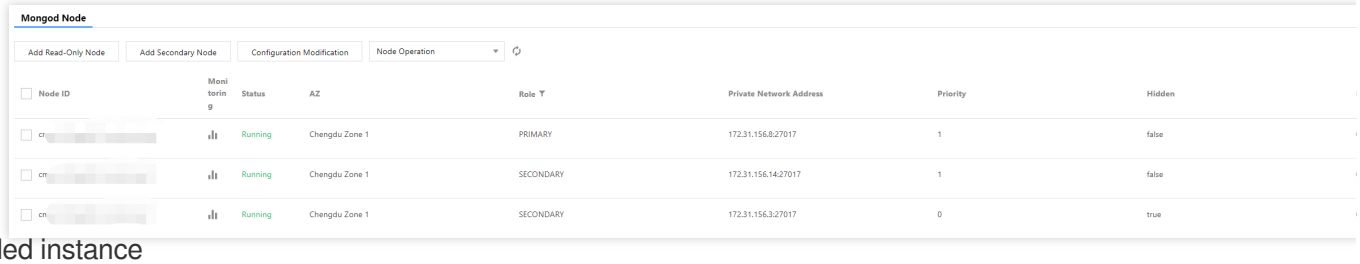
Prerequisite

Instance status should be Running.

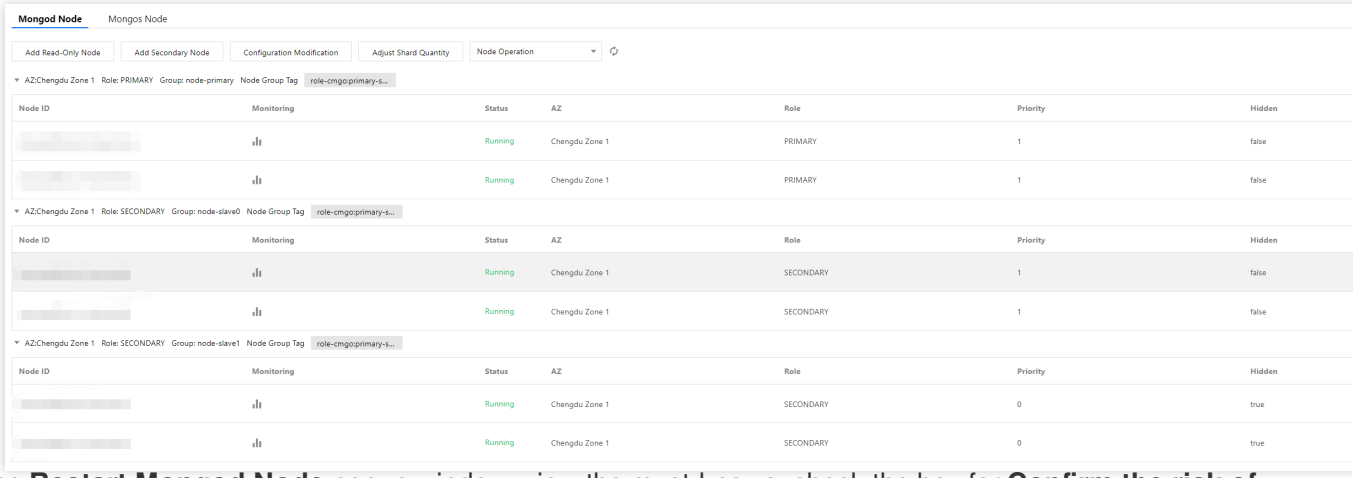
Restarting Mongod Node

1. Log in to the MongoDB Console.
2. In the left sidebar, choose **Replica Set Instance** or **Shard Instance** from the **MongoDB** dropdown.
3. At the top of the **Instance List** page on the right, select the region.
4. In the instance list, find the target instance for which you want to view the node.
5. Click the instance ID to enter the **Instance Details** page, and click the **Node Management** tab.
6. On the **Node Management** page, under the **Mongod Node** tab, locate the node you want to restart in the node list.
7. In the **Operation** column, click **Restart**.

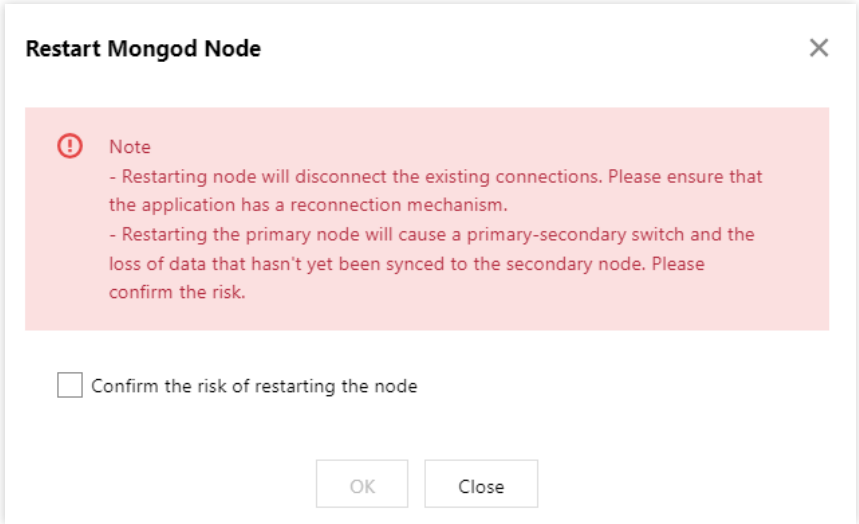
Replica set



Sharded instance



8. In the **Restart Mongod Node** popup window, view the must-knows, check the box for **Confirm the risk of restarting the node**, and click **OK**.



9. Wait for the task to be completed.

Restarting ConfigServer Node

1. Log in to the MongoDB Console.
2. In the left sidebar, select **Replica Set Instance** or **Shard Instance** from the **MongoDB** dropdown.
3. At the top of the **Instance List** page on the right, select the region.

4. In the instance list, find the target instance for which you want to view the node.
5. Click the instance ID to enter the **Instance Details** page, and click the **Node Management** tab.
6. On the **Node Management** page, under the **ConfigServer Node** tab, locate the node you want to restart in the node list.
7. In the **Operation** column, click **Restart**.

Version Upgrade

Last updated：2024-05-08 10:46:06

Overview

TencentDB for MongoDB allows you to upgrade both the major and minor database versions. You can enjoy more features by upgrading the major version from v3.6 to v4.0 or from v4.0 to v4.2.

Version Description

TencentDB for MongoDB supports upgrades from older versions to higher versions, but does not support cross-version upgrades. You can only upgrade the MongoDB from v3.6 to v4.0, v4.0 to v4.2, v4.2 to v4.4, or v4.4 to v5.0. For the feature differences between versions, see Storage Engine and Version.
You can also upgrade the minor version, for example, the minor version WT.40.3.34 of the major version 4.0.
During minor version upgrade, the system will automatically detect the minor version and upgrade to the latest version, and you cannot select a target version.
Note:
You can implement the upgrade of major version from v3.2 to v4.0 through data migration. For more information, see Creating Migration Task.

Note

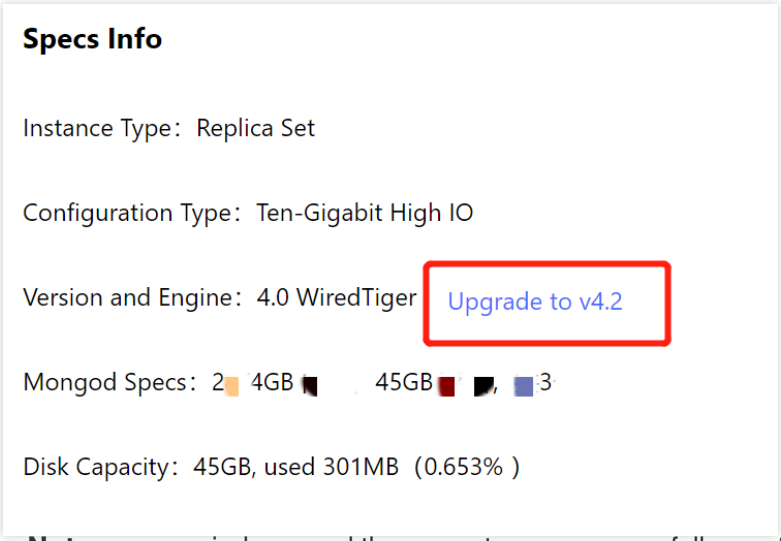
The upgrade process is completely automatic, and there will be several momentary interruptions during the process. We recommend that you upgrade during off-peak hours.

Prerequisites

The instance is not a read-only or disaster recovery instance.
The instance to be upgraded is in **Running** status and is not executing any tasks.
The target version is confirmed.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Instance ID/Name** column of the target instance, click the instance ID to enter the **Instance Details** page.
6. In the **Specs Info** section on the **Instance Details** page, upgrade the major or minor version of the instance.
If the major version of the instance is **3.6**, click **Upgrade to v4.0** after **Version and Engine** to upgrade the version from v3.6 to v4.0.
If the major version of the instance is **4.0**, click **Upgrade to v4.2** after **Version and Engine** to upgrade the version from v4.0 to v4.2.
Click **Upgrade Minor Version** to upgrade the minor version to the latest version by default. For more information, see Version Upgrade.



7. In the **Note** pop-up window, read the prompt message carefully, confirm the upgrade, and click **OK**.

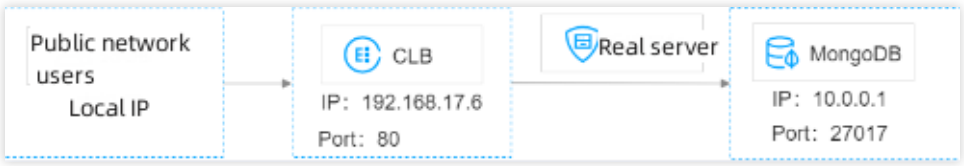
Network Configuration
Enabling Public Network Access

Last updated：2024-07-31 15:14:49

TencentDB for MongoDB supports both private and public network access. This document describes how to configure a public network access address in the console for you to access MongoDB database via the public network. By doing so, you can manage the database more flexibly and conveniently.

Implementation Scheme

TencentDB for MongoDB enables public network access through Cloud Load Balancer (CLB). After a CLB instance listening port is configured in the TencentDB for MongoDB console, the CLB instance will forward requests to the real server when its public IP address and port number are accessed over the public networks. The real server then maps the private and public networks and automatically forwards requests from the public network to the private network server of MongoDB. For more information about CLB and the real server, see CLB.
As is shown below, public network users access CLB via IP address 192.168.17.6 and port number 80. The real server of CLB forwards the request to the actual operating environment of MongoDB database where the private network IP address is 10.0.0.1 and the port number is 27017. In this way, public network users can access the MongoDB database via CLB.



Use Limits

Before enabling the public network access of the MongoDB database, you need to understand the relevant restrictions and requirements which involves MongoDB database, CLB, and network to ensure the security and stability of the database. For more information, see the table below.

Category	Feature	Description
TencentDB for MongoDB	Version	Only for MongoDB replica sets or sharded cluster v4.0, v4.4 and v5.0.
	Sharded cluster	It only supports binding the default instance access address (CLB address) to CLB. **Note:**

		The LB address of a MongoDB sharded cluster forwards client requests to the appropriate Mongos process for processing. For more information, see System Architecture. The CLB listener will listen to the LB IP address and port number of the MongoDB sharded cluster.
	Replica set	When a node is added or deleted for the MongoDB replica set, you need to modify the public network IP address and specify the listening rules for the new node.
Network	VPC	Only TencentDB for MongoDB CLB instances under the same VPC can be bound.
		After enabling the public network access, you can't modify the network for the instance. To modify it, you need to disable the public network access.
	Security group	You can't enable the public network access for the MongoDB instance unbound to a security group. We recommend that you configure a security group to restrict the visiting address. For detailed directions, see Configuring Security Group.
Password-free authentication	Password-free access	You can't enable public network access for the MongoDB instances with password-free access enabled.
CLB	Instance type	MongoDB instances can't be bound to the classic CLB instances. For the differences between CLB (former Application Load Balancer) and classic CLB, see Instance Types Comparison.
	Instance specifications	CLB instances fall into shared instances and LCU-supported instances. A shared instance can support a maximum of 50,000 concurrent connections per minute, but this may not be enough for some high-spec MongoDB instances. Therefore, we recommend that you choose LCU-supported instances to meet your needs. For differences between these two types of instances, see Instance Specifications Comparison.
	Account Types	MongoDB instances do not support binding with CLB instances under traditional account types. They only support binding with CLB instances under standard account types. To determine the account type and learn about methods for upgrading account types, see Account Type Description.
Operation Limits	Disabling Public Network Access	To disable public network access, it is crucial to perform any action through the TencentDB for MongoDB console. Do not attempt to manually delete listeners created by MongoDB within the CLB or remove the entire CLB instance. Failure to follow this instruction may lead to abnormal business connections.

	Network Changes	Modifying the number of instance nodes may affect the public network functionality. Therefore, when making such changes, you must update the public network configuration through the console to maintain uninterrupted public network connectivity.

Note

After the public network access is disabled, MongoDB will only clear the bound listener and will not release or repossess the CLB instance. You can purchase or delete a CLB instance in the CLB console.
We recommend that each MongoDB instance be bound to one dedicated CLB instance, after which MongoDB will manage and maintain the listener. To share a CLB instance with other resources, you need to manage your listener ports well and reserve enough listeners. Otherwise, when a CLB is used by multiple services, the management chaos may occur.

Note：

If the public network feature indicates a health check exception in the backend services, please navigate to the corresponding CLB console to ascertain whether there are any health check risks, or whether the health check source IP range has been allowed.

Prerequisites

You have created a TencentDB for MongoDB instance on v4.0 or later, and the instance runs normally.
You have created CLB instances in the same VPC as that of the MongoDB instance. For more information, see Creating CLB Instances.
For more information on downloading a Windows visual tool, see MongoDB Compass Download (GUI).

Directions

Step 1. Enable public network access

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.

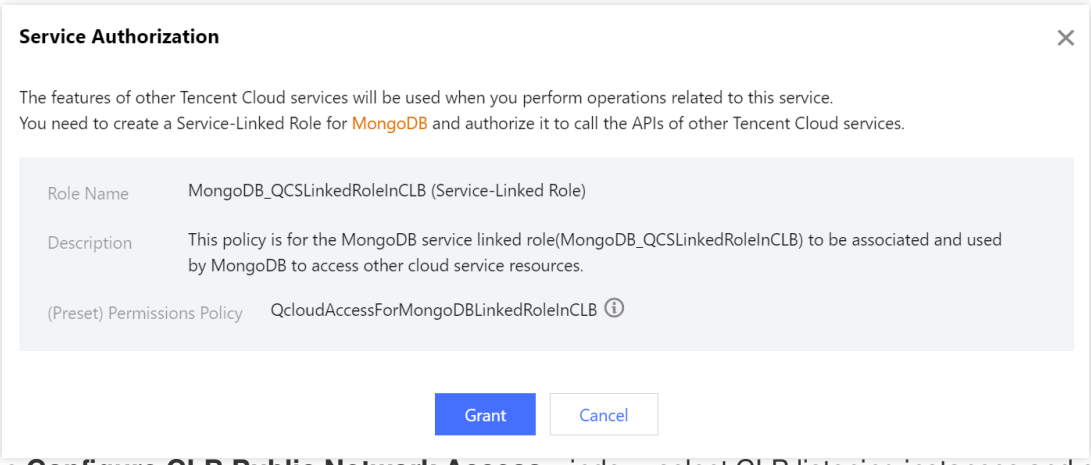
In the search box in the top-right corner, you can search for the target instance by instance ID, instance name, private IP, or tag key.

If you can't find the target instance in the instance list, select **Recycle Bin** on the left sidebar to check whether it is isolated there due to overdue payments. For more information, see Recycle Bin.

5. In the **Instance ID/Name** column of the target instance, click the **Instance ID** to enter the **Instance Details** page.

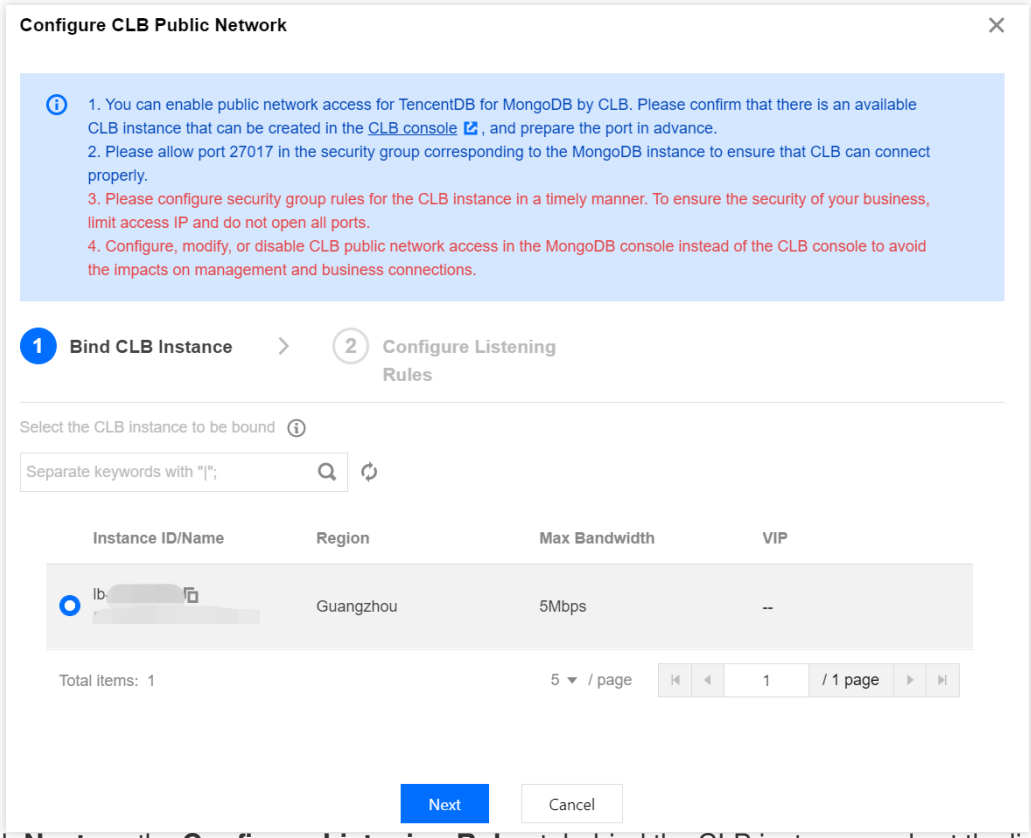
6. In the **Network Configuration** section on the **Instance Details** page, click **Configure CLB Public Network Access** next to **Public Network Access**.

7. In the **Service Authorization** pop-up window, click **Grant**.



8. In the **Configure CLB Public Network Access** window, select CLB listening instances and configure listening rules.

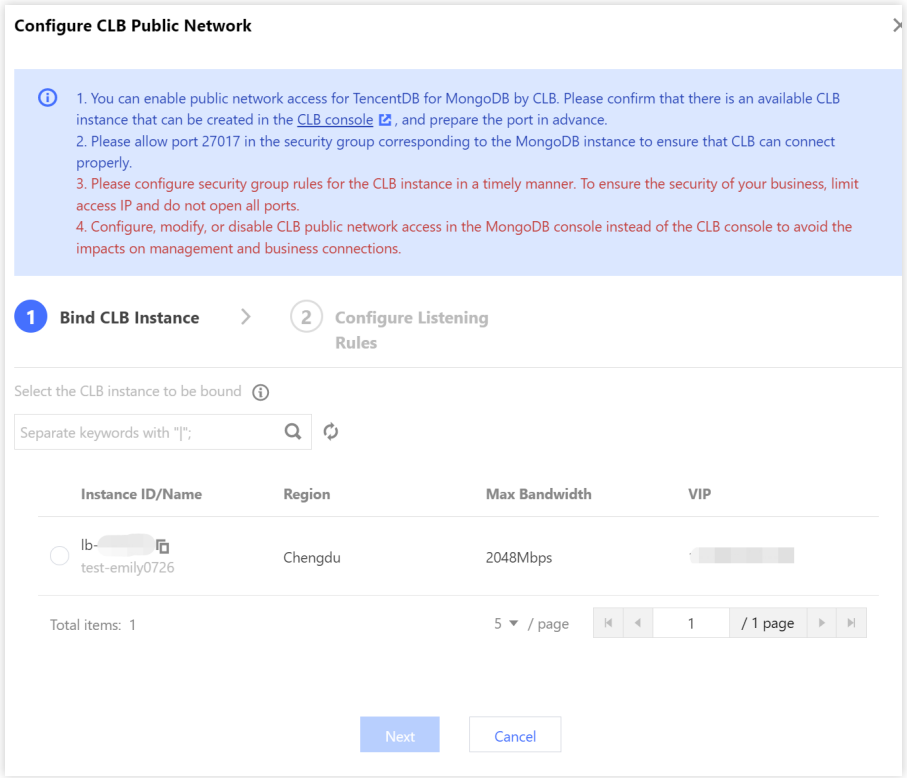
a. On the **Bind CLB Instance** tab, all CLB instances in the same VPC as the current MongoDB instance have been listed. Select the CLB instance to be bound based on the required bandwidth cap specification. VIP refers to the public IP address of the CLB instance.



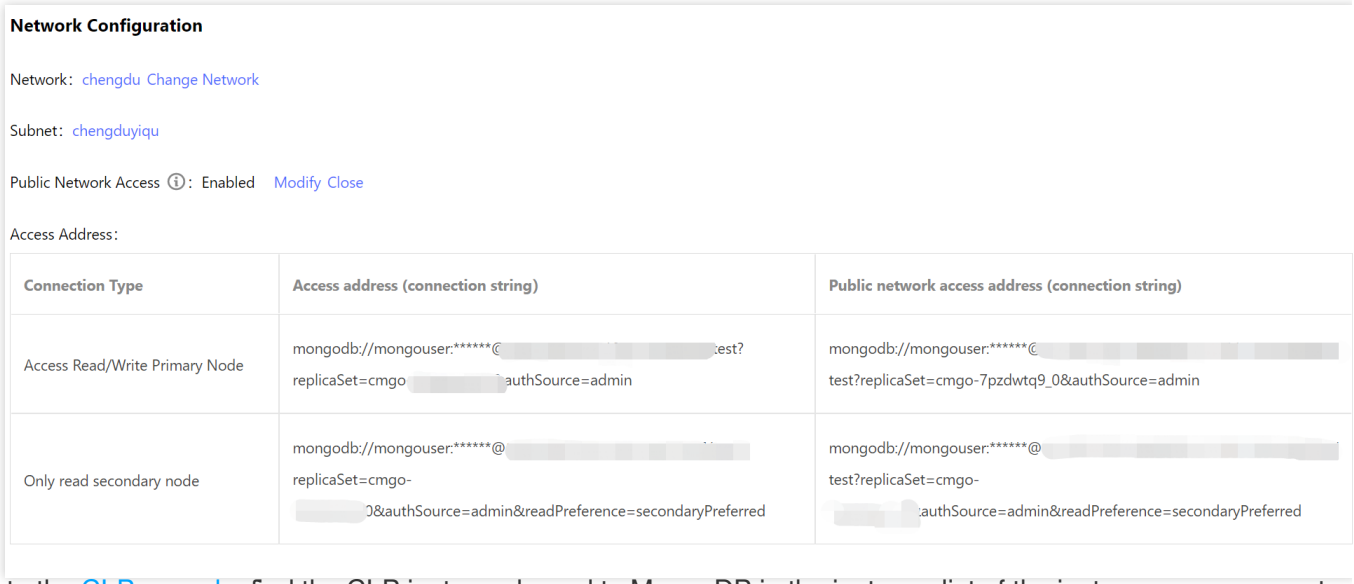
b. Click **Next**, on the **Configure Listening Rules** tab, bind the CLB instance and set the listening rules.

If it is a replica set, configure CLB listening port number for MongoDB primary node and secondary node.

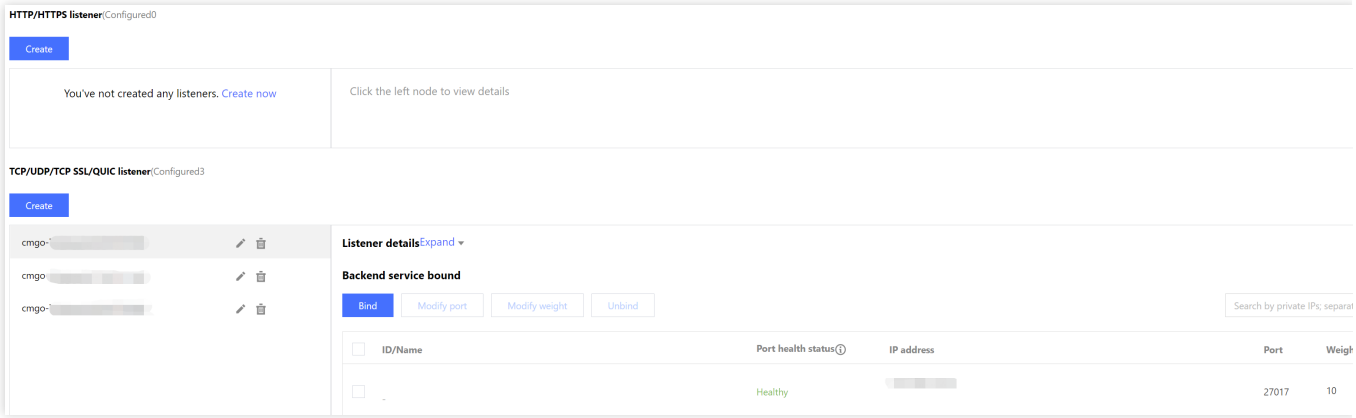
If it is a sharded instance, configure a listening port for the private network address.



9. Click **OK**,and wait for the task to complete. In the **Network Configuration** section on the **Instance Details** page, you can view the connection string of the public network address.



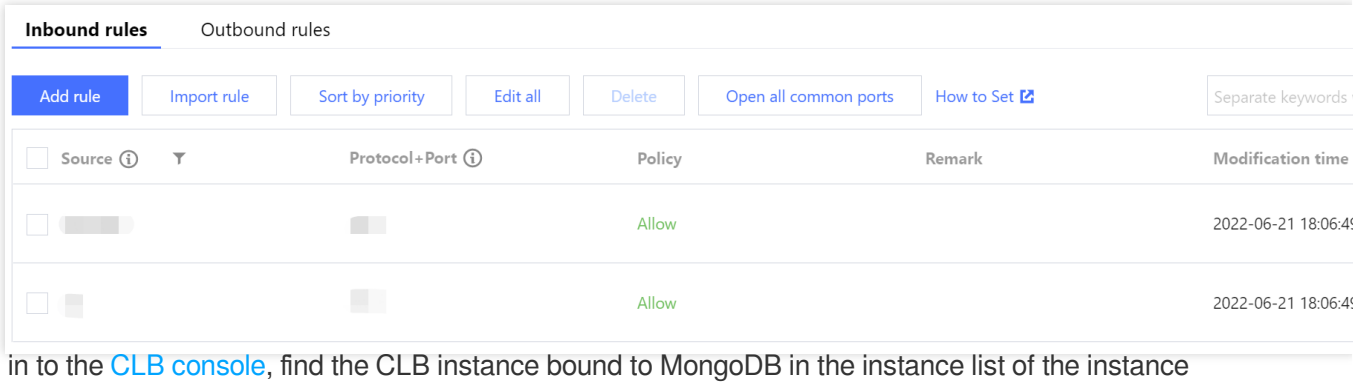
Log in to the CLB console, find the CLB instance bound to MongoDB in the instance list of the instance management page, click **Instance ID** to enter the **Basic Info** tab, select **Listener Management** tab to view the listener.



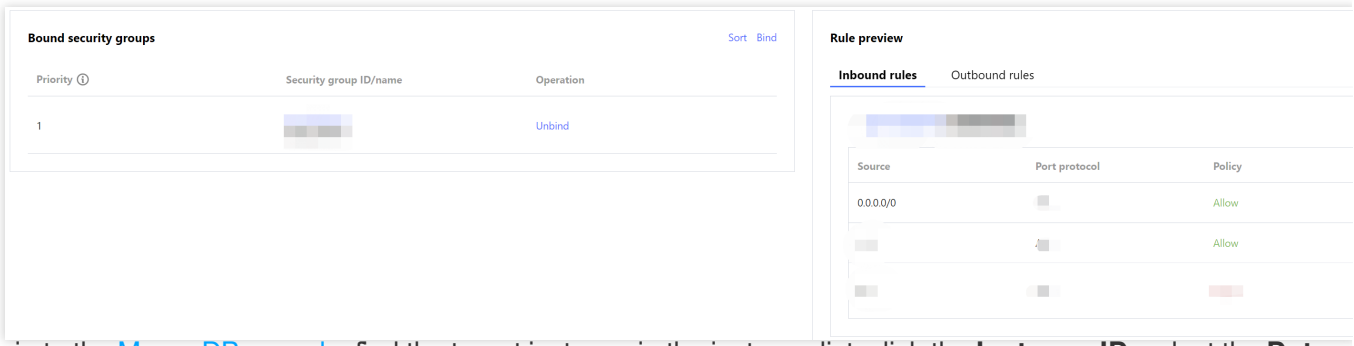
Step 2. Configure a security group

After enabling the public network access, you need to configure security group rules for the CLB and its MongoDB instance timely. By doing so, you can control the access sources to ensure the security of the data access.

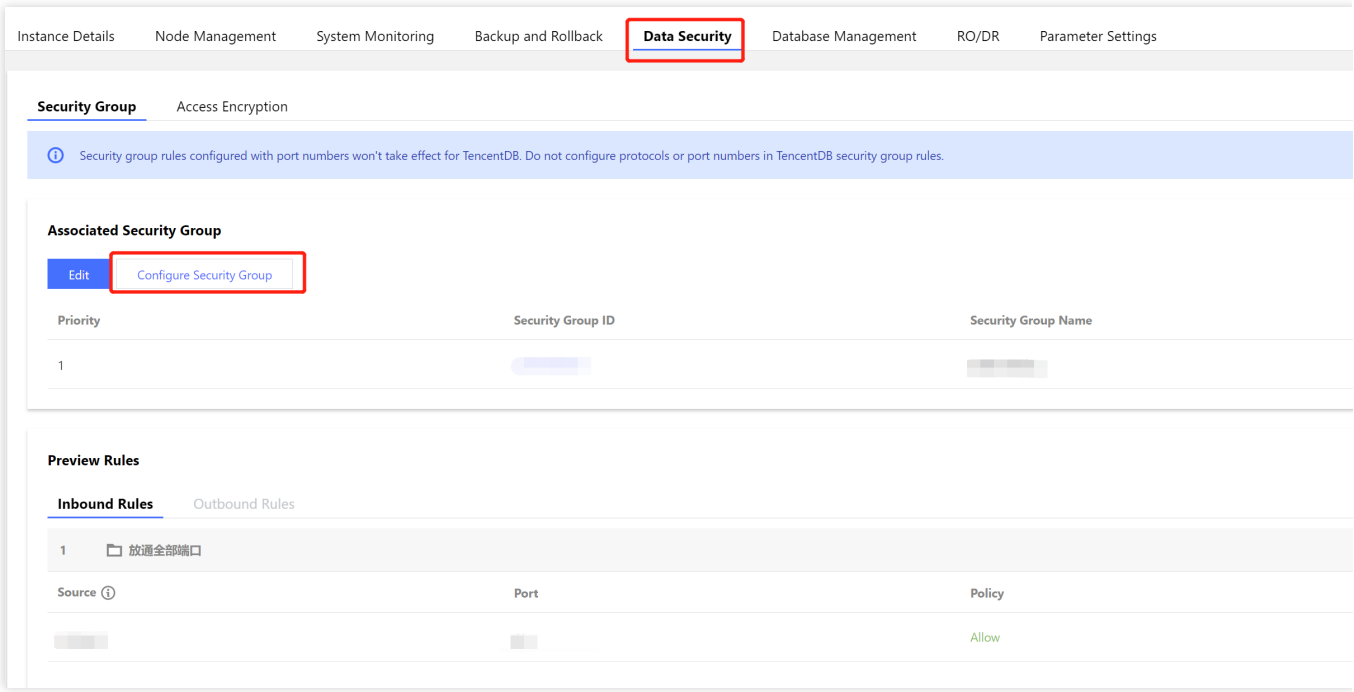
1. Log in to the CVM console > Security Group, create a security group, set the inbound rules, and open the client IP address of mongo-driver and the listening port of your specified MongoDB instance. For detailed directions, see Creating a Security Group.



2. Log in to the CLB console, find the CLB instance bound to MongoDB in the instance list of the instance management page, click **Instance ID** to enter the **Basic Info** tab, and select the **Security Group** tab.
Click **Bind** in the **Bound Security Groups** section, select a created security group in the **Configure Security Group** pop-up window, and click **OK**. For detailed directions, see Configuring CLB Security Group.

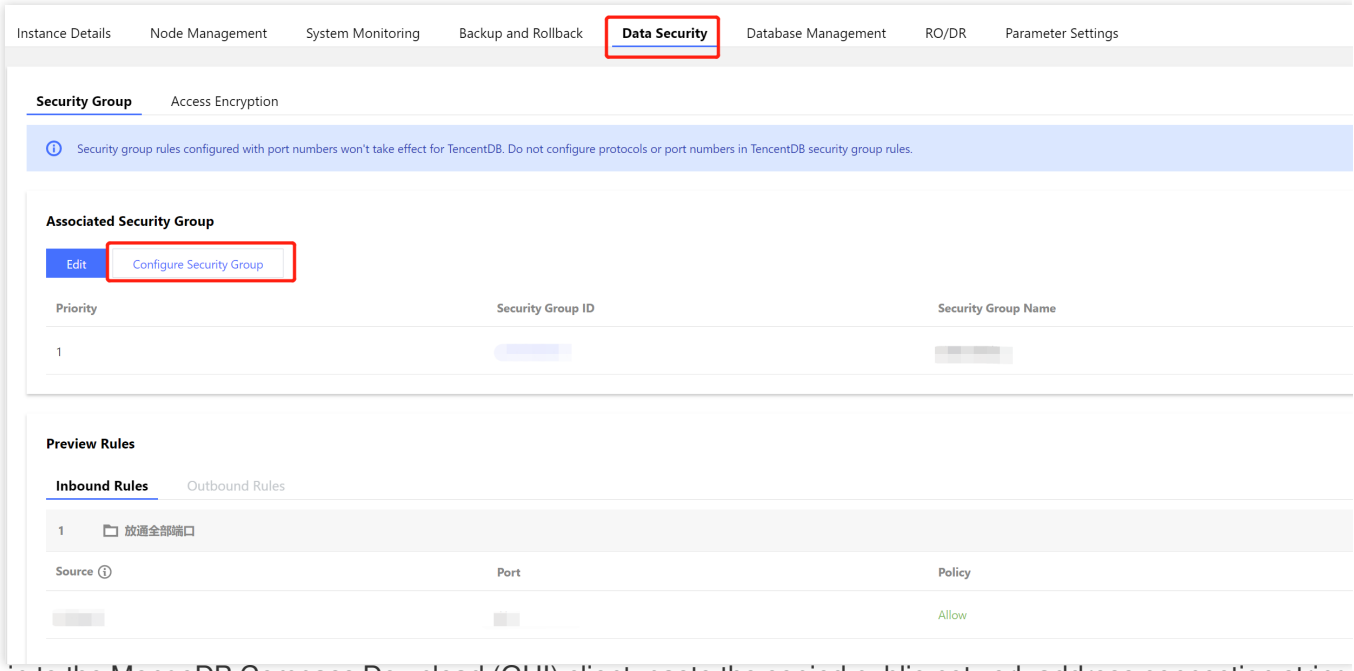


3. Log in to the MongoDB console, find the target instance in the instance list, click the **Instance ID**, select the **Data Security** tab, click **Configure Security Group** to select the desired security group, and click **OK**. For detailed directions, see Configuring Security Group.

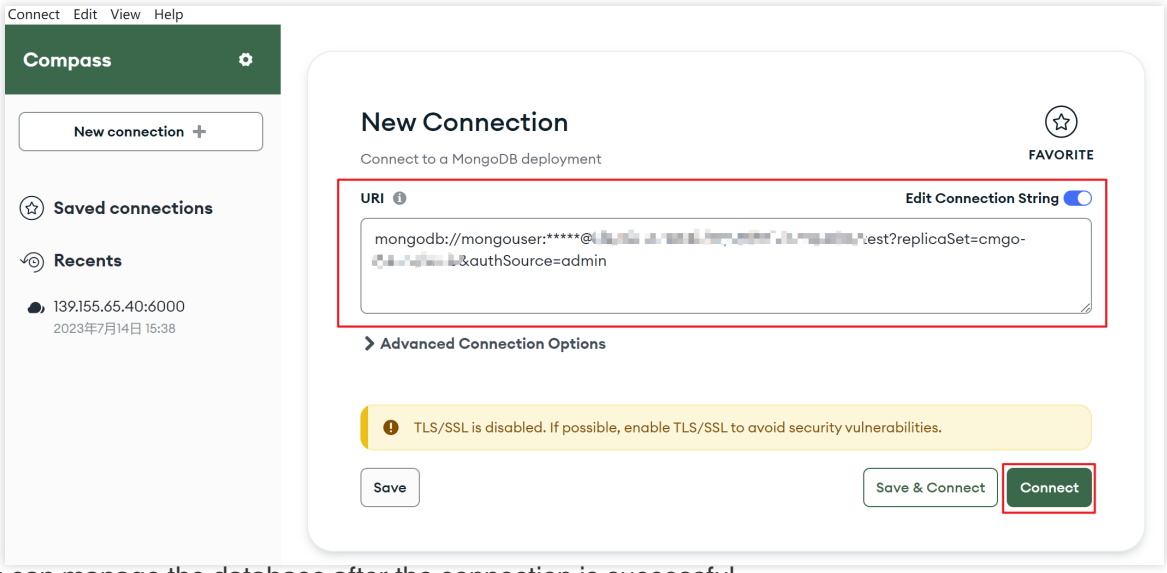


Step 3. Connect to a database instance

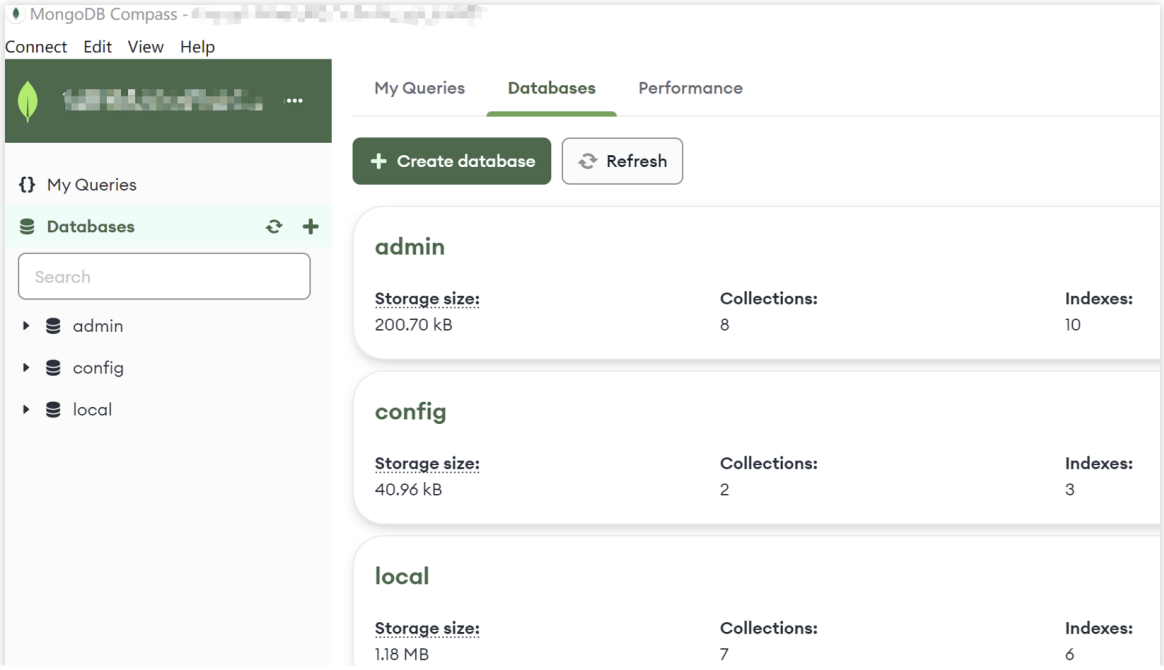
1. Log in to the MongoDB console. In the **Network Configuration** section on the **Instance Details** page, In the Access Address section, copy the connection string of **Access Read/Write Primary Node** or **Only read secondary node** in the **Public network access address (connection string)** column in the **Access Address**.



2. Log in to the MongoDB Compass Download (GUI) client, paste the copied public network address connection string into the **URI** input box. The password information in the connection string is hidden as ***, and you need to manually replace it with the access password of the instance, and click** Connect .



3. You can manage the database after the connection is successful.



FAQs

Last updated：2024-04-07 15:11:09

Although TencentDB for MongoDB manages the configuration of public network access uniformly, the public network access may be interrupted when you separately operate Cloud Load Balancer (CLB). To prevent this, the console will report some common errors.

Problem 1: CLB instance is mistakenly deleted, leading to the public network disconnection.

Issue description

The TencentDB for MongoDB console prompts that the CLB instance doesn't exist, check the instance status in the console.

Possible cause

After the public network access is enabled, the CLB instance bound to TencentDB for MongoDB was deleted.

Solutions

1. Log in to the CLB console, and create a CLB instance as needed.
2. Log in to the TencentDB for MongoDB console. In the **Network Configuration** section on the **Instance Details** page, click **Disable** next to **Public Network Access** to disable the public network access.
3. When it is disabled, click **Configure CLB Public Network Access**, select the CLB instance, configure the listening port for it, and enable the public network access again.

Problem 2: The listener doesn't exist, leading to public network disconnection.

Issue description

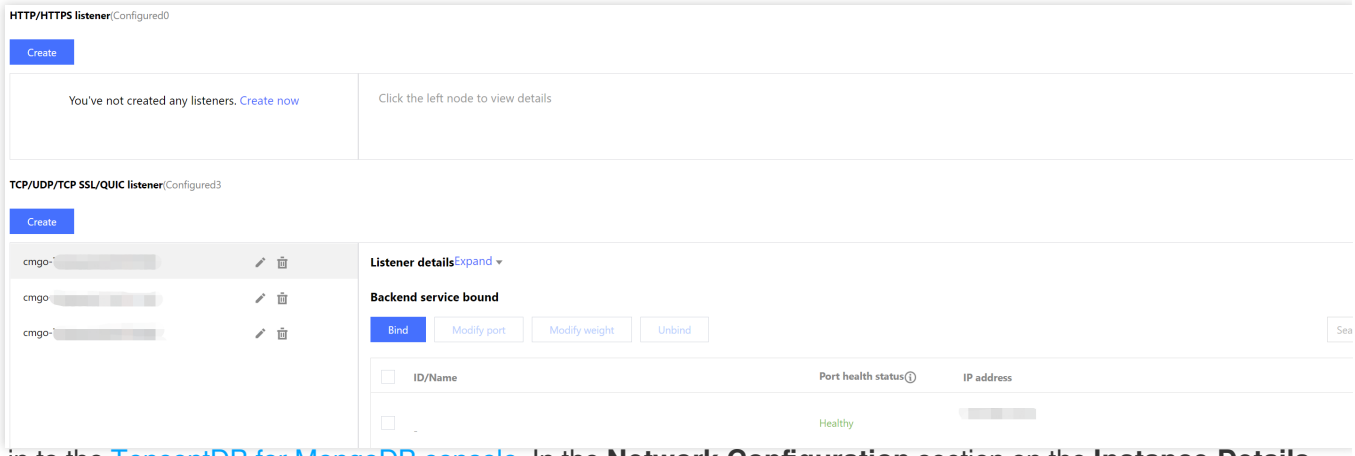
The TencentDB for MongoDB console prompts that the listener doesn't exist, check the listener status in the console.

Possible cause

After enabling the public network access, you may delete the listener configuration in the CLB console.

Solutions

1. Log in to the CLB console. Click **Instance ID** to enter the details page, select the **Listener Management** tab to check whether the listener is deleted mistakenly.



2. Log in to the TencentDB for MongoDB console. In the **Network Configuration** section on the **Instance Details** page, click **Disable** next to **Public Network Access** to disable the public network access.
3. When it is disabled, click **Configure CLB Public Network Access**, select the CLB instance, configure the listening port for it again, and enable the public network access.

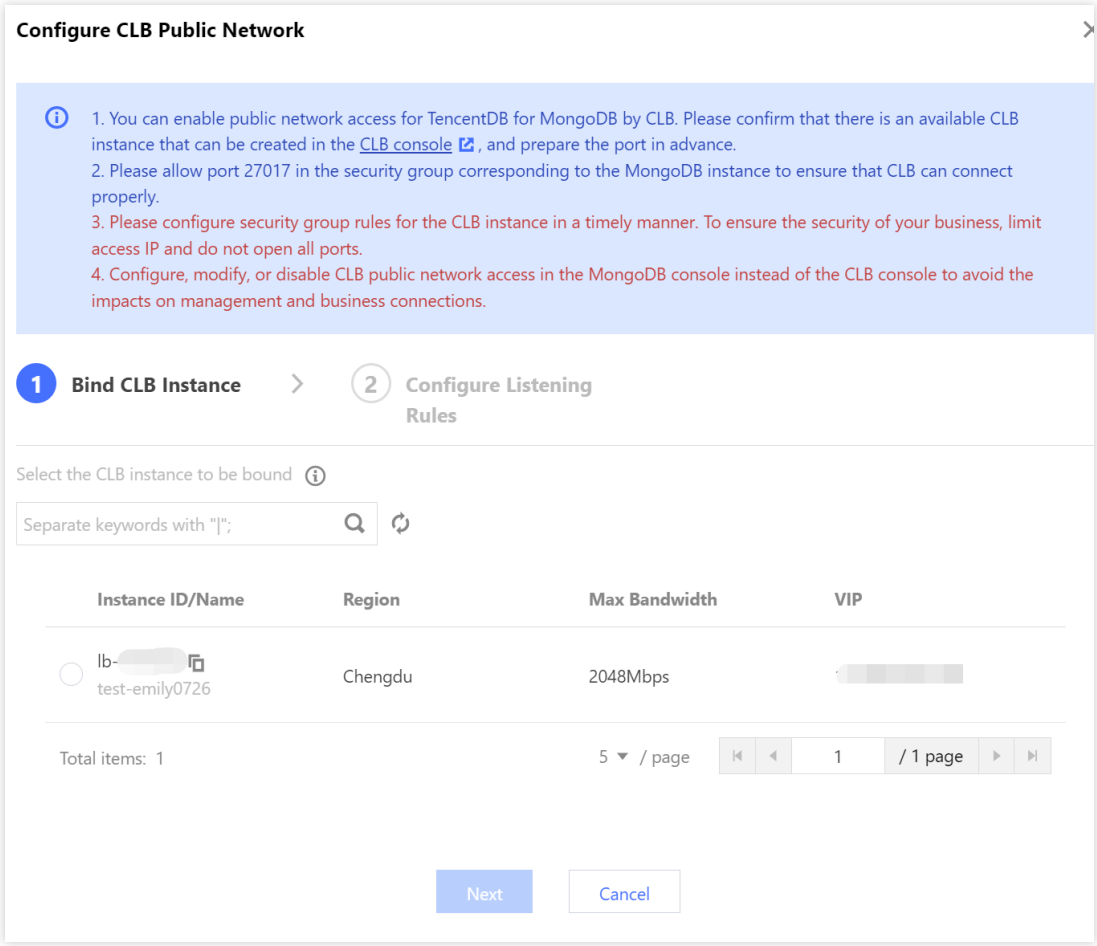
Problem 3: The number of the CLB listeners is not the same as that of the MongoDB VIPs, causing the public network to disconnect.

Issue description

The TencentDB for MongoDB console prompts that the number of the CLB listeners is not the same as that of the instance VIPs. To add nodes or delete the listeners for the current instance, click **Modify** to configure the rules of the public network access.

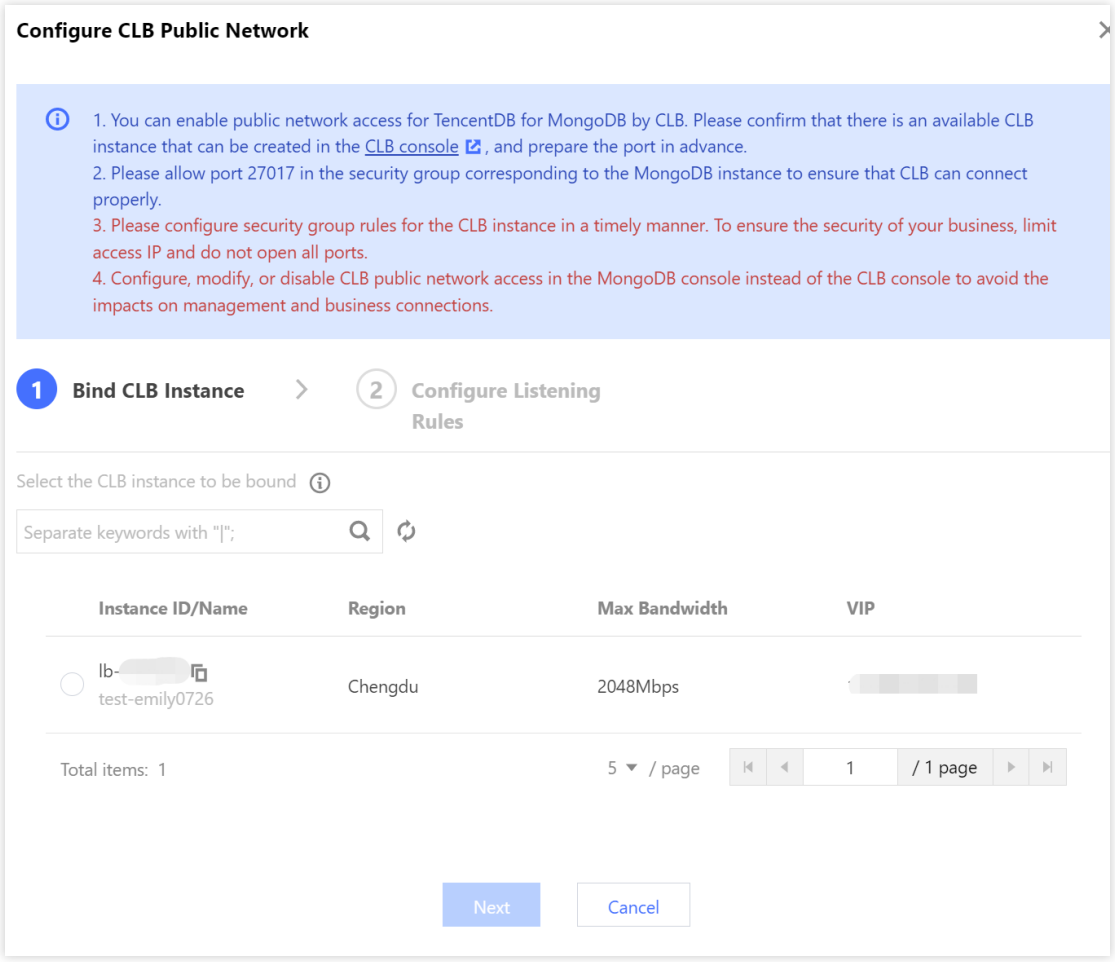
Possible cause

Each CLB listening port has a private network address of the MongoDB instance. When you add nodes for MongoDB instance as instructed in Adding Secondary Node, the newly added node cannot find the corresponding listener, causing an access error.



Solutions

Log in to the TencentDB for MongoDB console. In the **Network Configuration** section on the **Instance Details** page, click **Modify** next to **Public Network Access**. In the **Configure CLB Public Network Access** window, modify the configuration of the public network access.



Problem 4: The CLB listening port is not the same as the one configured. Check the listening rules of the CLB instance or modify the public network access rules.

Issue description

The TencentDB for MongoDB console prompts that the CLB listening port is not the same as the one configured. Check the listening rules of the CLB instance or modify the public network access rules.

Possible cause

You modified the IP port bound to a listener in the CLB console mistakenly, resulting in inconsistency between the listening port and the one actually configured.

Solutions

Option 1: Log in to the MongoDB console. In the **Network Configuration** section on the **Instance Details** page, click **Modify** next to **Public Network Access**. In the **Configure CLB Public Network Access** window, change the configured listening port of the public network to the same as that of the listener.

Option 2: Log in to the MongoDB console. In the **Network Configuration** section on the **Instance Details** page, click **Disable** next to **Public Network Access** to disable the public network access. When it is disabled, click **Configure CLB Public Network Access**, select the CLB instance, configure the listening port for it again, and enable the public network access.

Monitoring

Monitoring Feature

Last updated：2023-08-15 17:00:31

The monitoring feature provided by TencentDB for MongoDB allows you to view the real-time monitoring metric data of instance resources. It collects the monitoring statistics in various forms such as visual chart, table, and dashboard. In addition, it supports setting alarms and pushing alarm notifications promptly, so that you can stay up to date with database service exceptions and adjust your business in time to guarantee stable business operations.

Monitoring Granularity

TencentDB for MongoDB currently doesn't allow you to customize the monitoring data collection granularity. The adaptive policy is as follows:

Time Span	Monitoring Granularity	Retention Period
0–1 day	5 seconds	1 day
0–1 day	1 minute	15 days
0–1 day	5 minutes	31 days
0–1 day	1 hour	93 days
0–1 day	1 days	186 days
0–7 days	1 hour	93 days
0–7 days	1 day	186 days
7–30 days	1 hour	93 days
7–30 days	1 day	186 days

Instance Types for Monitoring

Instance: Primary, read-only, and disaster recovery instances can be monitored, and each instance is provided with a separate monitoring view.
Node: All mongod and mongos nodes can be monitored, and each node is provided with a separate monitoring view.

Monitoring Metrics

Instances

Dimension	Monitoring Metric	Parameter	Unit	Metric Description
CPU Monitoring	Max CPU Utilization of Mongod	mongod_max_mem_usage	%	Maximum CPU utilization among all mongod nodes in the cluster.
	Average Mongod CPU Utilization	monogd_avg_cpu_usage	%	Average CPU utilization of all mongod nodes in the cluster.
	Max CPU Utilization of Mongos	monogs_max_cpu_usage	%	Maximum CPU utilization among all mongos nodes in the sharded cluster.
	Average Mongos CPU Utilization	monogs_avg_cpu_usage	%	Average CPU utilization of all mongos nodes in the sharded cluster.
Memory Monitoring	Max Memory Utilization of Mongod	mongod_max_mem_usage	%	Maximum memory utilization among all mongod nodes in the cluster.
	Average Memory Utilization of Mongod	mongod_avg_mem_usage	%	Average memory utilization of all mongod nodes in the cluster.
	Max Memory Utilization of Mongos	mongos_max_mem_usage	%	Maximum memory utilization among all mongos nodes in the sharded cluster.
	Average Memory Utilization of Mongos	mongos_avg_mem_usage	%	Average memory utilization of all mongos nodes in the sharded cluster.
Disk Monitoring	Storage Space Utilization	disk_usage	%	Proportion of the used disk space to the disk space applied for.
Network Monitoring	Connections	cluster_conn	Pcs	Number of TCP connections to the instance.

	Connection Percentage	connper	%	Proportion of current connections to the maximum connections.
	Inbound Traffic	cluster_view	Bytes	Number of bytes in the traffic inbound to the cluster.
	Outbound Traffic	cluster_netout	Bytes	Number of bytes in the traffic outbound from the cluster.
Latency Monitoring	Average Latency of All Requests	avg_all_request_delay	ms	Average execution latency of all requests in the cluster.
	Average Update Delay	avg_update_delay	ms	Average latency of update requests in the cluster.
	Average Insertion Delay	avg_insert_delay	ms	Average latency of insertion requests in the cluster.
	Average Read Latency	avg_read_delay	ms	Average latency of read requests in the cluster.
	Average Latency of Aggregate Requests	avg_aggregate_delay	ms	Average latency of aggregate requests in the cluster.
	Average Count Delay	avg_count_delay	ms	Average latency of count requests in the cluster.
	Average getMore Delay	avg_getmore_delay	ms	Average latency of getMore requests in the cluster.
	Average Deletion Delay	avg_delete_delay	ms	Average latency of deletion requests in the cluster.
	Average Command Latency	avg_command_delay	ms	Average latency of command requests in the cluster other than INSERT, UPDATE, DELETE, and QUERY requests.
	10–50 ms	10 ms	-	Number of requests with an execution time between 10 and 50 ms.

	50–100 ms	50 ms	-	Number of requests with an execution time between 50 and 100 ms.
	100 ms	100 ms	-	Number of requests with an execution time of more than 100 ms.
Request Monitoring	Total Requests	success_per_second	Counts/sec	Number of requests successfully executed in the cluster per second.
	Insert Requests	insert_per_second	Counts/sec	Number of insertion requests executed in the cluster per second.
	Read Requests	read_per_second	Counts/sec	Number of read requests executed in the cluster per second.
	Update Requests	update_per_second	Counts/sec	Number of update requests executed in the cluster per second.
	Deletion Requests	delete_per_second	Counts/sec	Number of deletion requests executed in the cluster per second.
	Count Requests	count_per_second	Counts/sec	Number of count requests received by the cluster per second.
	getMore Requests	getmore_per_second	Counts/sec	Number of getMore requests received by the cluster per second.
	Aggregate Requests	aggregate_per_second	Counts/sec	Number of aggregate requests in the cluster per second.
	Command Requests	command_per_second	Counts/sec	Number of command requests received by the cluster per second other than INSERT, UPDATE, DELETE, and QUERY requests.
Request	Total requests	node_success	-	Total number of requests in

Count				the cluster.
	Insert Requests	node_inserts	-	Number of insertion requests received by the cluster.
	Read Requests	node_reads	-	Number of read requests received by the cluster.
	Update Requests	node_updates	-	Number of update requests in the cluster.
	Deletion Requests	node_deletes	-	Number of deletion requests in the cluster.
	Count Requests	node_counts	-	Number of count requests received by the cluster.
	getMore Requests	node_getmores	-	Number of getMore requests received by the cluster.
	Aggregate Requests	node_aggregates	-	Number of aggregate requests in the cluster.
	Command Requests	node_commands	-	Number of command requests received by the cluster other than INSERT, UPDATE, DELETE, and QUERY requests.

Mongod node

Dimension	Monitoring Metric	Parameter	Unit	Metric Description
CPU Monitoring	CPU Utilization	cpuusage	%	CPU utilization of the mongod node.
Memory Monitoring	Memory Utilization	memusage	%	Memory utilization of the mongod node.
Disk Monitoring	Used Disk Space	diskusage	MBytes	Disk capacity usage of the mongod node.
	Disk Reads	ioread	Counts/sec	Number of writes on the mongod node per second.

	Disk Writes	iowrite	Counts/sec	Number of writes on the mongod node per second.
Network Monitoring	Inbound Traffic	netout	Bytes	Number of bytes in the traffic inbound to the mongod node.
	Outbound Traffic	netin	Bytes	Number of bytes in the traffic outbound from the mongod node.
Average Request Latency Monitoring	Average Latency of All Requests	node_avg_all_requests_delay	ms	Average latency of all requests received by the mongod node.
	Average Update Delay	node_avg_update_delay	ms	Average latency of update requests on the mongod node.
	Average Insertion Delay	node_avg_insert_delay	ms	Average latency of insertion requests on the mongod node.
	Average Read Latency	node_avg_read_delay	ms	Average latency of read requests on the mongod node.
	Average Latency of Aggregate Requests	node_avg_aggregate_delay	ms	Average latency of aggregate requests on the mongod node.
	Average Count Delay	node_avg_count_delay	ms	Average latency of count requests on the mongod node.
	Average getMore Delay	node_avg_getmore_delay	ms	Average latency of getMore requests on the mongos node.
	Average Deletion Delay	node_avg_delete_delay	ms	Average latency of deletion requests on the mongod node.
	Average Command Latency	node_avg_command_delay	ms	Average latency of command requests on the mongod node.

	10-50 ms	10 ms	-	Number of requests with an execution time between 10 and 50 ms.
	50-100 ms	50 ms	-	Number of requests with an execution time between 50 and 100 ms.
	100 ms	100 ms	-	Number of requests with an execution time of more than 100 ms.
Request Monitoring	Total Requests	node_success_per_second	Counts/sec	Total number of requests on the mongod node per second.
	Insert Requests	node_insert_per_second	Counts/sec	Number of insertion requests on the mongod node per second.
	Read Requests	node_read_per_second	Counts/sec	Number of read requests on the mongod node per second.
	Update Requests	node_update_per_second	Counts/sec	Number of update requests on the mongod node per second.
	Deletion Requests	node_delete_per_second	Counts/sec	Number of deletion requests on the mongod node per second.
	Count Requests	node_count_per_second	Counts/sec	Number of count requests received by the mongod node per second.
	getMore Requests	node_getmore_per_second	Counts/sec	Number of getMore requests received by

				the mongod node per second.
	Aggregate Requests	node_aggregate_per_second	Counts/sec	Number of aggregate requests in the mongod node per second.
	Command Requests	node_command_per_second	Counts/sec	Number of command requests received by the mongod node per second other than INSERT, UPDATE, DELETE, and QUERY requests.
Kernel Monitoring	Active Write Requests	ar	-	Number of active write requests on the mongod node.
	Active Read Requests	aw	-	Number of active read requests on the mongod node.
	Queuing Read Requests	qr	-	Length of the client read request queue on the mongod node.
	Queuing Write Requests	qw	-	Length of the client write request queue on the mongod node.
	Pieces of Data Deleted via TTL	ttl_deleted	-	Number of documents deleted through TTL on the mongod node.
	TTL Initiation Times	ttl_pass	-	Number of documents deletions from the TTL collection performed by the backend process.
	Active Sessions	active_session	-	Number of active sessions on the node.
	Oplog Retention Period	node_oplog_reserved_time	hours	Oplog retention period.

	Primary/Secondary Delay	node_slavedelay	seconds	Delay time between the primary and secondary nodes.
	Cache Hit Rate	replicaset_node	%	Cache hit rate of the current cluster.
	Cache Utilization (%)	node_cache_used	%	Percentage of the used cache to the total cache space.
	Dirty Data (%) in Cache	node_cache_dirty	%	Percentage of the size of dirty data in the cache to the total cache space.
Request Count	Total requests	node_success	-	Total number of requests in the cluster.
	Insert Requests	node_inserts	-	Number of insertion requests in the cluster.
	Read Requests	node_reads	-	Number of read requests in the cluster.
	Update Requests	replicaset_node	-	Number of update requests in the cluster.
	Delete Requests	node_deletes	-	Number of deletion requests in the cluster.
	Count Requests	node_counts	-	Number of count requests received by the cluster.
	getMore Requests	node_getmores	-	Number of getMore requests received by the cluster.
	Aggregate Requests	node_aggregates	-	Number of aggregate requests in the cluster.
	Command Requests	node_commands	-	Number of command requests received by the cluster other than INSERT, UPDATE,

				DELETE, and QUERY requests.

Mongos node (sharded cluster)

Dimension	Monitoring Metric	Monitoring Metric	Unit	Metric Description
CPU Monitoring	CPU Utilization	cpuusage	%	CPU utilization of the mongos node.
Memory Monitoring	Memory Utilization	memusage	%	Memory utilization of the mongos node.
Network Monitoring	Private Inbound Traffic	netout	Bytes	Number of bytes in the traffic inbound to the mongos node.
	Private Outbound Traffic	netin	Bytes	Number of bytes in the traffic outbound from the mongos node.
Latency Monitoring	Average Latency of All Requests	node_avg_all_request_delay	ms	Average latency of all requests received by the mongos node.
	Average Update Delay	node_avg_update_delay	ms	Average latency of update requests on the mongos node.
	Average Insertion Delay	replicaset_node	ms	Average latency of insertion requests on the mongos node.
	Average Read Latency	node_avg_read_delay	ms	Average latency of read requests on the mongos node.
	Average Latency of Aggregate Requests	node_avg_aggregate_delay	ms	Average latency of aggregate requests on the mongos node.
	Average Count Delay	node_avg_count_delay	ms	Average latency of count requests on the mongos

				node.
	Average getMore Delay	node_avg_getmore_delay	ms	Average latency of getMore requests on the mongos node.
	Average Deletion Delay	node_avg_delete_delay	ms	Average latency of deletion requests on the mongos node.
	Average Command Latency	node_avg_command_delay	ms	Average latency of command requests on the mongos node other than INSERT, UPDATE, DELETE, and QUERY requests.
	10-50 ms	10 ms	-	Number of requests per second with an execution time between 10 and 50 ms.
	50-100 ms	50 ms	-	Number of requests per second with an execution time between 50 and 100 ms.
	100 ms	100 ms	-	Number of requests per second with an execution time of more than 100 ms.
Request Monitoring	Total Requests	qps	Counts/sec	Total number of requests on the mongos node per second.
	Insert Requests	inserts	Counts/sec	Number of insertion requests on the mongos node per second.
	Read Requests	reads	Counts/sec	Number of read requests on the mongos node per second.
	Update Requests	updates	Counts/sec	Number of update requests on the mongos node per second.
	Deletion Requests	deletes	Counts/sec	Number of deletion requests on the mongos node per second.
	Count Requests	counts	Counts/sec	Number of count requests received by the mongos node

				per second.
	getMore Requests	getmores	Counts/sec	Number of getMore requests received by the mongos node per second.
	Aggregate Requests	aggregates	Counts/sec	Number of aggregate requests in the mongos node per second.
	Command Requests	commands	Counts/sec	Number of command requests received by the mongos node per second other than INSERT, UPDATE, DELETE, and QUERY requests.
Request Count	Total requests	node_success	-	Total number of requests received by the mongos node.
	Insert Requests	node_inserts	-	Number of insertion requests received by the mongos node.
	Read Requests	node_reads	-	Number of read requests received by the mongos node.
	Update Requests	node_updates	-	Number of update requests received by the mongos node.
	Delete Requests	node_deletes	-	Number of deletion requests received by the mongos node.
	Count Requests	node_counts	-	Number of count requests received by the mongos node.
	getMore Requests	node_getmores	-	Number of getMore requests received by the mongos node.
	Aggregate Requests	node_aggregates	-	Number of aggregate requests received by the

				mongos node.
	Command Requests	node_commands	-	Number of command requests received by the mongos node other than INSERT, UPDATE, DELETE, and QUERY requests.

Viewing Monitoring Data

Last updated：2024-10-10 11:50:22

TencentDB for MongoDB allows you to view the change trend of each monitoring metric. This helps you stay up to date with the running status and performance of database resources, so that you can make prejudgments and prevent risks.

Background

Tencent Cloud Observability Platform (TCOP) is a real-time monitoring and alarming service for Tencent Cloud resources. It collects the data of various monitoring metrics of Tencent Cloud services and displays the data through visual charts, helping you intuitively understand the running status and performance of services. For more information, see TCOP.

Note:

Cloud Monitor (CM) was renamed **TCOP** on February 23, 2023.

In TencentDB for MongoDB, you can use TCOP to create dashboards and various types of charts to compare the metric data of multiple instances. In this way, you can efficiently analyze the changes of monitoring metrics. You can also use TCOP to configure real-time alarms for exceptions during database operations, allowing you to remove risks as soon as they arise.

Version description

Currently, all TencentDB for MongoDB versions support instance monitoring.

Billing Overview

Basic TCOP features such as alarming and monitoring data collection are free of charge.

Currently, only **alarm SMS messages** are billed.

Notes

The monitoring data is retained for 30 days.

After receiving the alarms reported by Tencent Cloud, you need to troubleshoot problems accordingly.

Prerequisites

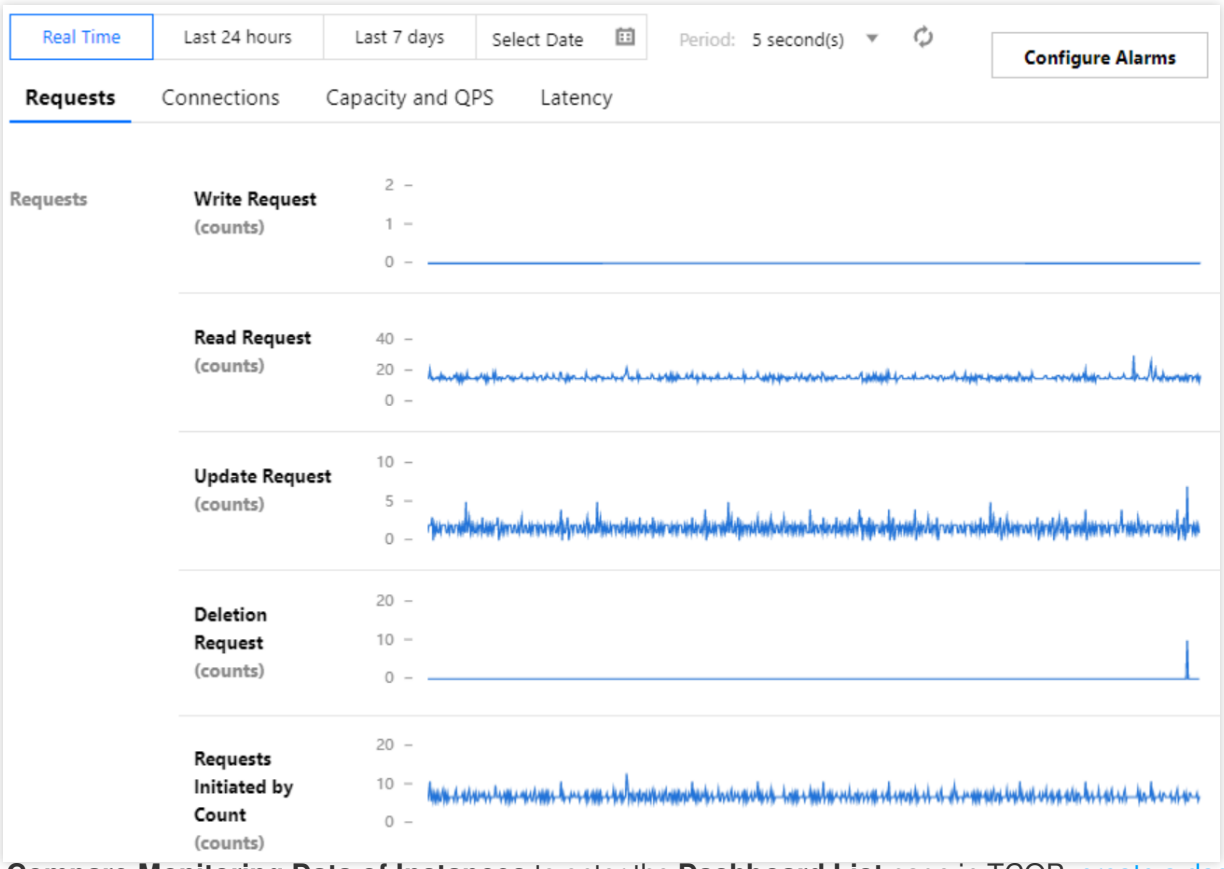
You have activated TCOP.
You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.

Directions

Quickly viewing instance monitoring data

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Shard Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Monitoring/Status** column of the target instance, click  to open the instance monitoring panel, where you can quickly view the instance monitoring data.

You can select **Real Time**, **Last 24 Hours**, **Last 7 Days**, or any time period to view the corresponding monitoring data.
On the **Requests**, **Connections**, **Capacity and QPS**, and **Latency** tabs, you can view the data of monitoring metrics in different categories.
In the **Time Granularity** drop-down list, you can set the time granularity for monitoring data collection to get finer-grained data.



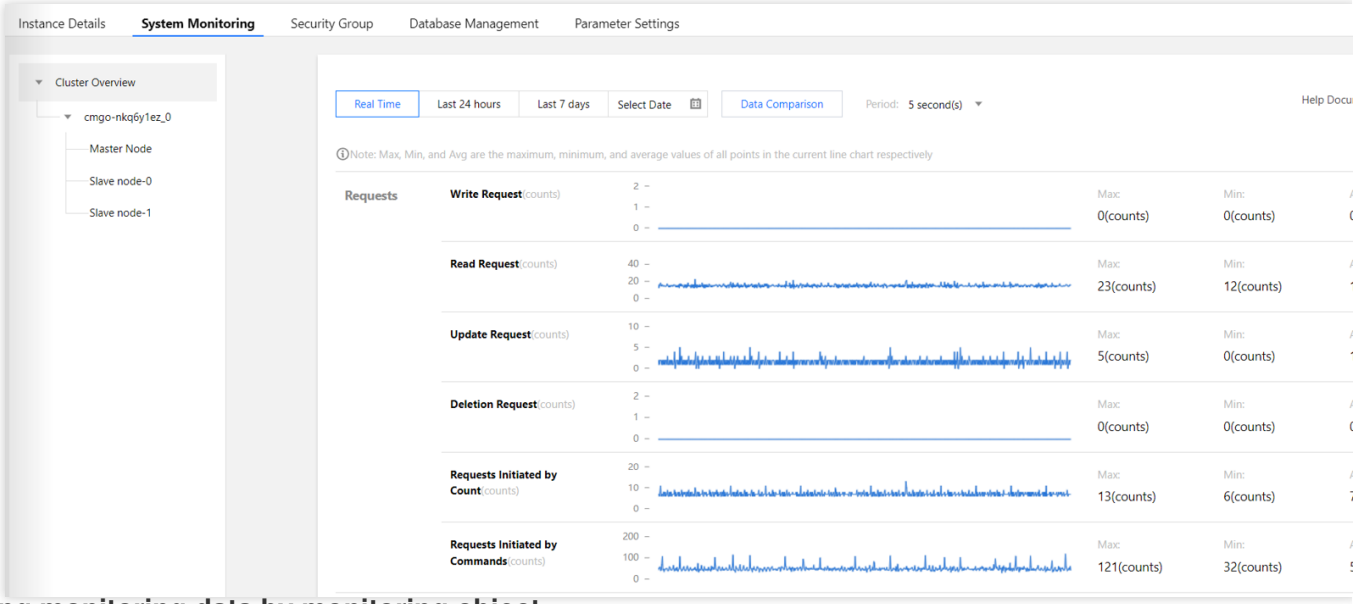
Select **Compare Monitoring Data of Instances** to enter the **Dashboard List** page in TCOP, create a dashboard, select the instances to be monitored, set the monitoring chart, and then you can compare the monitoring data of multiple instances in the same chart.

Click **Configure Alarms** to enter the **Create Alarm Policy** page in TCOP, set **Policy Type** to **TencentDB for MongoDB Instance**, select an **alarm recipient**, set the **trigger condition** of the monitoring metric, and configure

the alarm notification method. In this way, you can stay on top of the business exceptions and prevent risks and failures promptly. For detailed directions, see Creating Alarm Policy.

Viewing monitoring details

1. In the instance list, find the target instance.
2. Click the target instance ID to enter the **Instance Details** page.
3. Click the **System Monitoring** tab to view the change trend of each monitoring metric of the entire cluster as shown below (with a replica set instance as an example):



Viewing monitoring data by monitoring object

Replica set: On the left of the **System Monitoring** page, select the specific instance name, primary node, and secondary node under the **Cluster Overview** to view the monitoring metric data of different monitoring objects.
Sharded instance: On the left of the **System Monitoring** page, select the specific shard name, primary node, and secondary node under the **Cluster Overview** to view the monitoring metric data of different monitoring objects.

Viewing monitoring data for specified time period

In the top-right corner of the **System Monitoring** page, select **Real Time**, **Last 24 Hours**, **Last 7 Days**, or any time period to view the corresponding monitoring data.

Viewing monitoring data at different time granularities

In the top-right corner of the **System Monitoring** page, select **5 seconds**, **1 minute**, **5 minutes**, or **1 day** in the drop-down list after **Time Granularity** to view the monitoring data at different time granularities.

Zooming in change trend chart of single metric

In the metric list on the right of the **System Monitoring** page, find the target metric and click

to zoom in its change trend chart. You can select a time period and set a time granularity to analyze the metric change trend in a more refined manner.

Exporting monitoring chart

Exporting one metric: In the metric list, select the target metric, click

, and select **Export as Image** to export its change trend chart. You can also select **Export Data** to view and analyze the monitoring data with Excel locally.

Batch exporting monitoring data: Click **Export Data** above the metric list, select the target metrics in the **Export Data** window, click **Export**, and then you can view and analyze the monitoring data with Excel locally.

Setting alarm

In the top-right corner of the instance monitoring page, click **Configure Alarms** to enter the **Create Alarm Policy** page in TCOP, set **Policy Type** to **TencentDB for MongoDB Instance**, select an **alarm recipient**, set the **trigger condition** of the monitoring metric, and configure the alarm notification method. In this way, you can stay on top of the metric exceptions and prevent risks and failures promptly. For detailed directions, see Creating Alarm Policy.

Comparing data

In the top-right corner of the instance monitoring page, you can click **Data Comparison** and set the time period. By default, the data within the past hour is obtained. The curves of the monitoring metric on today and yesterday within the specified time range are displayed in different colors.

Configuring Alarm Policy

Last updated：2024-10-10 14:20:43

Overview

You can configure alarm rules for monitoring metrics to prevent your system operations from being disrupted when these metrics reach a certain value. When monitoring data meets the configured conditions, the system can check it automatically and send alarm notifications to the admin. This allows you to stay on top of business exceptions and solve them quickly.

Alarming and Monitoring Metrics

TencentDB for MongoDB provides alarm configurations in three dimensions: instance, replica set, and node. You can set alarm rules for metrics of each dimension. The details are as follows:
Instance: The instance dimension is for the entire MongoDB cluster. It monitors the number of requests, disks, latency, and connections of the entire cluster.
Replica set: Each replica set of TencentDB for MongoDB adopts a one-primary-multiple-secondary architecture, and each shard of the sharded cluster is also a replica set structure, so the database documents are stored in the replica set. This dimension is for the architecture of the stored document. It monitors the cache dirty data, cache utilization, request hit rate, disk utilization, oplog storage time, and primary-secondary delay.
Node(Mongod, Mongos): This dimension is for all nodes in the database cluster. It monitors the usage of Mongod and Mongos nodes, including CPU, memory, disk, inbound and outbound traffic, number of read and write requests, waiting queues, and number of connections.

Alert Metrics

Before alarms are configured, familiarize yourself with the monitoring metrics defined for different policy dimensions and view the alarm configuration recommendations for key metrics. For metrics without specific recommendations, configure them based on your actual business needs.

Instance dimension

Monitoring Metric	Unit	Description	Alarm Configuration Recommendation
Write Requests	-	Number of write requests received by the instance.	-

Read Requests	-	Number of read requests received by the instance.	-
Update Requests	-	Number of update requests received by the instance.	-
Deletion Requests	-	Number of deletion requests received by the instance.	-
Count Requests	-	Number of total requests received by the instance.	-
Aggregate Requests	-	Number of aggregate requests received by the instance.	-
Successfully Executed Requests	-	Number of requests received by the instance that are executed successfully.	-
Disk Utilization	%	The percentage of the used space to the total space of the current disk.	Monitoring cycle: 1 minute. If the utilization is greater than or equal to 90% and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Requests Consuming 10-50 ms	-	Number of requests with an execution time between 10 and 50 ms.	-
Requests Consuming 50-100 ms	-	Number of requests with an execution time between 50 and 100 ms.	-
Requests Consuming More Than 100 ms	-	Number of requests with an execution time of more than 100 ms.	Monitoring cycle: 1 minute. If the count is greater than or equal to 100 and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Connection Utilization	%	The percentage of current connections to the maximum connections of the cluster.	Monitoring cycle: 1 minute. If the utilization is greater than or equal to 90% and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.

Requests per Second	-	Number of requests received by the instance per second.	-
Command Requests	-	Number of command requests received by the cluster other than INSERT, UPDATE, DELETE, and QUERY requests.	-
Number of Connections	-	Number of TCP connections from cluster clients.	-

Replica set dimension

Monitoring Metric	Unit	Description	Alarm Configuration Recommendation
Dirty Data in Cache	%	The percentage of dirty data in the cache (in bytes) to the maximum cache size.	Monitoring cycle: 1 minute. If the percentage is greater than or equal to 20% and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Cache Utilization	%	The ratio of the actual cache usage to the configured maximum cache size.	-
Disk Utilization	%	The percentage of the used space to the total space of the current disk.	Monitoring cycle: 1 minute. If the utilization is greater than or equal to 90% and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Cache Hit Rate	%	The ratio of the number of requests that retrieve data from the cache to the total number of requests.	-
Oplog Retention Period	hours	Oplog is used to record the operation log of the database, and this metric counts its storage time.	-

Average primary-secondary delay in Unit Time	s	In the replica set architecture, the secondary node periodically polls the oplog (operation log) of the primary node to replicate the data from it. This metric counts the primary-secondary delay in data sync.	Monitoring cycle: 1 minute. If the latency is greater than or equal to 1800 seconds and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.

Mongod node

Monitoring Metric	Unit	Description	Alarm Configuration Recommendation
CPU Utilization	%	The percentage of time the CPU is executing processes to the total CPU time.	Monitoring cycle: 1 minute. If the utilization is greater than or equal to 80% and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Memory Utilization	%	The percentage of the used space in the current memory to the total memory space.	-
Inbound Network Traffic	KB/s	Number of bytes per second in the traffic inbound to the node.	-
Outbound Network Traffic	KB/s	Number of bytes per second in the traffic outbound from the node.	-
Read Requests in Queue	Pcs	Number of read requests waiting in the queue.	Monitoring cycle: 1 minute. If the count is greater than or equal to 40 and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Write Requests in Queue	Pcs	Number of write requests waiting in the queue.	Monitoring cycle: 1 minute. If the count is greater than or equal to 40 and the metric

			exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Number of Connections	Pcs	Number of client connections.	
Used Disk Space	MB	Used node disk capacity.	
WiredTiger Active Reads	Pcs	Number of active read requests in the memory.	Monitoring cycle: 1 minute. If the count is greater than or equal to 40 and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
WiredTiger Active Writes	Pcs	Number of active write requests in the memory.	Monitoring cycle: 1 minute. If the count is greater than or equal to 40 and the metric exception persists for 3 data points, an alarm will be triggered every 30 minutes.
Data Entries Deleted via TTL	Pcs	The number of data entries automatically deleted by the database after the TTL expires.	-
TTL Run Times	-	The number of times data is checked within the TTL time set in the database.	-

Mongos node

Monitoring Metric	Unit	Description	Alarm Configuration Recommendation
CPU utilization	%	Indicates the percentage of CPU time spent on executing processes compared to total CPU time.	Monitoring cycle: 1 minute. If the utilization is greater than or equal to 80% and the metric exception persists for 3 data points, an alarm will

			be triggered every 30 minutes.
Memory utilization	%	Indicates the percentage of memory currently in use on the Mongos node compared to total memory capacity.	-
Inbound network traffic	MB/s	Statistics of inbound traffic per second for the node.	-
Outbound network traffic	MB/s	Statistics of outbound traffic per second for the node.	-

Billing Overview

TCOP allows you to configure alarm policies to monitor the key metrics of instances and offers a free trial.
Currently, only **alarm SMS messages** are charged. For more information, see Purchase Guide.

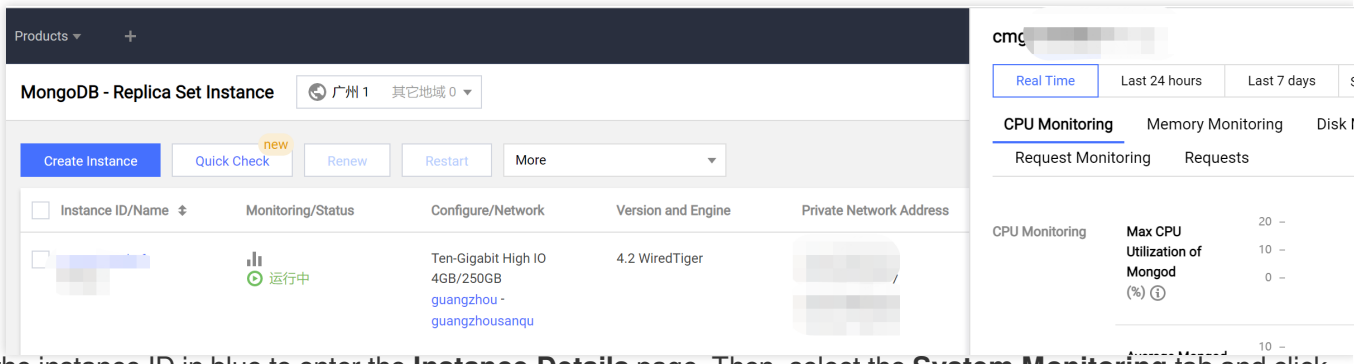
Prerequisites

You have activated TCOP.
The database instance is in **Running** status.
You have collected the information of the recipients of alarm notifications, such as email address.

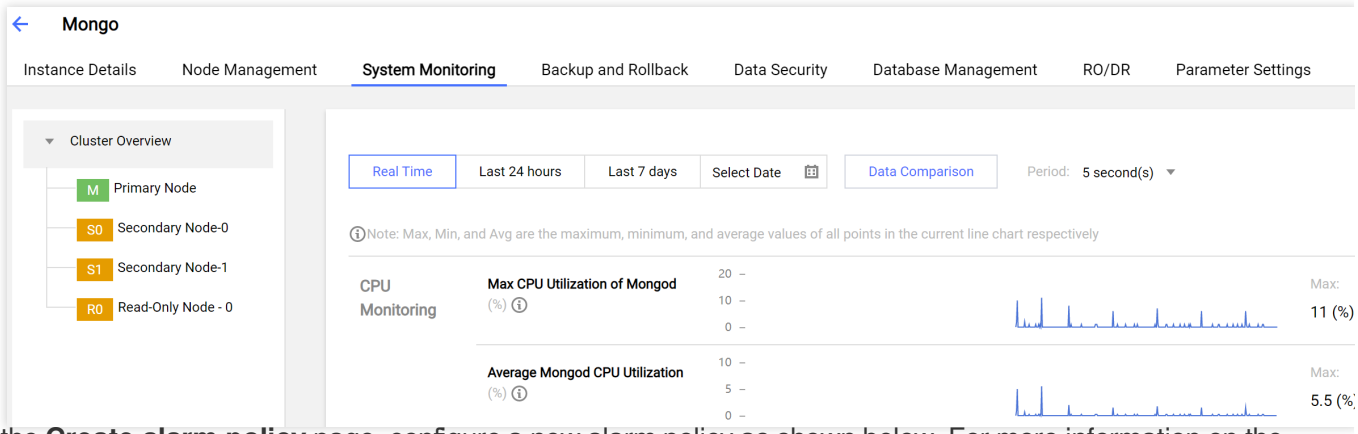
Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**.
The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the row of the target instance, enter the **Create Policy** page of TCOP in any of the following ways:
Click

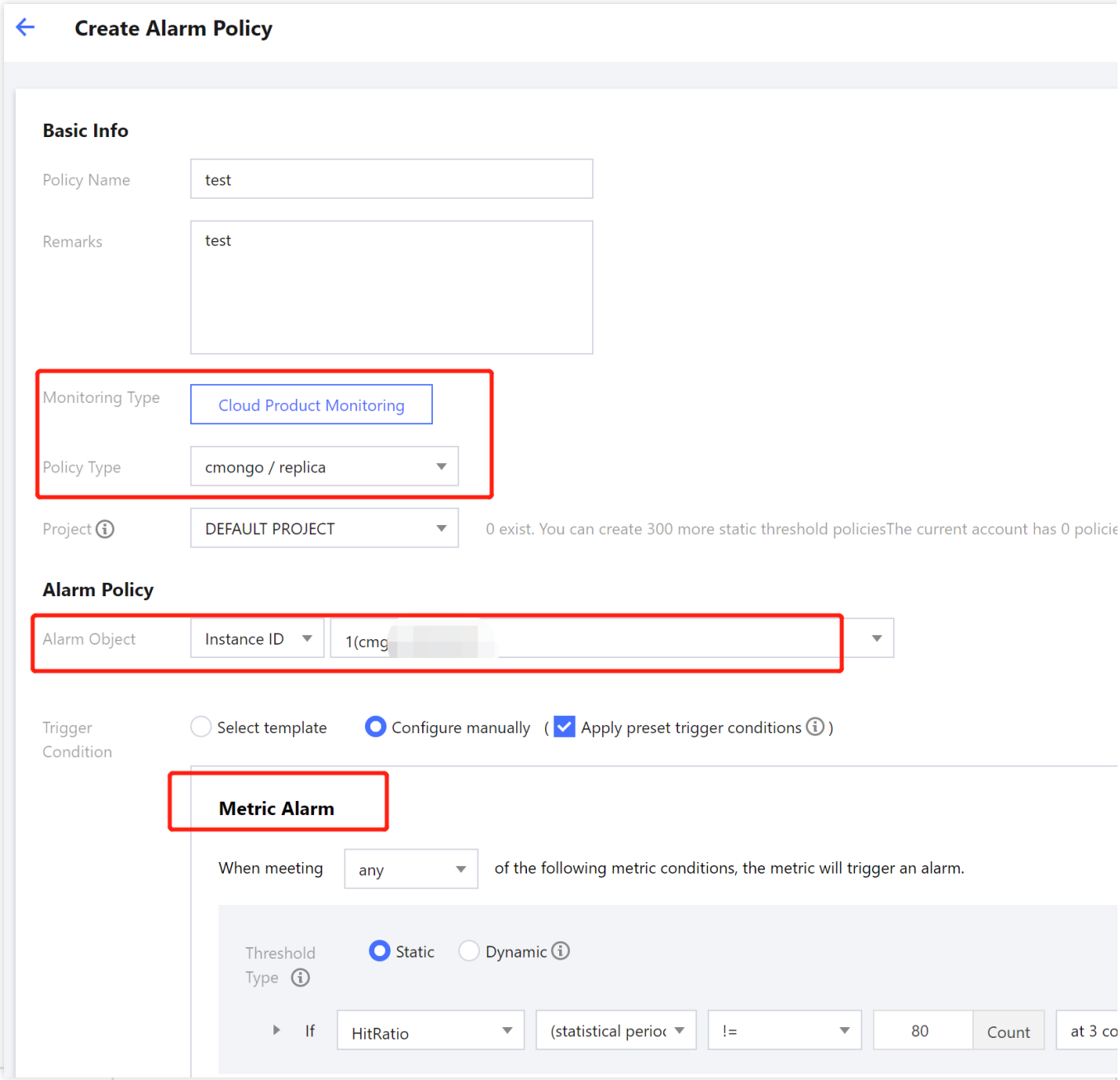
in the **Monitoring/Status** column and click **Configure Alarms** in the top-right corner of the instance monitoring dashboard.



Click the instance ID in blue to enter the **Instance Details** page. Then, select the **System Monitoring** tab and click **Configure Alarms**.



6. On the **Create alarm policy** page, configure a new alarm policy as shown below. For more information on the basic concepts of alarm policy, see Creating Alarm Policy.



Parameter	Description
Policy Name	Customize the alarm policy name for easier identification.
Remarks	Briefly describe the alarm policy for easier identification.
Monitoring Type	Select **Cloud Product Monitoring**.
Policy Type	Set **Policy Type** to **TencentDB/MongoDB/instance**, **TencentDB/MongoDB/node**, or **TencentDB/MongoDB/replica set**.
Project	Specify a project for the alarm policy. You can quickly locate all alarm policies of a project in the

	alarm policy list.
Alarm Object	If you select **Instance ID**, the alarm policy will be bound to the specified database instance. If you select **Instance Group**, the alarm policy will be bound to the specified database instance group. For more information on how to create an instance group, see Instance Group. If you select **All Objects**, the alarm policy will be bound to all instances the current account has permission on. If you select **Tag**, the alarm policy will be bound to all instances associated with the current tag key and tag value.
Trigger Condition	**Select template**: You can select a template file in the drop-down list, and alarms will be reported based on the trigger conditions preset in the template. For specific configurations, see Configuring Trigger Condition Template. **Configure manually**: You need to configure the threshold for triggering an alarm for each metric in the **Metric Alarm** section below. Alarm threshold type has the following values: If you select **Static**, you can manually set a fixed threshold, and alarms will be triggered when the threshold is reached. If you select **Dynamic**, exceptions will be determined based on the dynamic threshold boundaries calculated by machine learning algorithms. For more information, see Creating Alarm Policy.
Alarm Notification	You can select a preset or custom notification template. Each alarm policy can be bound to three notification templates at most. For more information, see Alarm Notification.

7. After confirming that the configuration is correct, click **Complete**. For more information on alarms, see Alarm Overview.

Related APIs

API Name	Description
CreateAlarmPolicy	Creates an alarm policy in TCOP

Configuring Event Alarms

Last updated：2024-05-07 12:37:13

Overview

TencentDB for MongoDB has been integrated with Tencent Cloud Observability Platform, supporting the reporting of Tencent Cloud Observability Platform events. All Tencent Cloud Observability Platform events will be automatically delivered to Tencent Cloud's EventBridge in the Cloud Service Event Bus. Tencent Cloud Event Bridge (EventBridge) is a secure, stable, and efficient serverless event management platform. An event is a data record of a status change. Releasing an event from the event source to EventBridge needs to comply with CloudEvents specifications. For more information about the specifications, see CloudEvents 1.0.

Event Target

An event rule can have multiple event targets. Before creating an event rule, first plan the event target types. Event Bridge currently supports the following **event targets:**

Message Push (only supports rules in the cloud service event bus)

CLS Log

Serverless Cloud Function (SCF)

Ckafka

TencentDB for MongoDB Event

Event Chinese Name	Event English Name	Event Type	Subordinate Dimension	Recovery Concept Availability	Event Description	Soluti Sugge
Insufficient Backup Oplog	oplogInsufficient	Exception event	Instance dimension	No	During the backup process of TencentDB for MongoDB, it is impossible to read the complete oplog from the last backup to the current backup,	It is recom to adj capac of the Tence for Mc oplog freque backu the Mc conso specif operat

					which will affect your ability to rollback the database to any point within 7 days.	see A Oplog Storag Capac
Number of Connections Exceeding Limit	connectionOverlimit	Exception event	Instance dimension	Yes	Number of instance connections exceeds the maximum limit.	Increa maxim numbe conne or rest instan specif operat see S for Ex Conne Limit. For da perfor optimi see A and Resol Metho Abnor High Conne Usage
Primary-Replica Switch	primarywitch	Exception event	Instance dimension	Yes	Instance primary node is abnormal, and switching with the secondary node occurs. This event may be triggered in case of physical	Confir wheth instan status norma

					server failure.	
Disk Space About to Run out	instanceDiskSpaceLow	Exception event	Instance dimension	Yes	The disk space is about to be filled, which may cause the instance to become read-only.	Clean space specif operat see S for Hig Space
Instance Rollback	instanceRollback	Exception event	Instance dimension	Yes	Instance data rollback. When some data on the primary node has not been timely synchronized to the secondary node, failure of the primary node concurrency leads to primary-replica switch, which may trigger this event.	Confir wheth instan status norma
Node CPU Exception	NodeCPUAbnormal	Exception event	Instance dimension	Yes	If any node in the cluster reaches 80% CPU usage, an alarm is immediately triggered.	For sp operat see S for Hig Usage

Billing Overview

Tencent Cloud offers EventBridge as a **pay-as-you-go** service. For more information, see EventBridge > Product Pricing.

Type	Pay-as-You-Go
Payment Method	Settlement is based on the number of events actually delivered to the event bus, calculated hourly.
Billing Unit	CNY/Million Events
Usage Scenario	For applications with low or high fluctuating message volumes, it can effectively avoid resource waste.

Directions

1. Log in to the EventBridge console. In the left sidebar, choose Event Rule.

2. At the top of the page on the right, under the **Region**, select **Guangzhou**, then select **Event Set** from the dropdown list and select **default** .

Note:

The cloud service event bus collects Monitor and Audit events generated by Tencent Cloud services across all regions. It is created by default in Guangzhou and cannot be deleted.

In the left sidebar, choose **Event Set**. In the Event Set list, click **Default** to view the **default** event bus which already includes TencentDB for MongoDB. For specific operations, see Tencent Cloud Service Event Source.

3. On the **Event Rule** page, click **Create**. On the **Event Pattern** navigation page, configure the page parameters according to the parameter explanations shown in the following table.

Interface Area	Interface Parameter	Parameter Explanation
Basic Information	**Region**	The region where the event rule is created.
	Event Bus	Information about the event bus to which the event rule belongs.
	Rule Name	Set the name of the event rule. It can only contain letters, numbers, underscores, hyphens, must start with a letter, and end with a number or letter, between 2 and 60 characters.
	Rule Description	A brief description of the event rule.
	Tag	Set the tag key and value for the event.
	Data Conversion	Check if data conversion is needed.

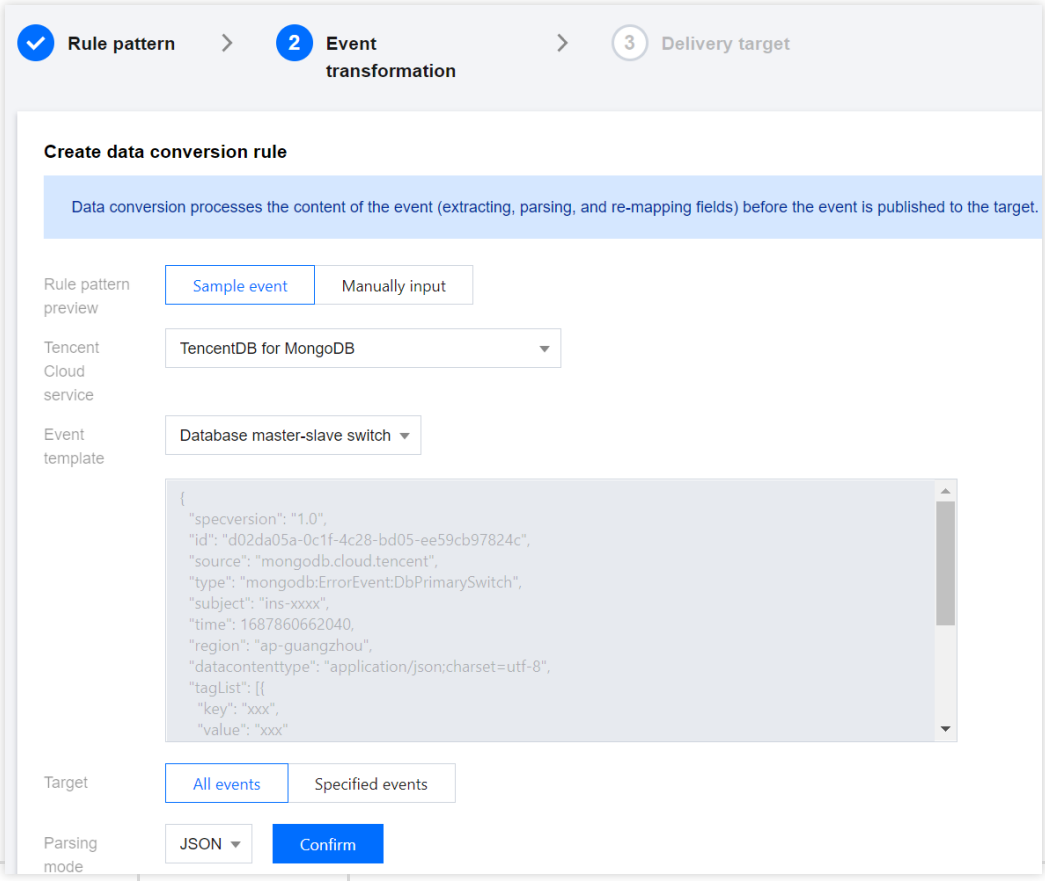
Event Example	**Event Example Selection**	In the dropdown list, you can search for MongoDB and view the relevant examples of MongoDB events.
Event Matching	**Writing Pattern**	**Form Pattern**: This pattern allows you to select **cloud service type** and **event type**, providing event matching rules. **Custom Event**: Define event matching rules in the input box below. For rule writing instructions, click **Rule Writing Guidelines**.
	Cloud Service Type	When **Writing Pattern** is set to **Form Pattern**, this parameter is shown. In the dropdown list, select **TencentDB for MongoDB**.
	Event Type	When **Writing Pattern** is set to **Form Pattern**, this parameter is displayed. In the dropdown list, select the supported event types.
	Event Matching Rule Preview	Preview the generated event matching rules.

4. click **Test Matching Rules** to test the defined event matching rules. After passing the test, click **Next**. If the test fails, correct it according to the prompt information.

5. (Optional) If you need to convert the data format, the **Event Transformation** page is displayed as shown in the following figure. Configure the format and fields for data transformation according to the parameter explanations in the table below.

Note:

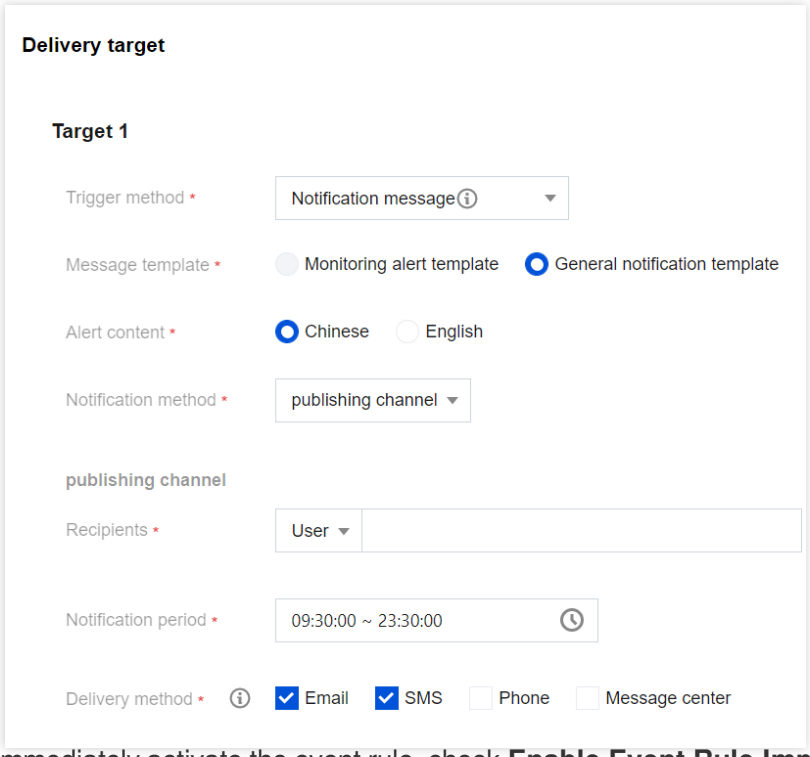
Data conversion offers a simple data processing feature. By the input data and configured items, it can carry out data formatting, return the processed structured data, distribute to downstream targets, and create a bridge between data sources and data processing systems.



Interface Area	Interface Parameter	Parameter Explanation
Create New Data Conversion	**Event Pattern Preview**	By selecting **Sample Event**, you can use the event template; By selecting **Manually Input**, you can define the event fields in the input box below.
	Event Template	For **Event Pattern Preview**, select **Sample Event** to display the parameter. In the dropdown list, you can search for MongoDB and select a template for MongoDB events. In the input box below, the specific field information of the event template will be displayed.
	Conversion Target	**Complete Event**: Routes the complete structure of event fields to the event target. **Partial Event**: Event Bridge uses **JSONPath** configuration to extract event fields and routes the specified event fields to the event target.
	JSONPath	When you select partial events for the conversion target, this parameter is displayed. Enter the event fields you wish to convert in the input box.
	Parsing Mode	Select a parsing mode. Supports JSON, separators, and regular expression extraction.
	Parsing Result	Click **Confirm** after the parsing mode to start parsing data, converting event rules into a Key-Value format.

	Filter	Configure the filter to only output data that meets the filter rules.
	Data Processing	For the current parsed data, select the data type in the **TYPE** column.
	Test Results	Click **Testing** to perform a validity check and output the final conversion result.
Failure Information Handling	**Dead Letter Queue**	Configure whether to deliver messages that fail to be processed properly to CKafka's dead letter queue.
	Delivery Type	The delivery type for failed messages is set to CKafka.
	CKafka Instance	Select the instance ID of the CKafka instance to which failed messages will be delivered.
	CKafka Topic	Choose the topic within the selected CKafka instance to which failed messages will be delivered. CKafka uses the concept of topics externally. Producers write messages to a topic, and consumers read them from it.

6. Click **Confirm** after the parsing mode to begin data parsing. Wait until the completion of data parsing. Set filter rules and data processing methods. For specific operations, see Configuring Data Conversion.
7. Click **Next** to select the event target bound to this rule. You can deliver the collected events to the specified delivery target for completing processing and consuming. The following figure takes the **Trigger Method** as **Message Push** as an example. To configure event alarm push, see Configuring Push Target.



8. To immediately activate the event rule, check **Enable Event Rule Immediately**, and click **Complete**.

Event Rule Related Interface

Interface Name	Interface Feature
CheckRule	Validation Rules
CreateRule	Create Event Rule
DeleteRule	Delete Event Rule
GetRule	Retrieve Event Rule Details
ListRules	Retrieve Event Rule List
UpdateRule	Update Event Rules

Backup and Rollback
Data Backup

Last updated：2024-01-15 14:40:06

To avoid data loss caused by system crashes or other problems, TencentDB for MongoDB supports data backup and rollback after system recovery to ensure data integrity.

Overview

Backup types

Automatic backup: Data is automatically backed up as scheduled based on the system's default backup policy (such as default backup interval and mode).

Manual backup: You can run a backup task at any time to meet your business Ops and troubleshooting requirements.

Backup modes

Physical backup: In this mode, physical database files in an instance are backed up, which is fast and easy to restore with a high success rate. However, it has no portability, and the backup environment and restoration environment must be completely the same.

Logical backup: In this mode, the database instance is connected to, and the mongodump tool is used to save the operation logs to a logical backup file to back up the data, which can be restored by replaying the operation logs. This mode is slow but has a high portability. You can restore the logical backup of a database to database on different versions.

Use Limits

A backup can contain up to 7 days of continuous data; that is, you can roll back data to any time point in the last 7 days.

Note

Instance backup doesn't affect your business.

Backup files are stored in COS without using the storage space of TencentDB for MongoDB instances. For more information on COS, see Cloud Object Storage (COS).

Version Description

Version	Instance Type	Automatic backup	Manual backup
v3.2	Replica Set	Default backup mode: Logical backup Supported backup modes: Logical backup	Default backup mode: Logical backup Supported backup modes: Logical backup
	Sharded cluster	Default backup mode: Logical backup Supported backup modes: Logical backup	Default backup mode: Logical backup Supported backup modes: Logical backup
v3.6	Replica Set	Default backup mode: Logical backup Supported backup modes: Logical backup	Default backup mode: Logical backup Supported backup modes: Logical backup
	Sharded cluster	Default backup mode: Logical backup Supported backup modes: Logical backup	Default backup mode: Logical backup Supported backup modes: Logical backup
v4.0 and later	Replica Set	Default backup mode: Logical backup Supported backup modes: Logical backup and physical backup	Default backup mode: Logical backup Supported backup modes: Logical backup and physical backup
	Sharded cluster	Default backup mode: Logical backup Supported backup modes: Logical backup and physical backup	Default backup mode: Logical backup Supported backup modes: Logical backup and physical backup

Billing Overview

Currently, backup is free of charge. We will notify you when billing for the backup space officially starts.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.

The TencentDB for MongoDB replica set or sharded cluster instance is in **Running** status.

Adjusting the automatic backup policy

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. Select the **Backup and Rollback** > **Backup Task List** page.
7. Select the **Auto-Backup Settings** tab and click **Edit**.
8. Edit **Backup Mode** and **Backup Start Time** based on the parameter descriptions in the following table.
9. Click **Save**.

Parameter	Note
Data Backup Retention	Data backup files can be retained for **7** days by default.
Backup modes	(Optional) Select the backup mode. TencentDB for MongoDB 3.6 replica set instances don't support this parameter.
Backup Start Time	The default start time is 10:00 PM–02:00 AM; that is, the system starts the backup task between 10:00 PM and 02:00 AM every day. You can select a different time period to start data backup as needed by your business. The specific start time varies by the specific scheduling of the backup task.

Manual Backup

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. In the top-right corner of the **Instance Details** page, click **Manual Backup**.
7. (Optional) Select the backup mode. TencentDB for MongoDB 3.6 replica set instances don't support this parameter.
8. Add remarks and click **OK**.

Downloading a backup file

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. Select the **Backup and Rollback** > **Backup Task List** page.
7. In the **Backup Task List**, find the target file and click **Download** in the **Operation** column.
8. In the **Generate Backup File** pop-up window, read the backup note carefully and click **OK**.
9. Select the **File Download List** tab and view the backup task progress.
10. After the task execution is completed, you can back up the data to your local device and view it as follows:

Over public network: Click **Download from Public Network** in the **Operation** column and directly use the browser to download the backup to your local device.

Over private network: Copy the private network address and run a `wget` command `wget -c 'private network address' -O backup.tar` in a CVM instance to download the backup at a high speed over the private network. For detailed directions on how to log in to CVM, see Customizing Linux CVM Configurations.

Related APIs

API Name	Description
DescribeDBBackups	Queries the list of backups of an instance
CreateBackupDBInstance	Backs up an instance
DescribeBackupDownloadTask	Queries the information of a backup download task
CreateBackupDownloadTask	Creates a backup download task

Data Rollback
Data Clone

Last updated：2024-05-07 17:30:35

Operation scenarios

When the current instance data encounters severe issues and needs to be rolled back to a previously backed-up state, you can directly clone a new instance from the current instance's backup file to quickly restore data. The data of the cloned instance is consistent with the backup file, allowing you to use the cloned instance to analyze historical data, or swap the IP of the cloned new instance with the original instance to achieve a rollback. This method avoids the tedious process of manually restoring data one by one, improving the efficiency and accuracy of data recovery.

Prerequisites

You have applied for a TencentDB for MongoDB Instance.
The TencentDB for MongoDB Replica Set Instance or Sharded Instance is in Running status.
You have Data Backup.

Operation step

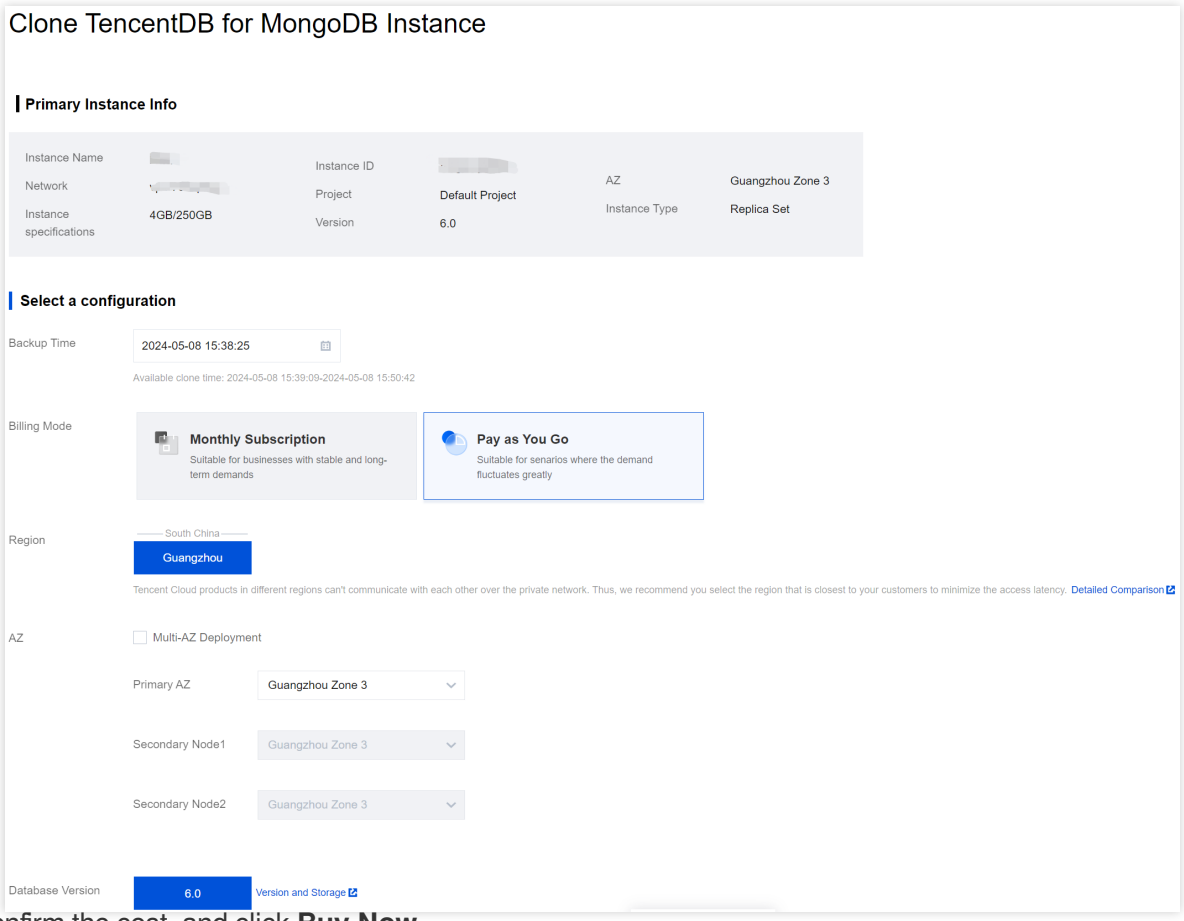
1. Sign in to the MongoDB Console.
2. In the MongoDB drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Instance**. The directions for the two types of instances are similar.
3. Above the instance list on the right, select the region.
4. In the instance list, find the target instance.
5. Click target Instance ID to enter the **Instance Details** page.
6. Select the **Backup and Rollback** tab, and enter the Backup Task List page.
7. In the Backup Task List, find the backup file to be restored.
8. In the **Operation** column, click **Clone**.

9. On the **Clone TencentDB for MongoDB Instance** page, confirm the master instance information, in the time frame of selecting **Backup Time**, choose the rollback point in time, select the new instance's billing model, configure

specifications, and purchase the new instance. For more information on how to configure additional parameters, see Create MongoDB Instance.

Note:

Rollback time only supports selecting data from any time point within the last 7 days before the current time.



10. Confirm the cost, and click **Buy Now**.

11. Return to the Instance List page. Once the instance has been created and the data from the source instance has been synchronized with the newly cloned instance, the new instance can be used. You may achieve the purpose of rolling back data by swapping the IP of the cloned new instance with the original instance.

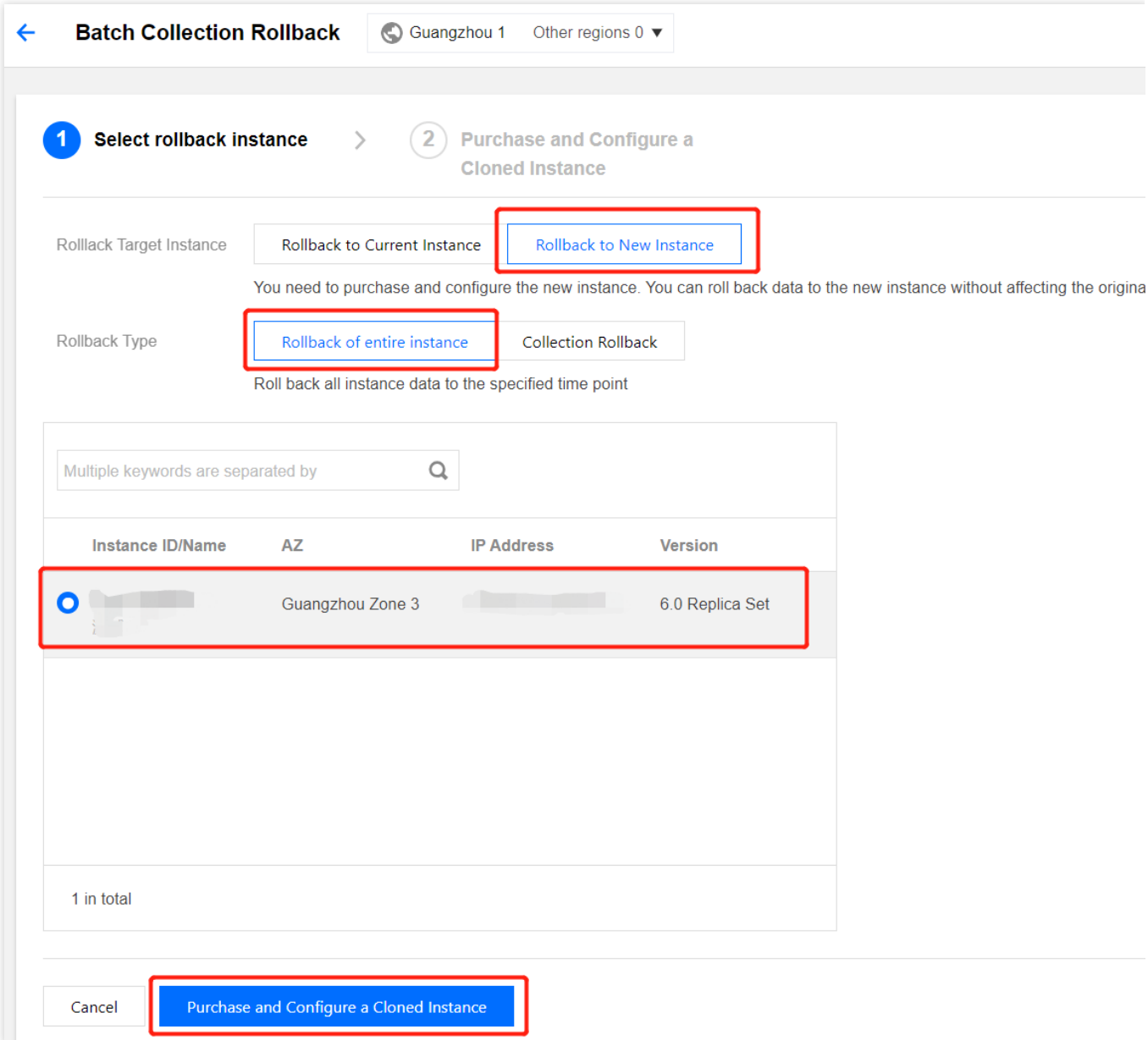
Note:

After the instance is cloned, the source instance can be retained or terminated based on your own needs.

More Entrances

1. In the **Backup and Rollback** tab's backup task list, find the backup file you need to restore.
2. In the **Operation** column, click **Collection Rollback**.
3. In the **Batch Collection Rollback** Configuration Wizard's **Select rollback Instance** tab, for **Rollback Target Instance** select **Rollback to New Instance**, for **Rollback Type** choose **Rollback of entire instance**.

4. In the instance list below, check an instance that is to be rolled back (you can search by Instance ID, Instance Name, or IP Address in the search box).



5. Click **Purchase and Configure a Cloned Instance**, enter the **Clone TencentDB for MongoDB Instance** page, confirm the primary instance information, choose the billing model, configuration specifications, and purchase the new instance. For more information on how to configure additional parameters, see Creating TencentDB for MongoDB Instance.

6. Confirm the cost, and click **Buy Now**.

Database and Table Rollback

Last updated：2024-05-07 09:58:01

Overview

When the business only requires restoration operations on multiple databases and tables, **Database and Table Rollback** can be performed through the console, restoring the data to either the existing instance or a new one. Compared to a rollback of entire instance, a database or table rollback involves less data and results in a quicker process.

Version Description

Version	Rollback Method
3.2, 3.6	Rollback of entire instance (logical backup) Collection rollback (logical backup)
4.0, 4.2, 4.4	Rollback of entire instance (logical backup, physical backup) Collection rollback (logical backup, physical backup)
5.0	Rollback of entire instance (logical backup, physical backup) Collection rollback (logical backup) **Note：** Version 5.0 instances temporarily do not support a database or table rollback to a new instance. Flashback by Key (Logical Backup)

Use Limits

You can select up to 2,000 collections per instance to roll back.
You can roll back data to any time point in the last 7 days.
Note:
Pay close attention to the **Oplog Time Lag** monitoring metric found within the **System Monitoring** section of the instance management page. In scenarios where your business experiences frequent write, update, and delete operations, the smaller this metric becomes, the higher the likelihood that the oplog will be at risk of being overwritten. If there are transaction operations on the client during the rollback process, you need to actively submit the transaction or set a timeout period to prevent the transaction from occupying lock resources for a long time and causing the

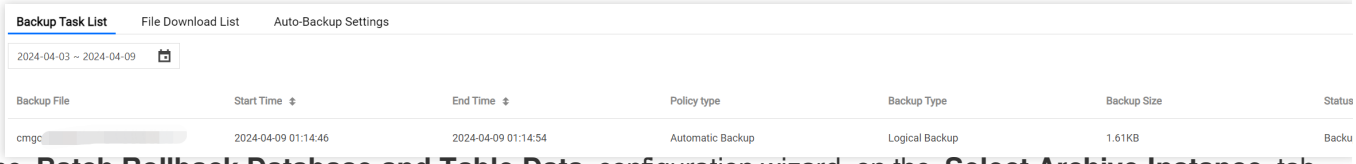
rollback task to be abnormal.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The status of the TencentDB for MongoDB instance is **Running** .
You have backed up the data. For more information, see Data Backup.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. On the **Instance Details** page, select the **Backup and Rollback** tab.
6. Navigate to the **Backup and Rollback** tab and access the **Backup Task List** page.
7. In the **Backup Task List**, find the backup file to be restored.
8. In the **Operation** column, click **Database and Table Rollback** .



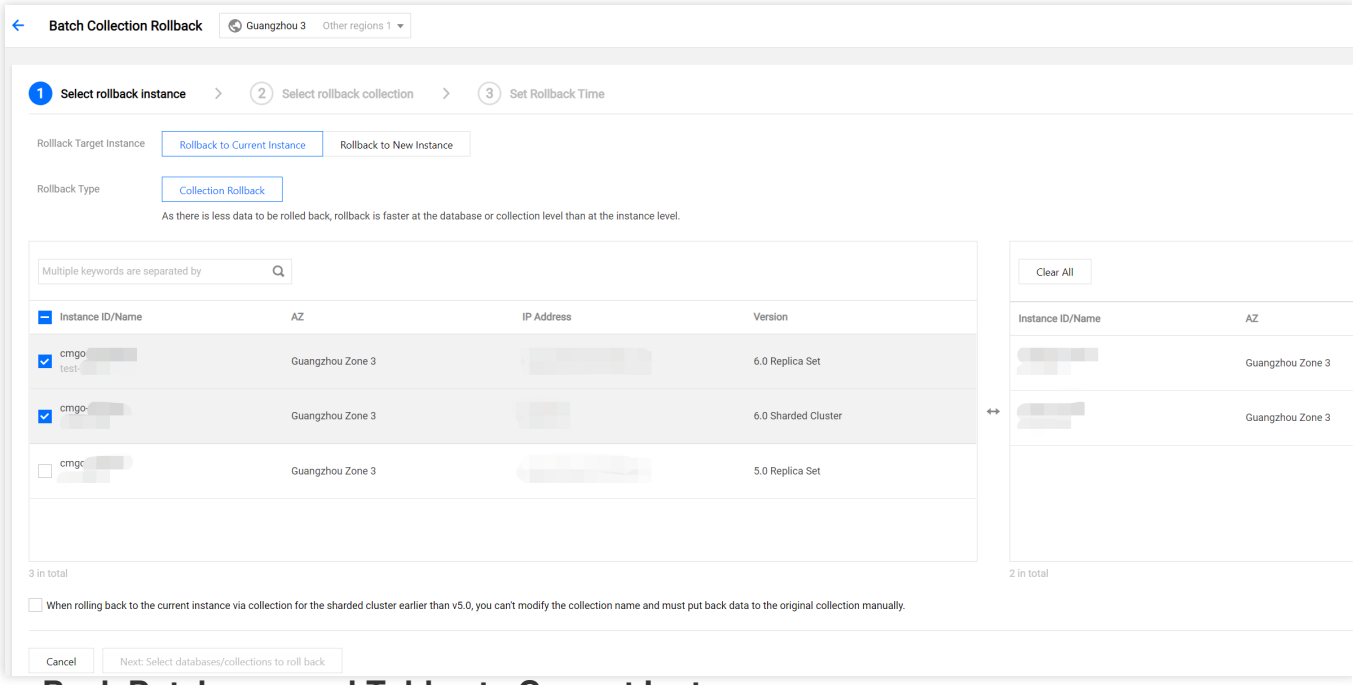
9. In the **Batch Rollback Database and Table Data** configuration wizard, on the **Select Archive Instance** tab, after the **Rollback Target Instance** , select either **Rollback to Current Instance** or **Rollback to New Instance** . Within **Select Rollback Type** , select **Database and Table Rollback** .

Note：

Rollback to the Current Instance eliminates the need to purchase a new instance by restoring databases and tables to the current instance. It supports the selection of multiple instances for batch rollback, allowing for database and table rollback or key-based flashback according to the actual scenario. In the instance list below, check one or more instances to be rolled back (you can search by instance ID, instance name, or IP address in the search box). For detailed operations, see Database and Table Rollback to the Current Instance.

Rollback to a New Instance requires the purchase of a new instance and does not affect the source instance. It does not support selecting multiple instances for batch rollback. Depending on the actual scenario, you can choose to perform a database and table rollback, a key-based flashback, or clone an instance. In the instance list below, check

only one instance to be rolled back (you can search by instance ID, instance name, or IP address in the search box). For specific operations, see Rolling Back Databases and Tables to a New Instance.



Rolling Back Databases and Tables to Current Instance

1. Click **Next: Select Databases and Tables to Roll Back** . On the **Select Rollback Database and Table** tab, select the databases and tables to be rolled back. Confirm the database and table information in the right-side box zone. As shown below.

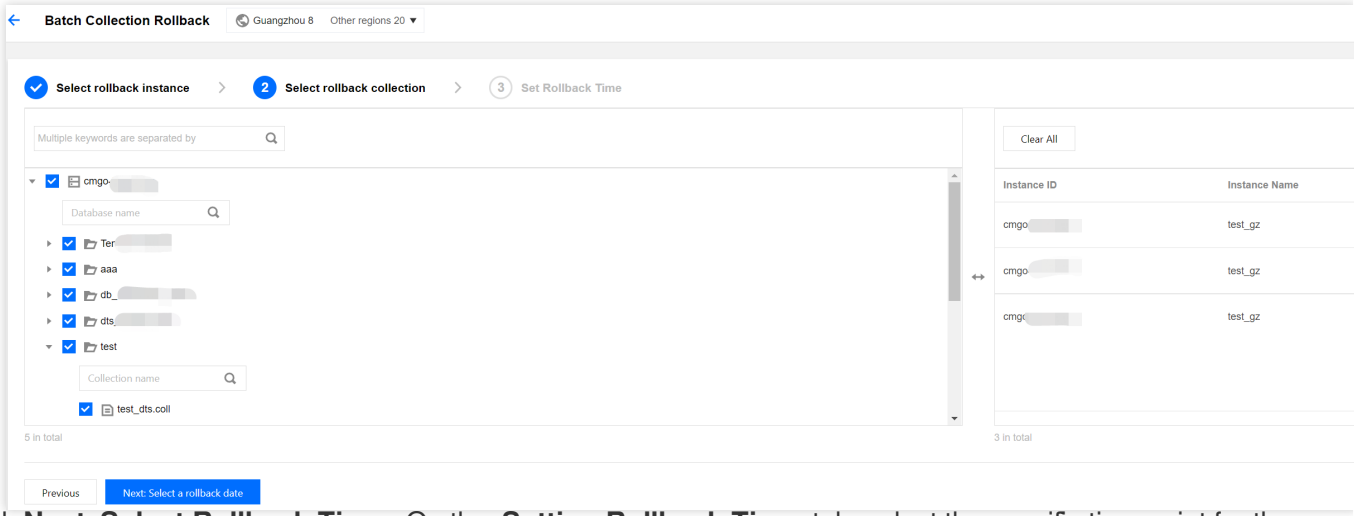
Note:

In the right-side zone, you can confirm and modify the selected databases and tables.

click **Clear Selection** to clear the selected databases and tables if the wrong selection is made.

click

to delete the selected databases and tables one by one.



2. Click **Next: Select Rollback Time** . On the **Setting Rollback Time** tab, select the specific time point for the rollback from the time frame following **Setting Rollback Time** . Confirm the instance information and the database and tablesinformation to be pre-rolled back.

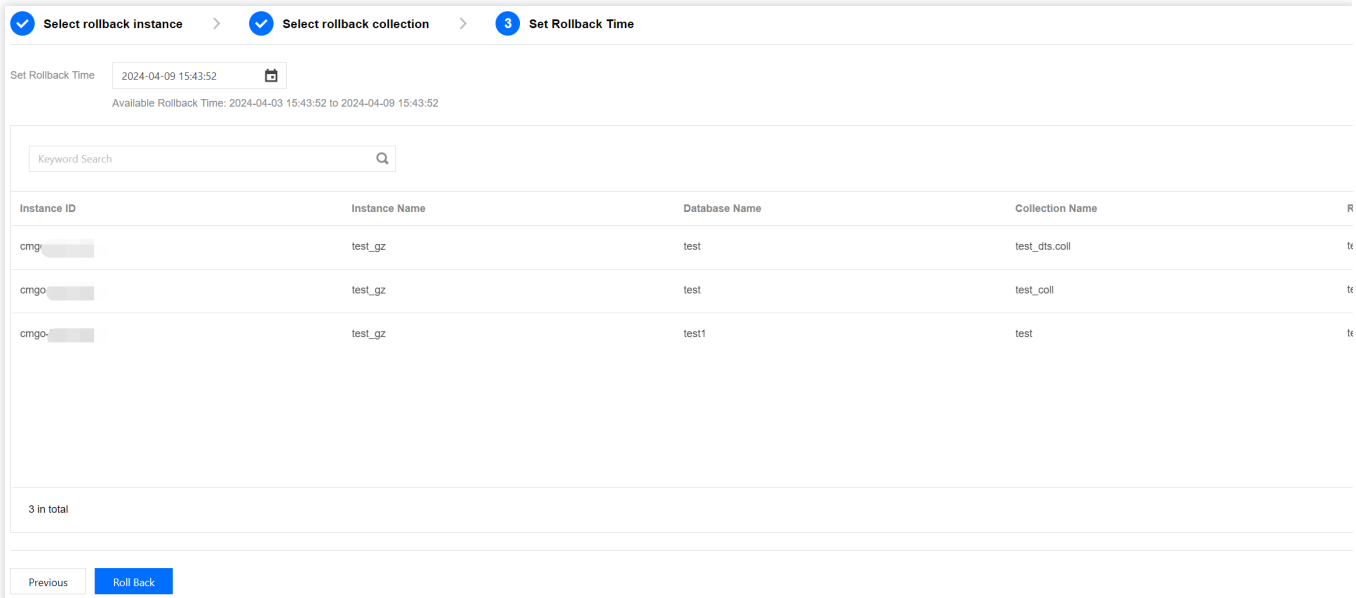
Note:

Rollback time only supports selecting data from any time point within the last 7 days before the current time.

Rollback to the current instance does not rollback the original databases and tables directly but creates a new backup file. For instance, if the source database or table is test, a new database or table test_bak will be created. As shown below, **Rollback Database and Table Names** are the names of the newly created databases and tables.

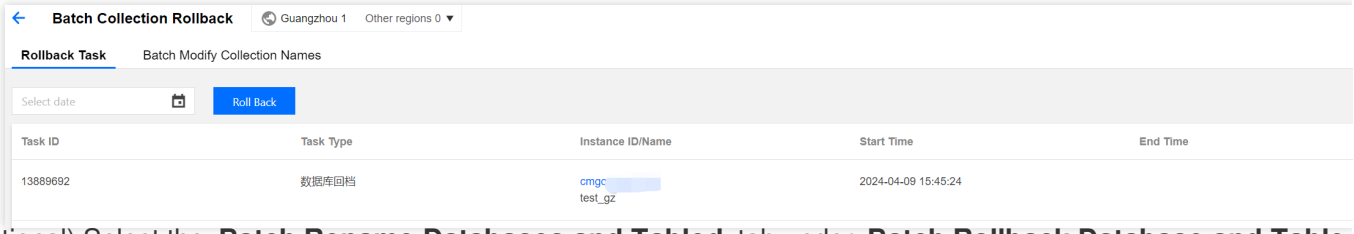
After the rollback task is completed, you can batch modify database and table names as needed.

In versions of TencentDB for MongoDB below 5.0, sharded clusters rolling back to the current instance cannot modify database and table names. Data must be manually replaced back into the original databases and tables.



3. Click **Start Rollback** . Return to **Rollback Task** tab of the **Batch Rollback Database and Table Data** . You can see ongoing rollback tasks. Click the **Operation** column's **Task Details** to view the detailed information of the

task. Wait until the task is completed. Then you can connect to the instance to confirm the correctness of the rolled-back data.



4. (Optional) Select the **Batch Rename Databases and Tabled** tab under **Batch Rollback Database and Table Data** . Find the completed rollback task. In its **Operation** column, click **Batch Modify Database and Table Names** to view information about the databases and tables to be modified on the right zone. The information includes original database and table names, new original database and table names, rollback database and table names, and new rollback database and table names.

Click

to download information of the databases and tables to be modified. Then you may view locally.

Confirm the changes. Click **Batch Rename Databases and Tables** at the bottom left to complete the modifications.

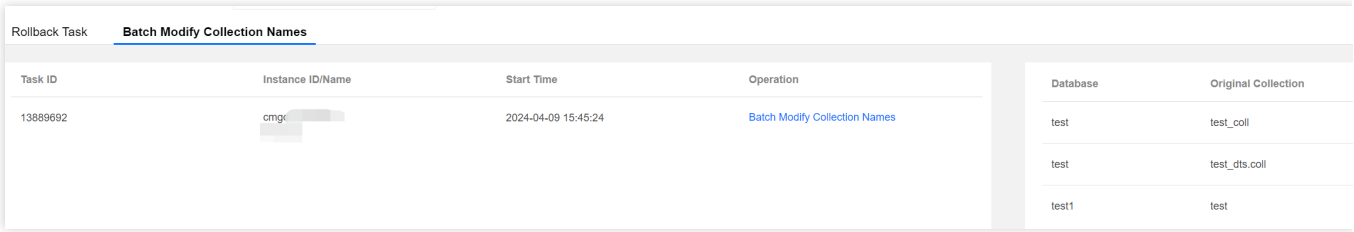
Note:

Batch Rename Databases and Tables can only modify all databases and tables under a single instance of a single rollback task. If a user initiates a batch rollback task and rolls back databases and tables of multiple instances, the database and table names need to be modified one by one. For specific operations, see Batch Rollback.

Batch Rename Databases and Tables includes changing the original database and table names and rollback database and table names.

For original databases and tables, the mark _ori is added to their original names.

For rollback databases and tables, change the rollback database and table names to the original ones.



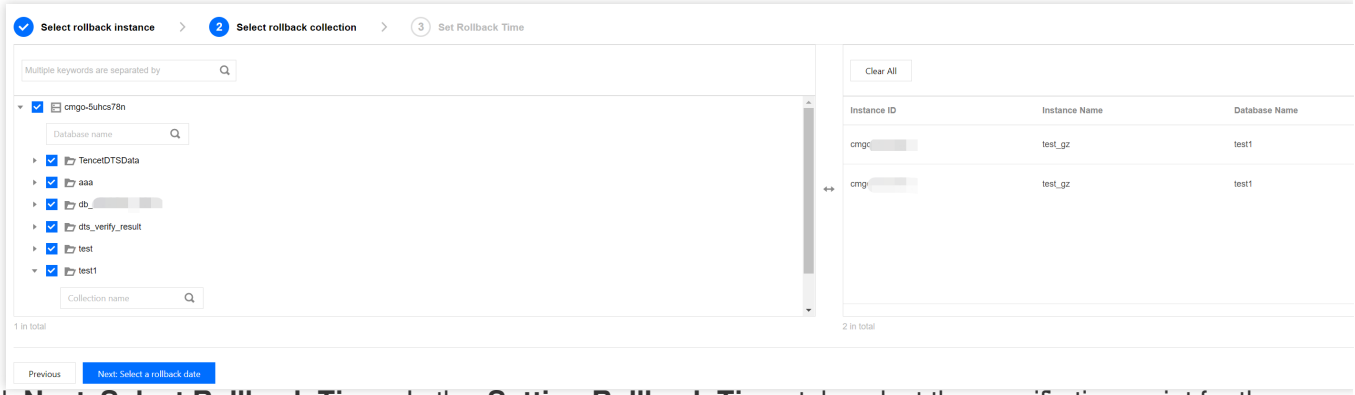
Rolling Back Databases and Tables to a New Instance

1. Click **Next: Select databases and tables to roll back** . On the **Select Rollback Database and Table** tab, choose the databases and tables of the source instance to be rolled back. In the search box, you can search for the databases and tables to be rolled back based on the database and table names. And in the right box zone, you can view the selected database and table information. As shown below. In the right box zone, you can manage the selected databases and tables.

Click **Clear Selection** to clear the selected databases and tables if the wrong selection is made.

Click

to delete the selected database and table one by one.



2. Click **Next: Select Rollback Time** . In the **Setting Rollback Time** tab, select the specific time point for the rollback from the time frame provided under **Setting Rollback Time** . Confirm the instance information and the database and table information to be pre-rolled back.

3. Click **Proceed to Purchase and Configure the Replica Instance** . Enter the **TencentDB for MongoDB clone instance** page and select the billing mode, configuration specifications, etc., for the new instance. For more information, see Create MongoDB Instance.

4. Confirm the cost, and click **Purchase Now** .

5. Return to the instance list page. Once the instance has been created, the source instance's databases and tables have been synchronized to the newly purchased clone instance. Then you can connect to the new instance to confirm the correctness of the rollback data.

Batch Rollback

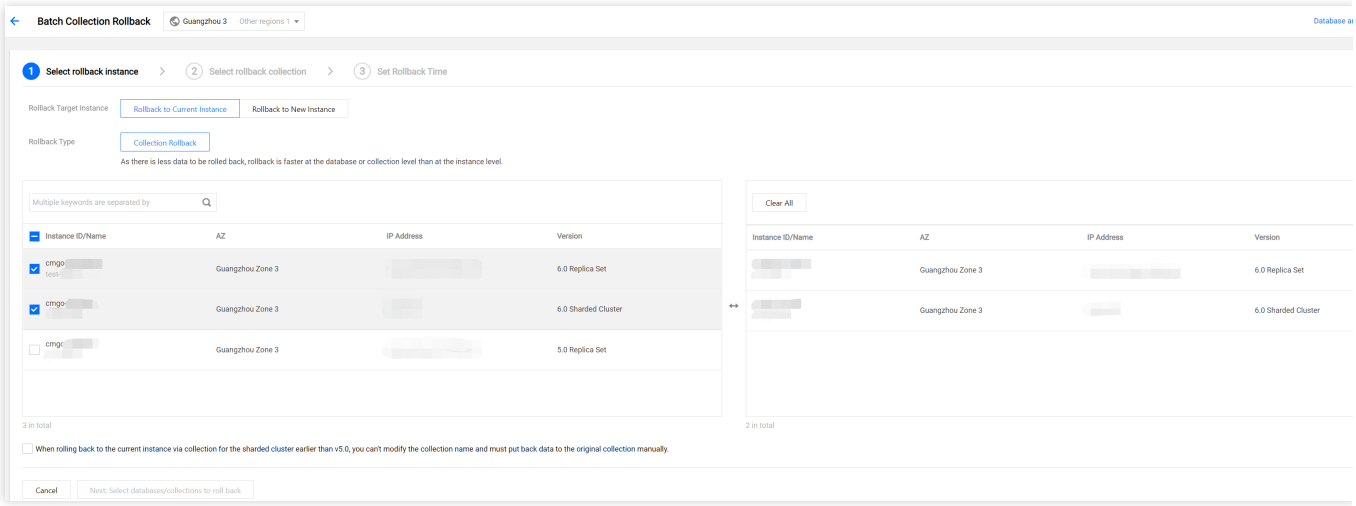
Last updated：2024-05-07 10:55:52

Overview

Batch rollback refers to the process of rolling back the database and table data of multiple instances at once, restoring the data of multiple instances to the source instance in a single batch operation. Database and table rollback or key-based flashback to the current instance supports batch operations, allowing multiple instances' databases and tables to be rolled back at once. The newly rolled back databases and tables are named with the suffix _bak. After the rollback is completed, the database and table name can be changed as needed to improve the efficiency and accuracy of data recovery and avoid the cumbersome process of manual individual recovery. Moreover, MongoDB supports viewing all batch rollback historical tasks under the current account. This helps you to quickly understand past operation records for convenient and unified operation and management.

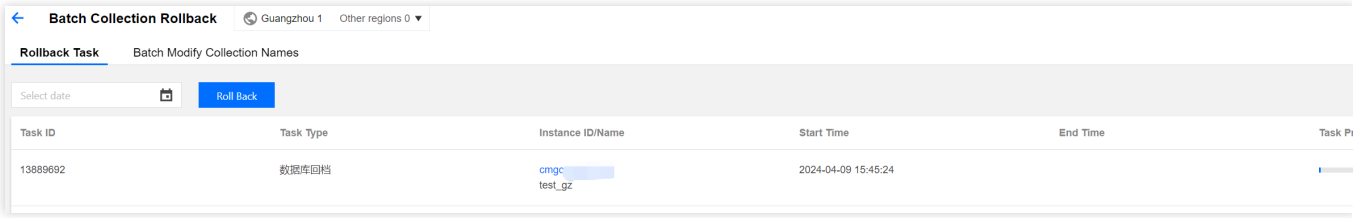
Initiating Batch Rollback Task

1. Log in to the MongoDB console.
2. In the **MongoDB** dropdown menu in the left sidebar, select **Batch Rollback.**
3. On the **Rollback Task** page, click **Initiate Rollback** to enter the **Batch Rollback Database and Table Data** configuration wizard. Here, you can configure the batch rollback database and table task and initiate a rollback task. For specific operations, see Database and Table Rollback.



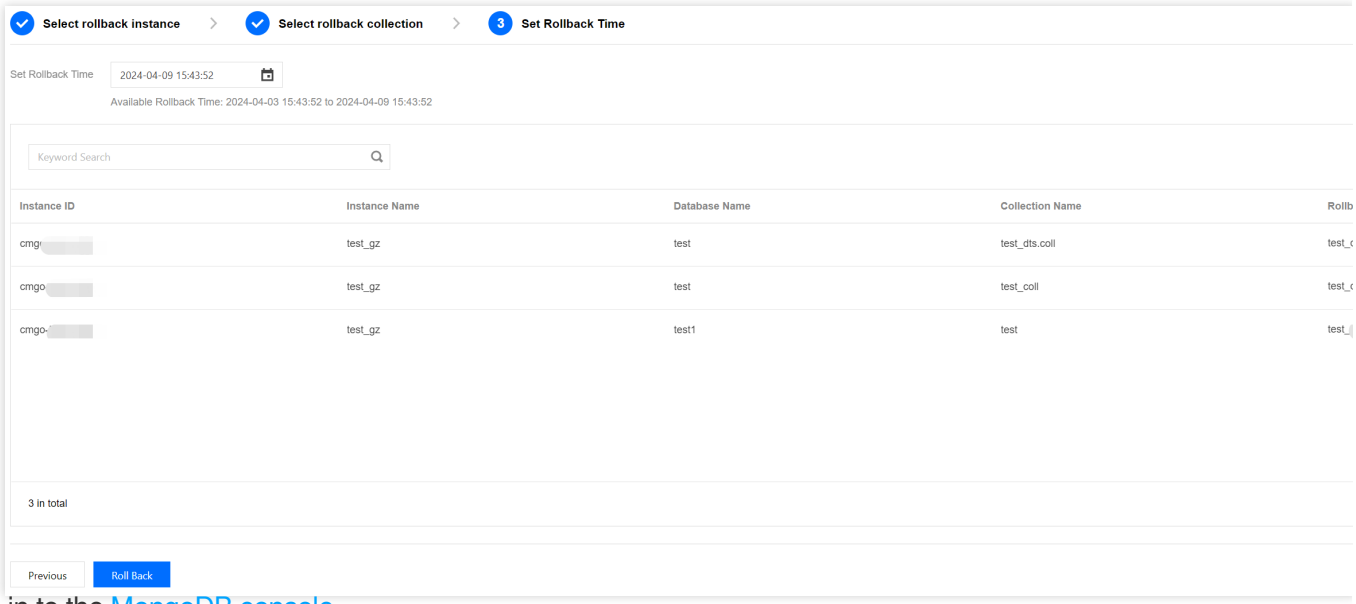
Viewing Batch Rollback Tasks

1. Log in to the MongoDB console.
2. In the **MongoDB** dropdown menu on the left sidebar, select **Batch Rollback.**
3. On the **Rollback Task** page, you can find all batch rollback tasks under the current account. As shown in the following figure. You can filter the tasks you want to view by selecting a time range in the time box.



Batch Modifying Rollback Database and Table Names

Database and table rollback does not directly roll back data to the original database and table, but instead creates a backup file. For example, if the original database or table is named test, a new database or table named test_bak will be created. As shown in the following figure, **Rollback Database or Table Name** is the name of the newly created database or table. After the rollback task is completed, you can batch modify the database and table names as needed.



1. Log in to the MongoDB console.
2. In the **MongoDB** dropdown menu on the left sidebar, select **Batch Rollback.**
3. On the **Batch Rename Database and Table** tab in **Batch Rollback Database and Table Data**, find the rolled-back tasks and select the instances with database and table names to be modified. In their **Operations** column, click **Batch Modify Database and Table Names** to see the information of the databases and tables to be modified on the right zone. The information includes original database and table names, new original database and table names, rollback database and table names, and new rollback database and table names. Confirm the modifications, click the

Batch Rename Database and Table at the bottom left to complete the modifications. As shown in the following figure.

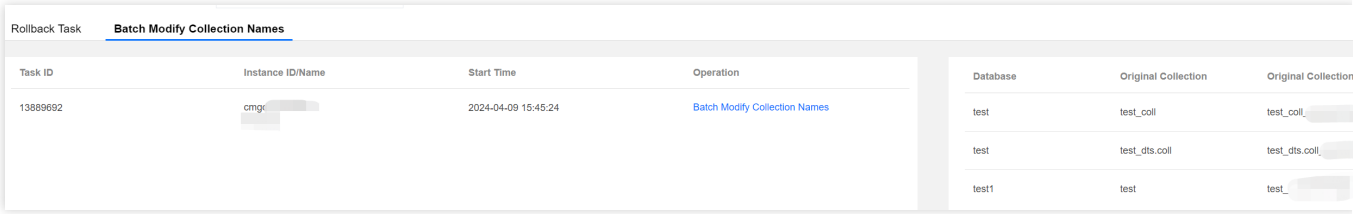
Note:

Batch Rename databases and tables can only modify all of them under a single instance of a single rollback task. If a user initiates a batch rollback task and rolls back databases and tables of multiple instances, the database and table names need to be modified one by one.

Batch Rename Database and Table includes changing the original database and tables names and rollback database and table names.

For original databases and tables, the mark _ori is added to their original names.

For rollback databases and tables, change their names to the original database and table names.



Restoring to Self-built Database

Last updated：2024-01-15 14:40:06

Restoring a Physical Backup to a Self-Built Database

A replica set instance has only one copy of data, while each shard of a sharded cluster has one copy of data. Restore your data based on your business needs. This document describes how to restore a single copy of data.

Restoring data to a single node

1. Copy the data to an empty data directory in the self-built database, such as /data/27017/.

```
cp -r * /data/27017/
```

2. Restart mongod and check the data. Below is a command sample:

```
./mongod --dbpath /data/27017 --port 27017 --logpath /var/log/mongodb/27017.log --f
```

Restoring data to a replica set

As a physical backup retains the configuration of the original instance by default, the original configuration needs to be removed first; otherwise, the data may become inaccessible.

1. Restore the data to a single-node self-built database and then restart the node in the form of replica set. Below is a command sample:

```
./mongod --replSet mymongo --dbpath /data/27017 --port 27017 --logpath /var/log/mon
```

2. Log in to the node and remove the replica set configuration of the original instance by running the following command:

```
rs.slaveOk()
use local
db.system.replset.remove({})
```

3. Restart the node, add a node to the replica set, initialize it, and check the data. The node added to the replica set should have been started with no data contained. Below is a command sample:

```
rs.initiate({"_id":"mymongo","members":[{"_id":0,"host":"127.0.0.1:27017"},{"_id":1
```

For more information on the `rs.initiate()` command, see rs.initiate().

Notes:

Data cannot be restored to a sharded cluster. As the route of physical backup is missing in a sharded cluster, mongos can only read the data of the primary shard even if the data of each shard is restored to the self-built replica set (each shard of the sharded cluster).

Restoring a Logical Backup to a Self-Built Database

To avoid any impact on data check after the data is restored to a self-built database, the self-built database must be empty.

For v3.6, you need to delete the `config` directory manually and then run the `mongorestore` command to restore the data of each shard.

```
[root@VM_0_5_centos 15452250299522893 95]# ll
total 16
drwxr-xr-x 2 root root 4096 Dec 25 10:38 admin
drwxr-xr-x 2 root root 4096 Dec 25 10:38 config
-rw-r--r-- 1 root root  668 Dec 25 10:38 oplog.bson
drwxr-xr-x 2 root root 4096 Dec 25 10:40 ycsb
[root@VM_0_5_centos 15452250299522893 95]# rm -rf config/
[root@VM_0_5_centos 15452250299522893 95]# ll
total 12
drwxr-xr-x 2 root root 4096 Dec 25 10:38 admin
-rw-r--r-- 1 root root  668 Dec 25 10:38 oplog.bson
drwxr-xr-x 2 root root 4096 Dec 25 10:40 ycsb
```

For v3.2, you need to merge all the files in individual collections manually before restoring the data. Below is a file merge operation sample:

The `c_10` collection is in the `ycsb` directory in the database and contains data files from `c_10.bson.gz.chunk-64` to `c_10.bson.gz.chunk-127`. The merge command is `cat c_10.bson.gz.chunk-* > ./c_10.bson.gz`.

Notes:

Chunk distinction will appear in some scenarios on v3.2.

Run the `mongorestore` command to restore the data, where the `-h` parameter specifies the self-built database address, `--dir` specifies the directory of the data file, and `--gzip` must be specified to decompress the backup file. The command is as follows:

```
./mongorestore  --gzip --drop -h127.0.0.1:27017 --dir ./1544517027220146694
```

Data Security
Configuring Security Group

Last updated：2024-01-15 14:40:06

You can configure a security group in the TencentDB for MongoDB console to control the outbound/inbound traffic.

Overview

Security group serves as a stateful virtual firewall with filtering feature for configuring network access control for one or more TencentDB instances. It is an important network security isolation tool provided by Tencent Cloud. Instances with the same network security isolation demands in one region can be put into the same security group, which is a logical group. TencentDB and CVM share the security group list and are matched with each other within the security group based on rules. For specific rules and limitations, see Security Group Overview.

Note:

TencentDB security groups currently only support network access control for VPCs but not the classic network.

As TencentDB doesn't have any active outbound traffic, outbound rules don't apply to it.

TencentDB for MongoDB security groups support primary instances, read-only instances, and disaster recovery instances.

TencentDB for MongoDB supports the security group feature which is implemented based on the allowlist. To use this feature, submit a ticket.

Directions

Step 1. Create a security group

1. Log in to the CVM console.

2. Select **Security Group** on the left sidebar, select a region above the instance list on the right, and click **Create**.

3. In the pop-up window, set the following configuration items, confirm that everything is correct, and click **OK**.

Template: Select a security group template in the drop-down list.

Open all ports: All ports are opened to the public and private networks. This may present security issues. Security group rules are added by default. You can click a security group template below to view its **Outbound Rules* and** Inbound Rules**.

Open ports 22, 80, 443, and 3389 and the ICMP protocol: Ports 22, 80, 443, and 3389 and the ICMP protocol are opened to the internet. All ports are opened to the private network. Security group rules are added by default. The port of TencentDB for MongoDB is 27017 by default. You can ignore this template.

Custom: You can create a security group and then add custom rules.

Name: Custom name of the security group.

Project: Select a project for easier management. By default, **Default Project** is selected.

Notes: A short description of the security group for easier management.

Advanced Configuration: You can add tags for the security group.

4. If you select **Custom** for **Template**, click **Set Now** in the **Note** window and perform the following steps.

Step 2. Set inbound rules in the security group

1. On the **Inbound Rule** tab of the **Security Group Rules** page, click **Add Rules**.

2. In the **Add Inbound Rules** pop-up window, set the rules.

Type: Select **Custom** as the default type.

Source: Set the source for database access, i.e., the inbound source. The following formats are supported:

Source Format	Format Description
CIDR notation	A single IPv4 address or an IPv4 range is represented in CIDR notation, such as `203.0.113.0`, `203.0.113.0/24`, or `0.0.0.0/0`, where `0.0.0.0/0` indicates all IPv4 addresses will be matched. A single IPv6 address or an IPv6 range is represented in CIDR notation, such as `FF05::B5`, `FF05:B5::/60`, `::/0`, or `0::0/0`, where `::/0` or `0::0/0` indicates all IPv6 addresses will be matched.
Security group ID	Reference a security group ID to match the IP address of the server associated with the security group.
Parameter template	Reference IP address object or IP address group object in a parameter template.

Protocol Port: Enter the protocol type and port for the client to access TencentDB for MongoDB. You can view the port information in the **Private Network Address** column in the instance list. The default port is 27017.

Policy: **Allow** or **Reject**. **Allow** is selected by default.

Allow: Traffic to this port is allowed.

Reject: Data packets will be discarded without any response.

Notes: A short description of the rule for easier management.

3. Click **Complete**.

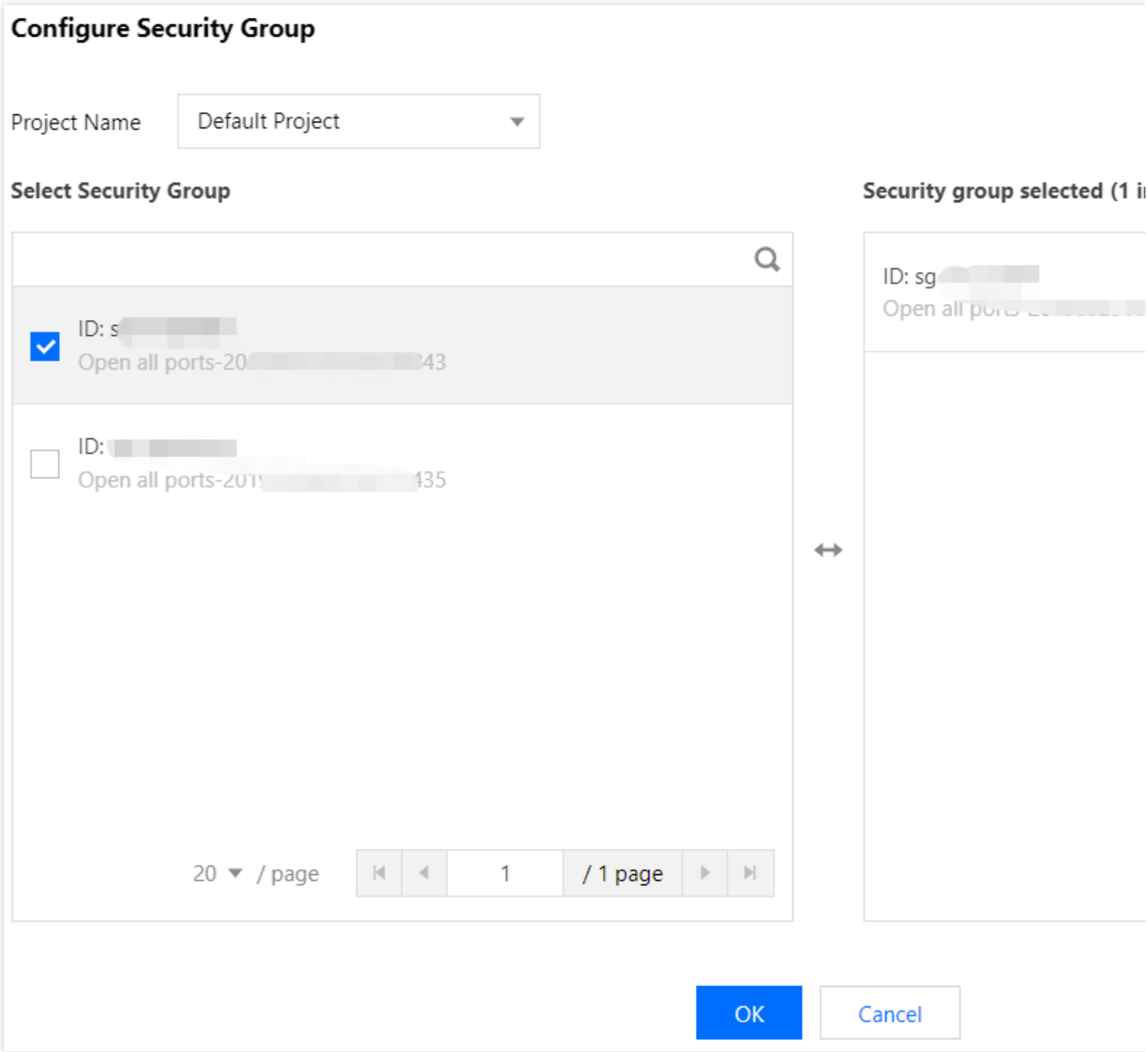
Step 3. Bind the security group to an instance

Note:

Currently, security groups can be configured only for TencentDB for MongoDB instances in VPC.

1. Log in to the TencentDB for MongoDB console.

2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**.
The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Operation** column of the target instance, select **More** > **Security Group**.
You can also click the target instance name, select the **Data Security** tab, and click **Configure Security Group**.
6. In the **Configure Security Group** pop-up window, select the target security group and click **OK**.



More Operations

Adjusting the priority of a bound security group

1. Log in to the TencentDB for MongoDB console.

2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID, select the **Data Security** tab, and view all security groups of the instance.
6. Click **Edit**. You can click

or

in the **Operation** column to adjust the filtering priorities of security groups.
7. Click **Save**.

Adjusting an inbound/outbound rule

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID, select the **Data Security** tab, and view all security groups of the instance.
6. In the security group list, click the target **security group ID** or name to enter the Security Group page.
7. Find the security group rule to be modified and click **Edit** in the **Operation** column to edit it.

Importing a security group rule

1. On the Security Group page, click the ID/name of a security group.
2. On the **Inbound Rule** or **Outbound Rule** tab, click **Import Rule**.
3. In the pop-up window, select an edited inbound/outbound rule template file and click **Import**.

Note:

As existing rules will be overwritten after importing, we recommend that you export the existing rules before importing new ones.

If there are no existing rules in the security group, download a template and edit it before importing it.

Cloning a security group

1. On the Security Group page, find the target security group and click **More** > **Clone** in the **Operation** column.
2. In the pop-up window, select the target region and project and click **OK**.

If the new security group needs to be associated with a CVM instance, do so by managing the CVM instances in the security group.

Deleting a security group

1. On the Security Group page, find the security group to be deleted and click **More** > **Delete** in the **Operation** column.
2. In the pop-up window, click **OK**.

If the current security group is associated with a CVM instance, it must be disassociated first before being deleted.

References

For more information, see Security Group.

SSL Authentication
Enabling SSL Authentication

Last updated：2024-10-10 14:26:09

Overview

Secure Sockets Layer (SSL) authentication is a process that authenticates the connection from the user client to the TencentDB server. After SSL encryption is enabled, you can get a CA certificate and upload it to the server. Then, when the client accesses the database, the SSL protocol will be activated to establish an SSL secure channel between the client and the server. This implements encrypted data transfer, prevents data from being intercepted, tampered with, and eavesdropped during transfer, and ultimately ensures the data security for both the client and the server.

Billing Overview

SSL encryption is free of charge.

Notes

You need to restart the instance to enable SSL. Perform this operation during off-peak hours, or ensure that your application has a reconnection feature.
Enabling SSL encryption ensures the security of data access and transfer but will significantly increase CPU utilization. We recommend that you enable it only when encryption is required.
When SSL is enabled, you will receive an expiration alarm 30 days, 15 days, and 7 days before the expiration of your certificate and on its expiration date. Refresh the SSL certificate in time; otherwise, the access authentication through SSL certificate will fail.

Version description

New instances of TencentDB for MongoDB 4.0 and later support SSL authentication.
Existing instances of TencentDB for MongoDB 3.6 need to be upgraded to v4.0 to support SSL authentication.

Prerequisites

The database instance is in **Running** status, with no ongoing tasks.
The operation is performed in off-peak hours, or the client has an automatic reconnection mechanism.

Directions

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**.
The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. In the **Instance ID/Name** column of the target instance, click the instance ID in blue font to enter the **Instance Details** page.
6. Click the **Data Security** tab and select the **Access Encryption** tab.
7. Click


after **Enable SSL**.
8. In the **Enable SSL** window, confirm the impact of enabling SSL and click **OK**.
9. Wait for the **Enable SSL** status to become **Enabled** and click **Download Certificate**.
If you receive a certificate expiration warning message, and the certificate has expired. Click **Refresh Certificate** to update the certificate file.
10. In the bottom-left corner of the page, get the certificate **MongoDB-CA.crt**.
11. You can use Mongo Shell to connect to TencentDB for MongoDB. For detailed directions, see Using Mongo Shell to Connect to Database by SSL Authentication.
You can use multi-language SDKs to connect to TencentDB for MongoDB. For detailed directions, see Using Multi-Language SDKs to Connect to Database by SSL Authentication.

Using Mongo Shell to Connect to Database by SSL Authentication

Last updated：2024-01-15 14:40:06

Overview

When using Mongo Shell to connect to database, you can enable Secure Sockets Layer (SSL) encryption feature to improve the security of the data linkage. The network connection can be encrypted at the transport layer with the SSL encryption feature to improve the communication data security and ensure data integrity.

Prerequisites

You have created a Linux CVM instance in the same VPC and region as the TencentDB for MongoDB instance.
You have obtained the username and password information for database instance access on the **Account Management** tab on the **Database Management** page. For detailed directions, see Account Management.
You have obtained the private IP and port for database instance access in the **Instance List**. For detailed directions, see Viewing Instance Details.
You have enabled SSL encryption feature on the instance. For details, see Enabling SSL Authentication.

Directions

This document uses the Linux operating system as an example to demonstrate the specific operation process.
1. Download the SSL CA certificate. For detailed directions, see Enabling SSL Authentication.
2. Upload the certificate file **MongoDB-CA.crt** to the CVM instance with Mongo Shell installed.
3. On the CVM instance with Mongo Shell installed, run the following command to connect to the MongoDB database.
Note:
For MongoDB 4.2 and later, Transport Layer Security (TLS) is used to perform data authentication. TLS is the security protocol of transport layer, an upgraded version of SSL. When you are not sure whether to use SSL authentication or TLS authentication, you can execute `./mongo_ssl -h` to confirm the authentication method.
SSL Authentication

```
./bin/mongo -umongouser -plxh***** 172.xx.xx.xx:27017/admin --ssl --sslCAFile Mongo
```

Replace the following parameters as needed.
-u: Database connection username

-p: Username password

`172.xx.xx.xx` and `27017` specify the IP (port number included) and port of the TencentDB for MongoDB instance respectively. If you forgot the username and password, view and modify the account and password as instructed in Account Management.

--sslCAFile: Certificate file path of SSL authentication

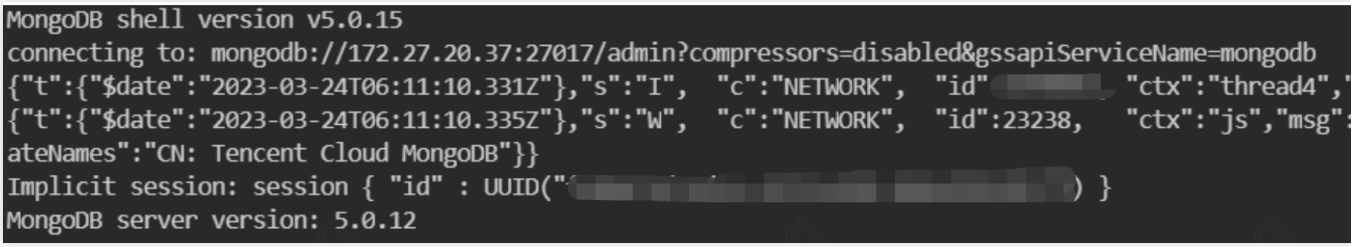
TLS Authentication

```
./bin/mongo -umongouser -plxh***** 172.xx.xx.xx:27017/admin --tls --tlsCAFile /data
```

--tlsCAFile: Certificate file path of TLS authentication

4. After a successful connection, the following information will be displayed:

The prompt information may vary by MongoDB shell version. The following takes v5.0.15 as an example.



References

For SDK connection in other languages, see Using Multi-Language SDKs to Connect to Database by SSL Authentication.

Using Multi-Language SDKs to Connect to Database by SSL Authentication

Last updated：2024-01-15 14:40:06

Java

Keytool is a native key and certificate management tool in Java, which is convenient for you to manage your public/private keys and certificates for authentication services. Keytool stores keys and certificates in keystore. Converting certificate format with keytool:

```
keytool -importcert -trustcacerts -file <certificate file> -keystore <trust store>
```

`-file <certificate file>` : SSL certificate or TLS certificate file **MongoDB-CA.crt**

`-keystore <trust store>` : Specified keystore name

`-storepass <password>` : Specified keystore password.

To set the keystore of JVM system property, you need to change the value of `trustStore` and `password` as required to refer to correct keystore. You also need to replace the URI combination with the user password information that is used to access the database.

```
System.setProperty("javax.net.ssl.trustStore", trustStore);
System.setProperty("javax.net.ssl.trustStorePassword", password);

import com.mongodb.MongoClientURI;
import com.mongodb.MongoClientOptions;

String uri = "mongodb://mongouser:password@10.x.x.1:27017/admin";
MongoClientOptions opt = MongoClientOptions.builder().sslEnabled(true).sslInvalidHo
MongoClient client = new MongoClient(uri, options);
```

Go

The following is a code example of using GO language to connect to database by SSL authentication. You need to replace the path of the certificate file MongoDB-CA.crt, the account and password, IP information and port information concatenated in the URI as needed.

```
package main

import (
```

```
    "context"
    "crypto/tls"
    "crypto/x509"
    "io/ioutil"

     "go.mongodb.org/mongo-driver/mongo"
     "go.mongodb.org/mongo-driver/mongo/options"
)

func main() {
    ca, err := ioutil.ReadFile("MongoDB-CA.crt")
    if err != nil {
        return
    }
    pool := x509.NewCertPool()
    ok := pool.AppendCertsFromPEM([]byte(ca))
    if !ok {
        return
    }
    tlsConfig := &tls.Config{
        RootCAs:      pool,
        InsecureSkipVerify: true,
    }
    uri := "mongodb://mongouser:password@10.x.x.1:27017/admin?ssl=true"
    clientOpt := options.Client().ApplyURI(uri)
    clientOpt.SetTLSConfig(tlsConfig)

     client, err := mongo.Connect(context.TODO(), clientOpt)
     if err != nil {
         return
     }
    client.Disconnect(context.TODO())
}
```

Python

The following is a code example of using Python language to connect database by SSL authentication. You need to replace the path of the certificate file MongoDB-CA.crt, the account and password, IP information and port information concatenated in the URI as needed.

```
from pymongo import MongoClient
uri = "mongodb://mongouser:password@10.x.x.1:27017/admin"
client = MongoClient(uri,
           ssl=True,
```

```
            ssl_ca_certs='MongoDB-CA.crt',
            ssl_match_hostname=False)
```

Database Management
Account Management

Last updated：2024-01-15 14:40:06

You can create an account, set account permissions, and change the account password in the TencentDB for MongoDB console to manage database access permissions more easily.

Overview

TencentDB for MongoDB has two default users: `rwuser` and `mongouser`. TencentDB for MongoDB 3.2 supports both of them by default, while v3.6, v4.0, v4.2, and v4.4 only support the `mongouser` user by default.
Only **rwuser** is authenticated with MONGODB-CR.
Both **mongouser** and users created in the TencentDB for MongoDB console are authenticated with SCRAM-SHA-1.
You can set multiple accounts and grant each of them different database read/write permissions for database access at a finer granularity and higher data security.

Version Description

All TencentDB for MongoDB versions support database account management.

Note

After you create an account and grant it the access permission, it will take effect in 2 minutes after the system performs the backend configuration.
We recommend that you reset the database password periodically at least once every three months.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB replica set or sharded cluster instance is in **Running** status.

Directions

Viewing the account information

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded cluster instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. Select the **Database Management** > **Account Management** page to view the information of all accounts of the current database.

Creating an account

1. On the **Account Management** page, click **Create Account**.
2. On the **Create Account** tab in the **Create Account** pop-up window, configure the account information according to the table below and click **OK**.

Parameter	Required	Description	Value Range/Valid Values	Example
Account ID	Yes	Set the name of the new account.	The account name requirements are as follows: It can contain 1–32 characters. It can contain letters, digits, underscores, and hyphens.	test
Account Password	Yes	Set the password of the new account.	The password requirements are as follows: It can contain 8–32 characters. It must contain at least two of the following types of characters: letters, digits, and special symbols `!@#%^*()_` .	test@123
Confirm Password	Yes	Confirm the password of the new account.	The password requirements are as follows: It can contain 8–32 characters. It must contain at least two of the following types of characters: letters, digits, and special symbols `!@#%^*()_` .	test@123
Remarks	No	Remarks	Any characters	test
mongouser password	Yes	Enter the password of the `mongouser` user.	The password of the `mongouser` user. Required password strength: It can contain 8–32 characters. It can contain letters and digits. It can contain special symbols `!@#%^*()_` .	test@123

			It cannot all be letters or digits.	

3. On the **Set Permissions** page, set the database access permissions for this account.

Parameter	Description	Value Range/Valid Values
Global Permission	Set the global permission to access all databases for this account.	No permission: No data read/write permission. Read-Only: Only data read permission. Read/Write: Data read/write permission.
Instance Details	Set the permission to access a specific database for this account.	Inherit global data: Global permission is inherited. No permission: No data read/write permission. Read-Only: Only data read permission. Read/Write: Data read/write permission.

4. (Optional) Click **Create Database**, and a new database will be added to the database list. Enter the name of the new database in the input box, click **OK** after the input box, and set the access permission of this database.

Note:

The created new database is not a real database but is only used to preset the access permission of this database.

5. Click **OK**, wait 2 minutes for the system configuration to take effect, and then you can use this account to access databases.

Modifying the account permission

1. In the account list on the **Account Management** tab, find the target account.
2. Click **View/Set** in the **Operation** column.
3. In the **Set Permissions** pop-up window, modify the account permission.
4. Click **OK**.

Changing the account password

1. In the account list on the **Account Management** tab, find the target account.
2. Click **Reset Password** in the **Operation** column.
3. In the **Reset Password** pop-up window, enter the **New Password** and **Confirm Password**.

The password requirements are as follows:

It can contain 8–32 characters.

It must contain at least two of the following types of characters: letters, digits, and special symbols `!@#%^*()_` .

4. Click **OK**.

Relevant Operations

Viewing the account URI

1. In the account list on the **Account Management** tab, find the target account.
2. Click **Connection URI** in the **Operation** column.
3. In the **Connection help** pop-up window, view the information of the connection URI of the account.

For more information on instance connection, see Connecting to TencentDB for MongoDB Instance.

4. Click **OK**.

Deleting an account

1. In the account list on the **Account Management** tab, find the target account.
2. Click **Delete** in the **Operation** column.
3. In the **Delete User** pop-up window, confirm the information of the account to be deleted.
4. Click **OK**.

Related APIs

API Name	Description
ResetDBInstancePassword	Changes the password of an instance user

Slow Log Management

Last updated：2024-01-15 14:40:06

You can view and analyze the slow logs generated during database operations in the TencentDB for MongoDB console for targeted database performance optimization.

Overview

Slow logs are often used as the basis for optimizing business operations in MongoDB. For more information, see Database Profiler.
The system provides two query methods as described below:
Query statistics: Slow logs for the specified time period are queried, and the query results are aggregated and analyzed by command (operation) type.
Query details: A specific operation command is specified to query slow logs, and the query results are displayed in a list displaying the execution durations and log details.

Version Description

Currently, all TencentDB for MongoDB versions support slow log management.

Note

The system logs operations with an execution time of more than 100 ms.
The slow logs are retained for 7 days. The time span for a single query cannot exceed 1 day.
Only the first 10,000 slow logs can be queried. If the query is slow, you can narrow down the query time period.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB replica set or sharded instance is in **Running** status.

Directions

Querying slow logs

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for replica set instances and sharded instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. Select the **Database Management** > **Slow Log Query** tab.
7. On the **Slow Log Query** tab, select a **query method** to query slow logs.

Query statistics: Select a **query time period**, set the **time consumed** threshold, and click **Query**.

Query details: Select the specific executed command to be queried in **Executed Command**, select a **query time period**, set the **time consumed** threshold, and click **Query**.

8. View and analyze the slow logs.

A **statistics query** result contains four fields:

Query Method: Statistics query.

Sample Command: Output statements aggregated in the command type dimension, which records the operation of the slow log. You mainly need to refer to the command when troubleshooting problems.

Note:

Pay attention to keywords such as `command`, `COLLSCAN`, `IXSCAN`, `keysExamined`, and `docsExamined`. For more log descriptions, see Log Messages.

- `command` indicates an operation request recorded in a slow log.

`COLLSCAN` indicates that a full-collection scan is performed. `IXSCAN` indicates that an index scan is performed. `keysExamined` and `docsExamined` indicate the numbers of index entries and documents scanned respectively. Larger **keysExamined** and **docsExamined** values indicate that no index is created or the created index is less distinctive. For more information on how to optimize indexes, see Optimizing Indexes to Break Through Read/Write Performance Bottlenecks.

Average Execution Time (ms): Average execution time (in ms) of operations aggregated in the command type dimension.

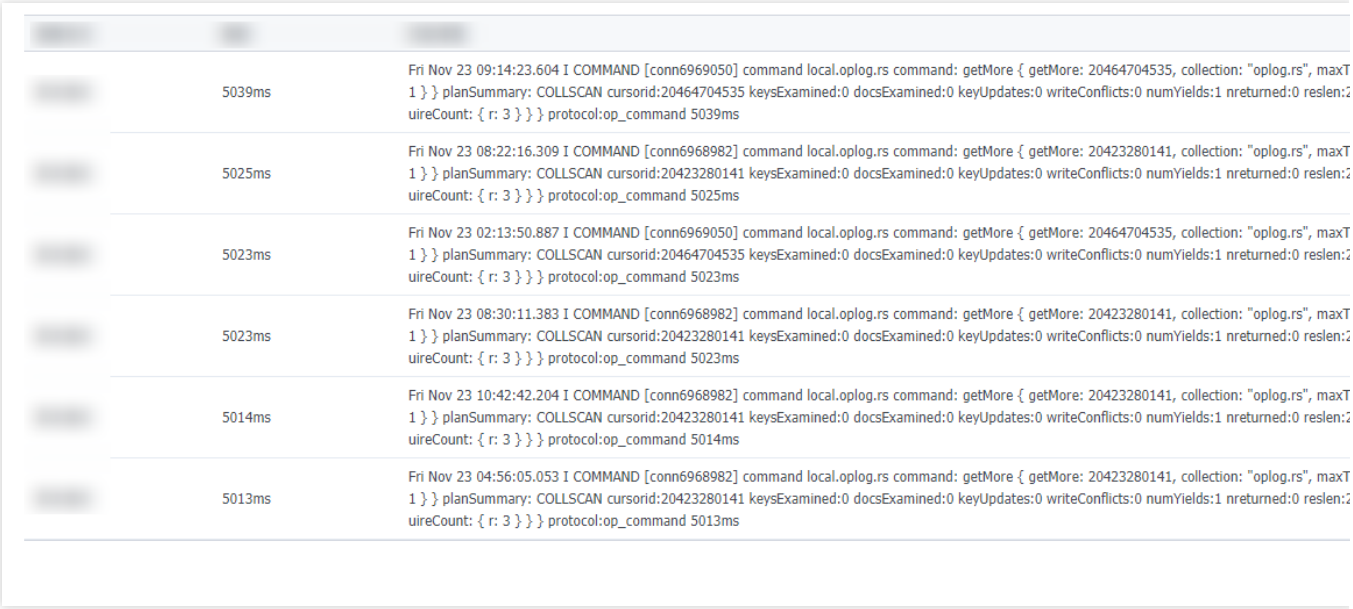
Total: Total occurrences of operations aggregated in the command type dimension.

A **details query** result contains three fields:

Query Method: Details query.

Time Consumed: Execution time of the business command (in ms).

Log Details: Details of the business command.



	5039ms	Fri Nov 23 09:14:23.604 I COMMAND [conn6969050] command local.oplog.rs command: getMore { getMore: 20464704535, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20464704535 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5039ms
	5025ms	Fri Nov 23 08:22:16.309 I COMMAND [conn6968982] command local.oplog.rs command: getMore { getMore: 20423280141, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20423280141 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5025ms
	5023ms	Fri Nov 23 02:13:50.887 I COMMAND [conn6969050] command local.oplog.rs command: getMore { getMore: 20464704535, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20464704535 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5023ms
	5023ms	Fri Nov 23 08:30:11.383 I COMMAND [conn6968982] command local.oplog.rs command: getMore { getMore: 20423280141, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20423280141 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5023ms
	5014ms	Fri Nov 23 10:42:42.204 I COMMAND [conn6968982] command local.oplog.rs command: getMore { getMore: 20423280141, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20423280141 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5014ms
	5013ms	Fri Nov 23 04:56:05.053 I COMMAND [conn6968982] command local.oplog.rs command: getMore { getMore: 20423280141, collection: "oplog.rs", maxT 1 } } planSummary: COLLSCAN cursorid:20423280141 keysExamined:0 docsExamined:0 keyUpdates:0 writeConflicts:0 numYields:1 nreturned:0 reslen: uireCount: { r: 3 } } } protocol:op_command 5013ms

Managing slow logs

Viewing a slow log request statement

1. On the **Slow Query Management** page, you can view the slow log request statements.
2. In the search box in the top-right corner, enter the information to be queried for search.

Parameter	Description
Query Command	Query statement
Op Type	Operation type
Node Location	Node of the executed operation
Namespace	Namespace of the database collection
Executed Time	Time consumed
Details	Details of the executed statement

Batch killing

1. On the **Slow Query Management** page, select the slow log request statements to be cleared.
2. Click **Batch Kill** above the list.
3. In the **Note** pop-up window, read the prompt carefully.
4. Click **OK**.

Downloading the slow log file

1. On the **Slow Log Download List** page, you can view current slow log files.
2. Find the file to be downloaded and click **Download** in the **Operation** column.

Related APIs

API Name	Description
DescribeSlowLogs	Gets the slow log information
DescribeSlowLogPatterns	Gets the slow log statistics

Connection Management

Last updated：2024-01-15 14:40:06

Overview

TencentDB for MongoDB records the IPs of clients connected to the current instance and the number of connections. When there is a large number of concurrent application requests, if the configured upper limit of connections is insufficient, the current database specification cannot sustain such requests. In this case, you can directly increase the upper limit in the console to sustain business peaks.

Version description

Replica set: All TencentDB for MongoDB versions support connection management.
Sharded cluster: TencentDB for MongoDB 3.2, 3.6, 4.0, 4.4, and 5.0 support connection management, while v4.2 doesn't.

Notes

The system records the IPs of clients connected to the current instance and the number of connections. You can choose to manually release connection requests.
If the number of connections reaches or exceeds 80% of the upper limit and affects the establishment of new connections, you can click **Increase Connections** in the console to increase the maximum number of connections to 150% of the original limit for the next 6 hours.
If the problem persists, contact the aftersales service or submit a ticket for assistance.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The TencentDB for MongoDB replica set or sharded cluster instance is in **Running** status.

Directions

Viewing the number of connections

1. Log in to the TencentDB for MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. Above the **Instance List** on the right, select the region.
4. In the instance list, find the target instance.
5. Click the target instance ID to enter the **Instance Details** page.
6. Select the **Database Management** > **Manage Connection** tab.
7. View all client connection statistics of the current database.

Parameter	Description
Real-time connections:	Number of all connections to the current database.
Connection percentage	Percentage of all client connections to the current database to the maximum number of connections.
Maximum connections	Upper limit of the number of connections.
Remaining	The remaining usage duration of the increased upper limit.
Client IP	The client IP from which the database is connected.
Connections	Number of connections.

Increasing connections

1. On the **Manage Connection** tab, click **Increase Connections**.
2. In the **Note** window, confirm the notes and click **OK**.

Related APIs

API	Description
DescribeClientConnections	Queries the client connection information of an instance

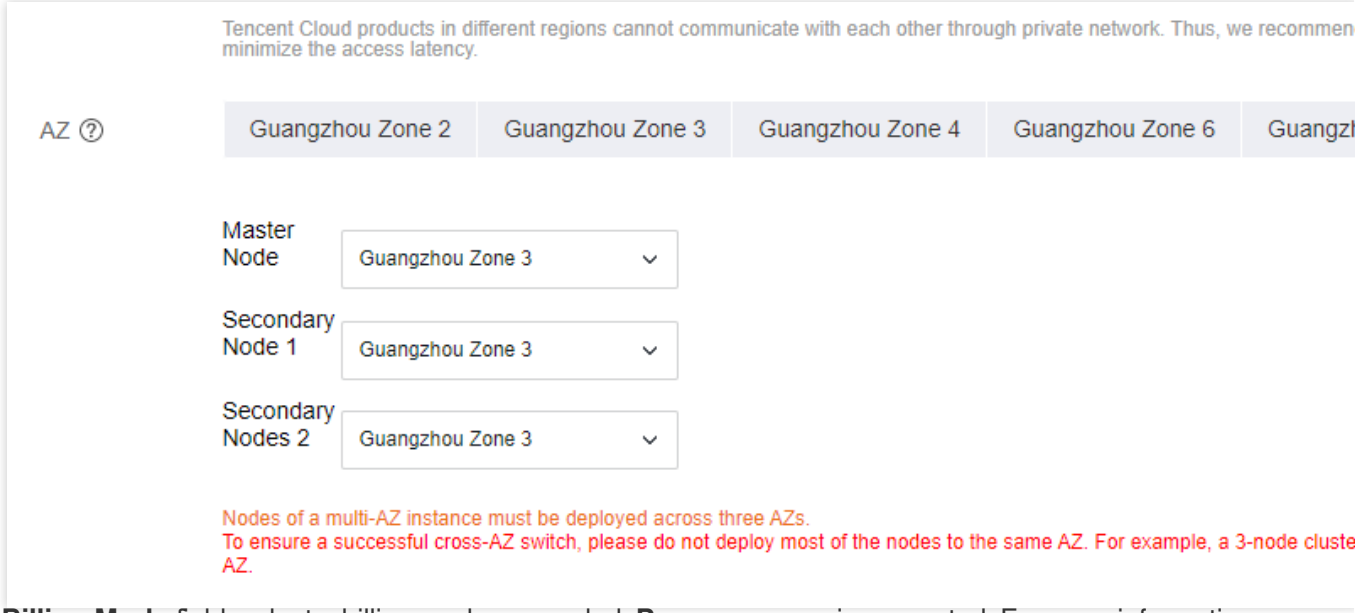
Multi-AZ Deployment

Last updated：2024-01-15 14:40:06

Multi-AZ deployment refers to deployment of TencentDB for MongoDB replicas across multiple AZs in the same region. Multi-AZ deployed instances have higher availability and better disaster recovery capability than single-AZ deployed instances.

Creating a multi-AZ deployed instance

1. Log in to the TencentDB for MongoDB purchase page with a Tencent Cloud account.
2. On the purchase page, configure the multi-AZ deployment parameters.



In the **Billing Mode** field, select a billing mode as needed. **Pay-as-you-go** is supported. For more information, see Billing Overview.

In the **Region** field, select the region of the multi-AZ deployed instance. We recommend that you select the region closest to your end users to minimize the access latency.

In the **AZ** field, you can click **Multi-AZ Deployment** and select the AZ in the drop-down lists after **Primary Node**, **Secondary Node 1**, and **Secondary Node 2** respectively. To guarantee a smooth cross-AZ switch, multi-AZ deployment does not support deploying most cluster nodes in the same AZ; that is, the primary and secondary nodes can be deployed only in three different AZs separately.

For more information on how to configure other parameters, see Creating TencentDB for MongoDB Instance.

3. If you select **Pay as You Go**, you can click **Billing Details** to view product pricing and confirm the total fees.
4. Click **Buy Now**. After the purchase success message is displayed, click **Go to Console** to enter the instance list page. After the instance status in the **Monitoring/Status** column becomes **Running**, the multiple AZs of the instance

will be displayed in the **AZ** column.

Accessing a multi-AZ deployed instance

You can use MongoDB Shell or a concatenated URI through the SDK client for multiple programming languages to access a multi-AZ deployed instance. For detailed directions, see Connecting to TencentDB for MongoDB Instance.

Upgrading from single-AZ deployment to multi-AZ deployment

You can upgrade a single-AZ deployed instance to a multi-AZ deployed instance. For detailed directions, see Modifying Instance AZ.

Disaster Recovery/Read-Only Instances Overview

Last updated：2024-01-15 14:40:06

Basic Concepts

Read-only instance

TencentDB for MongoDB allows you to create one or multiple read-only instances in the source instance AZ or another AZ based on the cluster architecture and storage engine of the source instance. The data in the source instance will be automatically synced to read-only instances, which are granted the read-only permission. In this way, the read requests of the source instance can be distributed to read-only instances to increase the read/write performance and application throughput.

Disaster recovery instance

TencentDB for MongoDB allows you to create one or multiple disaster recovery instances in another region based on the cluster architecture and storage engine of the source instance. The data in the source instance will be automatically synced to disaster recovery instances, which are granted the read-only permission. When the region of the source instance is disconnected due to a force majeure event such as power outage or network issue, a disaster recovery instance can be promoted to primary instance to implement cross-region disaster recovery and quickly sustain business needs. This helps you improve the business continuity at low costs and guarantee the data reliability.

Differences between read-only instance and disaster recovery instance

Both read-only and disaster recovery instances are built based on the cluster architecture and storage engine of the source instance. They have the following differences:

Difference	Description	Read-Only Instance	Disaster Recovery Instance
Architecture type	The system architecture of a read-only or disaster recovery instance cluster, which can be **replica set** or **sharded cluster** but not **single-node**.	Same as that of the source instance.	Same as that of the source instance.
Cross-region	Whether a read-only or disaster recovery	No	Yes

	instance can be created in another region based on the source instance.		
Cross-AZ	Whether a read-only or disaster recovery instance can be created in another AZ in the current region based on the source instance.	Yes	Yes
Database version	The compatible MongoDB version, including v4.4, v4.2, v4.0, v3.6, and v3.2. v3.2 is no longer for sale.	Same as that of the source instance and cannot be upgraded.	Same as that of the source instance and cannot be upgraded.
Storage engine	The storage engine, which is WiredTiger by default.	Same as that of the source instance.	Same as that of the source instance.
Instance specifications	The CPU, memory, and disk capacity requirements of a read-only or disaster recovery instance to guarantee the service capacity.	Cannot be lower than those of the source instance.	Cannot be lower than those of the source instance.
Data write	Data write and database creation and deletion	No	No
Backup and rollback	Data backup and restoration.	No	No
Account management	Database access account creation and deletion.	No	No
Manual disassociation from the	Whether you can manually disassociate a read-only or disaster recovery	No. Only after the source instance is terminated can a read-only instance be disassociated from it	Yes. You can promote a disaster recovery instance to primary instance, and it will become a general instance

source instance	instance from the source instance in the console.	automatically. After disassociation, the read-only instance will be promoted to general instance and can be read/written normally.	that can be read/written normally to quickly sustain the business needs.
AZ upgrade	Whether the single-AZ deployment of a read-only or disaster recovery instance can be upgraded to multi-AZ deployment.	Yes	Yes

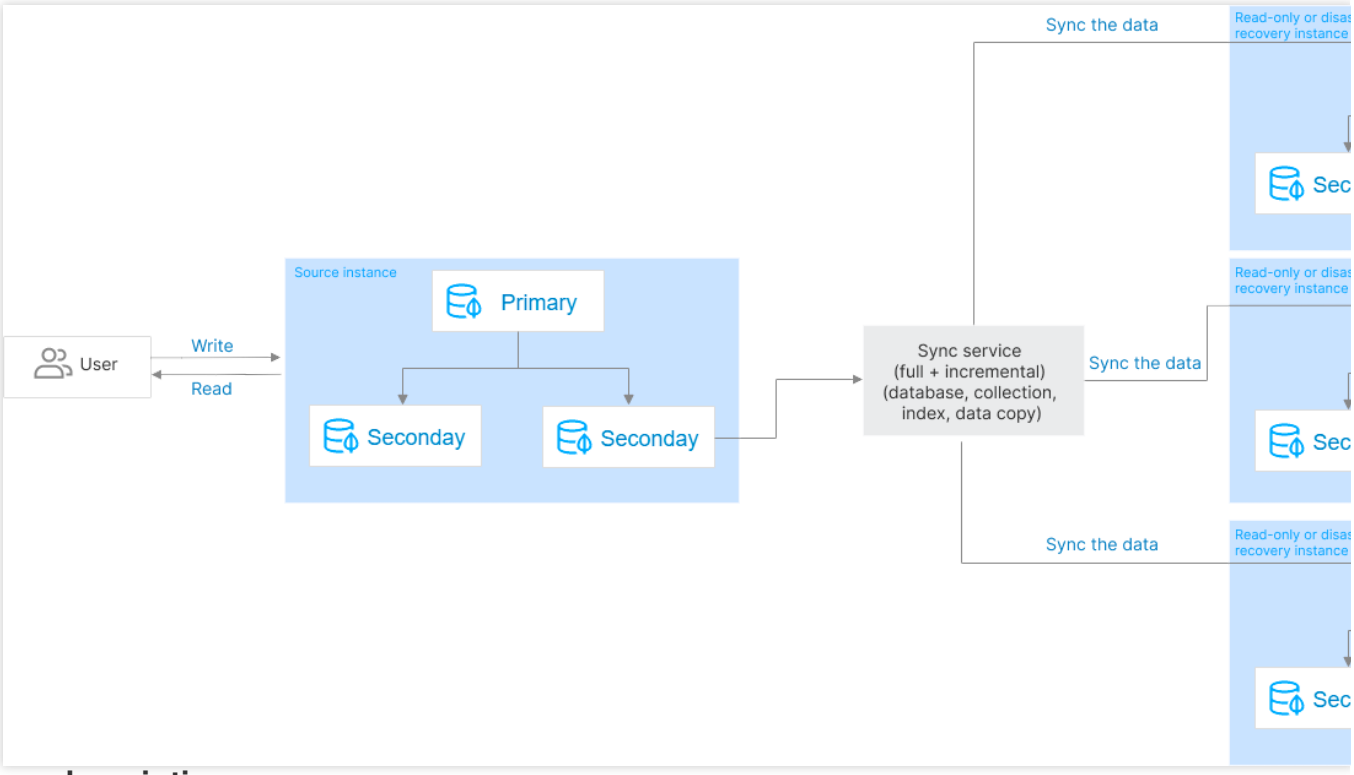
How It Works

Read-only and disaster recovery instances can be understood as a data sync service, which continuously syncs the existing and incremental data (including databases, collections, indexes, and documents) of the source instance to the target read-only or disaster recovery instance and grants the target instance only the read-only permission to relieve the pressure on the source cluster.

Once a read-only or disaster recovery instance is created, full data sync will be initiated, and data sync operations will be logged as oplog. After full data sync is completed, the oplog can be replayed to keep incrementally syncing changed or added data. This works similarly to primary-secondary sync in a replica set. The entire process has two stages:

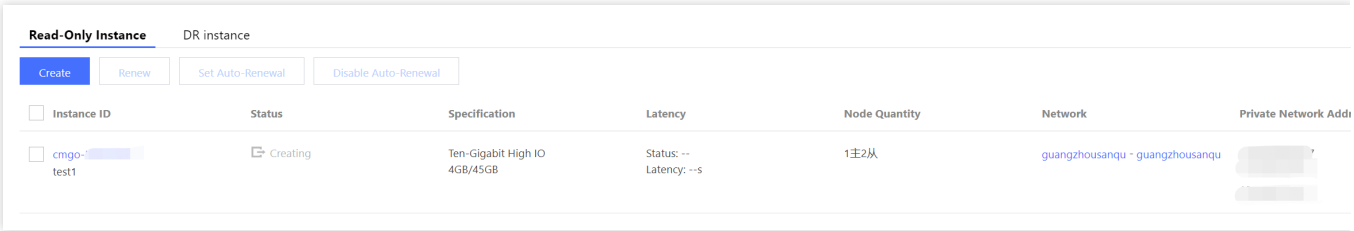
Full sync stage, where full data is synced. Before it starts, the latest oplog timestamp of the source cluster will be recorded. After it starts, the metadata, indexes, and data in all collections in the source cluster will be read and concurrently inserted to collections with the same name in the target. The duration of full sync is proportional to the data volume of the source cluster.

Incremental sync stage, which follows the full sync stage. During this stage, the oplog of the source cluster will be pulled based on the oplog timestamp recorded when the full sync stage starts and then replayed in the target.



Latency description

Due to the delay in data sync, the real-timeness of data sync for read-only instances may not be guaranteed. If your business requires read/write separation and high real-timeness, we recommend that your business read the secondary nodes of the primary instance. For more information, see Connecting to TencentDB for MongoDB Instance. You can log in to the console and go to the **RO/DR** page to view the status and latency of sync from the primary instance to the target instance.



Performance optimization

Similar to concurrent replay of MongoDB replica set oplogs, the read-only/disaster recovery data sync service pulls the oplogs to the cache, parses them concurrently, hashes them by collection name to ensure that the oplogs are sequential at the collection level, linearly hashes each oplog segment by document ID to assign oplogs of the same document to the same thread, and then concurrently sends the threads to the target instance. This guarantees the incremental data sync performance and keeps the sync latency within seconds.

Data security

During incremental sync, the sync service persistently stores the currently synced latest oplog timestamp, and the replay process of the sync service is idempotent. Therefore, the sync service supports checkpoint restart during the

incremental sync stage. Even if a failure occurs in the source or target cluster, data security issues won't occur in the sync service.
During incremental data sync, if a primary/secondary switch occurs in the target cluster due to a disk failure or network issue, data may get lost. To address this problem, TencentDB for MongoDB adds an oplog that records the sync progress. The data sync service regularly inserts a sync progress record into the oplog transactions of the target cluster. After a new primary node takes effect, it will locate the latest record in its oplog transactions and sync the data again to prevent data loss.

Sync stability

Each read-only or disaster recovery instance is sustained by a separate data sync service, and each data sync service uses distributed locks and the lease mechanism to guarantee the service uniqueness and availability, monitors the sync task in real time, and fine-tunes the instance regularly to guarantee the data sync stability and reliability.

Impact and Limits

Impact on the source cluster

The impact of the read-only and disaster recovery data sync service is only limited to secondary nodes, and data will be pulled from a secondary node (hidden one preferably) in the source cluster.
During full data sync, the `getMore` request is used to continuously pull data.
During incremental data sync, the `getMore` request is used to continuously pull oplogs.
During both the full and incremental data sync stages, the sync service will create a cursor for sequential reads on the secondary node to mark the read progress, which has little impact on the secondary node.

Use limits

Read-only and disaster recovery instances belong to the source instance and cannot exist independently.
Read-only and disaster recovery instances will be unwritable after being created.
After the source instance is terminated, the system will automatically disconnect the sync service and promote a read-only instance to general instance for normal read/write.
Database version: For sharded clusters, read-only and disaster recovery instances can be on versions 4.0, 4.2, and 4.4. For replica sets, they can be on versions 3.2, 3.6, 4.0, 4.2, and 4.4. If the current instance has a read-only or disaster recovery instance, the database versions of both the current instance and the read-only or disaster recovery instance cannot be upgraded.
Quantity: You can create up to three read-only instances and up to three disaster recovery instances for an instance.
Cluster architecture and storage engine: The cluster architecture and storage engine of a read-only or disaster recovery instance are always the same as those of the source instance and cannot be changed.
Account management: When a read-only or disaster recovery instance is created, the account information in the source instance will be synced to it automatically. You cannot create or delete accounts in it. If the access account or

password in the source instance is changed, the account won't be automatically synced to it. You need to manually modify the account information in it during connection; otherwise, an error will occur, and connection will fail.
Due to network isolation, you cannot create disaster recovery instances in finance zones for source instances in general regions, and vice visa.
Backup and rollback: They are not supported for both read-only and disaster recovery instances.
Data migration: Migrating data to read-only or disaster recovery instances is not supported.

Sync limits

Read-only or disaster recovery instance sync in a replica set is implemented by parsing oplogs, and all DDL operations are supported.
Read-only or disaster recovery instance sync in a sharded cluster is implemented by parsing change streams. A change stream covers an oplog with a layer of application and provides an API to push data in real time. In addition to basic CRUD operations, DDL operations related to database/collection structures and indexes are also supported for the pushed data, including `createIndexes`, `drop`, `rename`, `dropDatabase`, `create`, `createIndexes`, `dropIndexes`, `collMod`, and `convertToCapped`.

Billing Overview

The billing mode of a read-only or disaster recovery instance is the same as that of the source instance. You can select a billing mode based on your business needs. The pay-as-you-go billing mode is supported, and billable items include compute and storage resources. For more information, see Billing Overview.

Creating Read-Only Instances

Last updated：2024-10-10 14:30:10

Scenario Description

In application scenarios with a small amount of write requests but a large number of read requests, a single instance may not withstand the reading pressure, and may even impact the core business. You can create one or more new read-only instances in the source instance's availability zone or other availability zones based on the current instance's cluster architecture and storage engine. This shifts the read requests from the current instance to the read-only instances, achieving elasticity in read capability, enhancing read/write performance, and increasing the application's throughput.

Notes

Due to data synchronization delays, the real-time performance of data synchronization in read-only instances might not be guaranteed. If the business requires read-write separation and has high real-time requirements, it is recommended to read from the secondary node of the primary instance. The synchronization latency between each read-only instance and the primary instance can be checked on the console.
The connection method for the read-only instance is the same as that of the primary instance. See Connecting to MongoDB Instance.
During the lifecycle of a read-only instance, it can only perform read operations and cannot execute data write or update operations.
Read-only instances do not support manual disconnection from the source instance. They are automatically disconnected from the source instance only when the source instance is terminated. The read-only instance then becomes a standard instance, capable of normal reading and writing.

Version Description

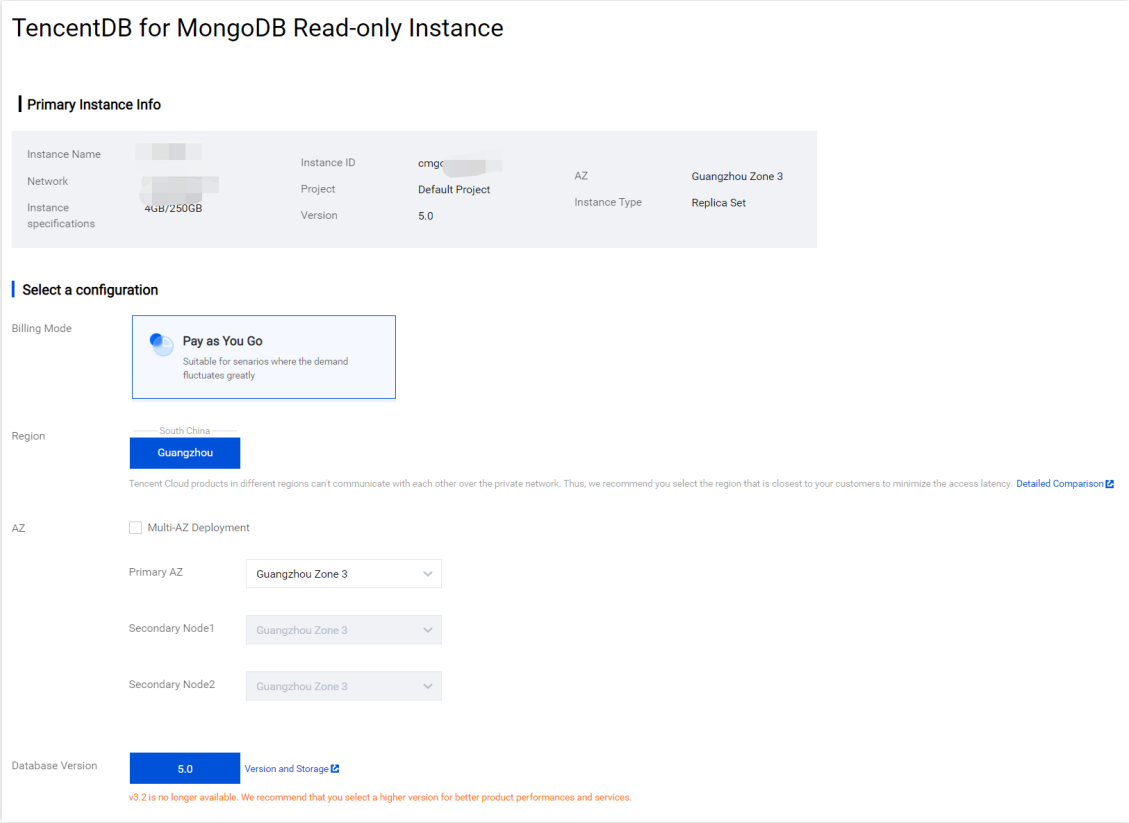
The current MongoDB versions 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 support the creation of read-only instances for replica set instances. Sharded instances are supported only in versions 4.0 and above.

Prerequisites

The current instance is operating normally but with a high read request volume and significant latency, leading to slow database performance. For more information, see Monitoring Overview.
The availability zone for the read-only instance and its associated network have been planned.
The storage specifications and purchase quantity for the read-only instance have been estimated.
The billing model has been selected based on the business scenario, and the costs associated with the read-only instance have been budgeted.

Creating Read-Only Instance

1. Log in to the MongoDB console.
2. On the left sidebar, choose **NoSQL Database** > **MongoDB**.
3. In the **MongoDB** drop-down list, select **Replica Set Instance** or **Shard Instance**. The operations for replica sets and sharded clusters are similar.
4. Above the instance list on the right, select the region.
5. In the instance list, find the target instance.
6. Click the target instance ID to enter the **Instance Details** page.
7. Select the **DO/DR** tab to access the **Read-Only Instance** page.
8. On the **Read-Only Instance** page, click **Create**.
9. On the **TencentDB for MongoDB Read-Only Instance** purchase page, confirm the **Primary Instance Info** and select the required configuration.



See the table below to configure the instance specifications based on your actual needs.

Parameter Name	Parameter Description
Billing Mode	Only supports pay-as-you-go billing mode.
Region	The region of the read-only instance is fixed and matches that of the source instance. It cannot be changed.
Availability Zone	Select whether to enable multiple availability zone deployment. Settings can be adjusted based on actual highly available business requirements.
Database Version	The database version is fixed and matches that of the source instance. It cannot be changed.
Architecture Type	The architecture type is fixed and matches that of the source instance. It cannot be changed. For specific information about the architecture type, see System Architecture.
Storage Engine	The default storage engine is WiredTiger.
Mongod Specification	Select the compute specifications for the database instance from the dropdown list. The CPU core number and memory capacity of a read-only instance must be equal to or greater than those of the source instance, with higher specifications resulting in higher IOPS. For specific supported specifications, see Product Specification. After creating an instance, you can adjust its compute specifications. For specific operations, see Adjusting Instance Configuration.

MongoDB Shard Quantity	When the architecture type is set to Sharded Cluster, this parameter is displayed. It is used for setting the number of shards in the sharded cluster, with a value range of [1,20]. The shard quantity of a read-only instance must be greater than or equal to that of the source instance. Each shard is a replica set. Increasing the shards quantity can enhance the cluster's storage capacity. Select as needed. After creating an instance, you can adjust the MongoDB shard quantity. For specific operations, see Adjusting Instance Configuration.
Disk Capacity	Select the database instance's storage capacity on the slider. The disk capacity of a read-only instance must be greater than or equal to that of the source instance. MongoDB specifications differ, hence the range of disk capacities varies. See Product Specification. The system by default allocates 10% of the selected storage capacity for Oplog's storage space. The size of Oplog can be adjusted in the console instance list. For specific operations, see Adjusting Oplog Storage Capacity. After creating an instance, you can adjust the instance's disk capacity. For specific operations, see Adjusting Instance Configuration.
Primary/Secondary Nodes Quantity	When the architecture type is set to replica set, this parameter is displayed. The default is 3 nodes (1 primary, 2 secondaries), with three storage nodes forming a one-primary-two-secondary architecture. Custom replication numbers are not currently adjustable. You may select from the dropdown list: 5 nodes (1 primary, 4 secondaries), or 7 nodes (1 primary, 6 secondaries). After creating a read-only instance, you can increase the number of secondary nodes. For specific operations, see Adding Secondary Node.
Primary/Secondary Nodes Quantity per Shard	When the architecture type is set to Sharded Cluster, this parameter is displayed. It is used for setting the number of nodes in each shard of the sharded cluster, with the default being 3 nodes (1 primary, 2 secondary nodes), meaning each shard follows a one-primary-two-secondary, three-node architecture. You can select from the dropdown list: 5 nodes (1 primary, 4 secondary nodes), or 7 nodes (1 primary, 6 secondary nodes). Custom node numbers are not currently supported. After creating the instance, you can increase the number of secondary nodes per shard. For specific operations, see Adding Secondary Node.
Read-Only Node Quantity	Setting the number of read-only nodes, supporting 0 to 5 read-only nodes. Only versions 4.0 and 4.2 support configuring the number of read-only nodes; version 3.6 does not support this. After creating a read-only instance, you can increase the number of read-only nodes. For specific operations, see Adding Read-Only Node.
Configuration Instructions	Based on the configured MongoDB specification, calculate the maximum number of connections per instance to help you predict whether the current specification meets expectations.
Mongos Specifications	When the architecture type is set to Sharded Cluster, this parameter is displayed. It is used for configuring Mongos specifications. After configuring the MongoDB specification,

	Mongos will have default specification configurations. For example, if MongoDB is set to 2-core 4GB, Mongos is automatically configured for 1-core 2GB by default. Upgrading the Mongos specification will incur charges. For pricing, see Product Pricing. The connection limit for a sharded cluster will be determined by the Mongos specifications and quantities you choose. You can view the maximum number of connections in the Configuration Instructions. After creating an instance, you can change the Mongos configuration. For specific operations, see Changing Mongos Node Configuration Specifications.
Mongos Quantity	When the architecture type is set to Sharded Cluster, this parameter is displayed. It is used for configuring the number of Mongos. If the instance deployment is in the same availability zone, the range of Mongos numbers is [3, 32]. If the multi-availability zone deployment is enabled and the instance deployment is across different availability zones, then the range of Mongos numbers is [6, 32]. Increasing the number of Mongos will incur charges. For pricing, see Product Pricing. After creating an instance, you can adjust the number of Mongos. For specific operations, see Add New Mongos Node.
Network Type	Only Virtual Private Cloud is supported.
IPv4 Network	Select a specific Virtual Private Cloud and its subnet. It is recommended to select the same Virtual Private Cloud in the same region as the cloud server. A Virtual Private Cloud has a region attribute (e.g., Guangzhou), while a subnet has an availability zone attribute (e.g., Guangzhou Zone 1). A Virtual Private Cloud can be divided into one or more subnets. Subnets under the same Virtual Private Cloud are by default inter-connected through intranet, whereas Virtual Private Clouds (whether in the same region or not) are by default isolated from each other. After purchasing an instance, you can switch the Virtual Private Cloud. For specific operations, see Switch Instance Network. You can also click **Create a New Virtual Private Cloud** and **Create a New Subnet** to recreate the required network environment. For specific operations, see Creating Virtual Private Cloud.
IPv6 Network	Enable/disable IPv6 access. Currently not supported.
Security Group	Set security group rules for the instance to control inbound traffic to the database. You can select an existing security group from the dropdown menu, or click **Custom Security Group** to set new inbound rules for the security group. For detailed information, see Configuring Security Group.
Specify Project	Allocate the instance to the corresponding project. You can manage instances by project.
Tag	Set tags for your instance. You can classify and manage instances based on tags. Click **Add** to select tag keys and values.
Instance Name	Set the instance name, with a default of 500. Set a recognizable name. Supports Chinese, English, and numbers with a length less than 60, including hyphens "-" and underscores "_".

Purchase Quantity	A single instance can support up to 3 read-only instances.
Total Fees	Pay-as-you-go, displaying the hourly cost. Click **Billing Details**, and see Product Pricing.

10. Confirm that the configuration is correct, click **Buy Now**. After the purchase is successful, click **Go to Console**. In the instance list, once the instance status shows as **Running**, it can be used normally.

Viewing Read-Only Instance

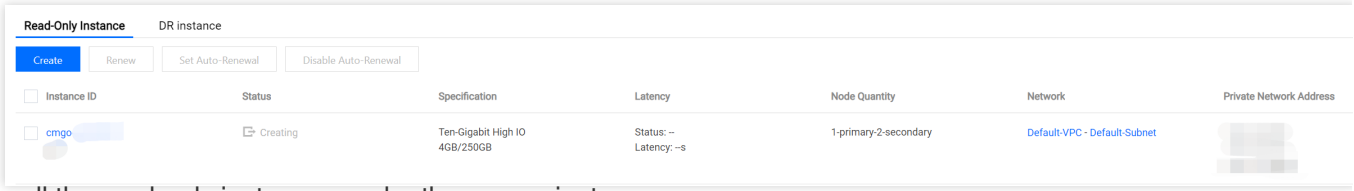
View the source instance from the read-only disaster recovery page.

1. Log in to the MongoDB console.
2. In the **MongoDB** drop-down list on the left sidebar, select **Replica Set Instance** or **Shard Instance**. The operations for the two types of instances are similar.
3. Above the instance list on the right, select the region.
4. In the instance list, find the source instance of the read-only instance.

You can search for the target instance by entering the instance ID, instance name, private network IP or tag key in the search box in the upper right corner of the instance list.

If the instance is not found in the instance list, select **Recycle Bin** from the left sidebar to confirm whether the instance has been isolated in the recycle bin due to expiration of fees. For more information, see Recycle Bin.

5. In the **Instance ID** / **Name** column of the source instance, click the instance ID to enter the **Instance Details** page.
6. Click the **RO/DR** tab and select the **Read-Only Instance** tab.



7. View all the read-only instances under the source instance.

Parameter	Parameter Description
Instance ID	Read-only instance ID and its name. Click the instance ID in blue font to jump to the read-only instance details page. For more information, see Viewing Instance Details.
Status	The current running status of the instance. When it is normal, it should be Running.
Specification	Instance specification information. It includes memory and disk capacity.
Latency	The read-only instance is based on the synchronization status from the source instance, and its latency.

Nodes Quantity	The number of primary and secondary nodes in a read-only instance.
Network	The name of the Virtual Private Cloud to which the read-only instance belongs.
Private Network Address	The private IPv4 address assigned by the Virtual Private Cloud. When accessing the database, the private IP address and its port information need to be configured. For specific operations, see Connecting to MongoDB Instance.
Region	Information about the region and availability zone.
Operation	By clicking configuration changes, you can adjust the specifications of the read-only instance. When adjusting the specifications of the source instance, be sure to synchronously upgrade the specifications of the read-only instance; otherwise, data loss may occur.

Related APIs

APIs	Description
DescribeDBInstances	Queries the list of MongoDB instances.
RenameInstance	Modifies the instance name.

Creating Disaster Recovery Instance

Last updated：2024-10-10 14:43:00

Overview

For scenarios with strong requirements for business continuity and data reliability, or due to regulatory needs, you can create one or more disaster recovery instances across regions based on the current instance's cluster architecture and storage engine. If the region where the current instance is located loses communication due to power, network, or other force majeure factors in any AZ, and the HA system fails, the disaster recovery instance can be promoted to the primary instance. This allows cross-region disaster recovery, ensuring the continuity of data services.

Must-Knows

Due to data synchronization delay, the real-time synchronization of disaster recovery instances may not be guaranteed. The synchronization delay between each disaster recovery instance and the primary instance can be viewed in the console.
During the lifecycle of a disaster recovery instance, it is read-only and cannot be used for data writing or updates.
When the source instance of the disaster recovery instance is terminated, or the disaster recovery instance is manually promoted, it becomes a regular instance with full read/write capabilities, quickly supporting business needs.

Version Information

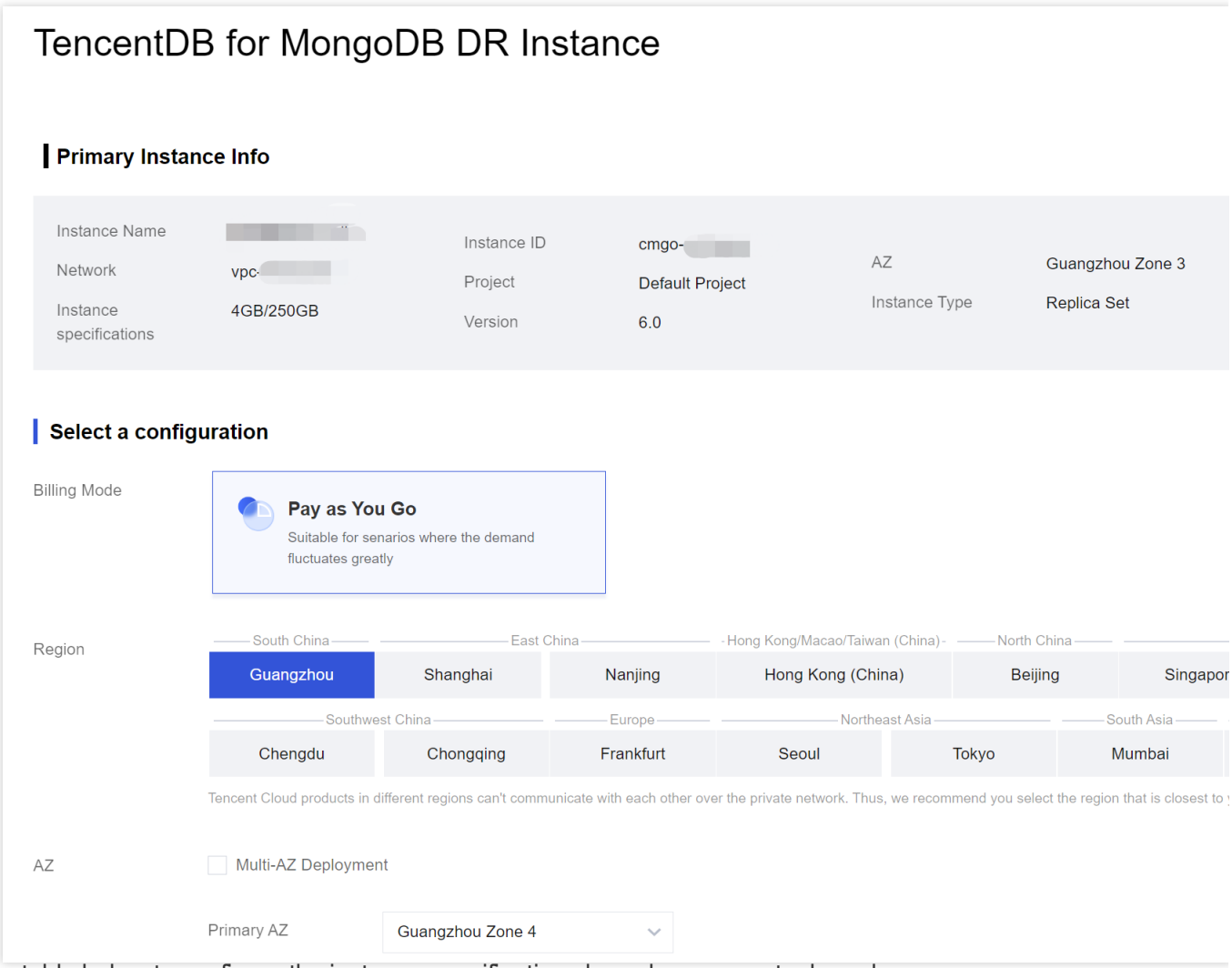
The current replica set instances of versions 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, and 6.0 all support creating read-only instances. Sharded instances support this feature only in version 4.0 and later.

Prerequisite

The current instance is running normally.
The region, AZ, and network for the disaster recovery instance have been planned.
The storage specifications and the number of disaster recovery instances have been estimated.
The billing mode has been selected based on the business scenario, and the necessary budget for the disaster recovery instance has been planned.

Creating a Disaster Recovery Instance

1. Log in to the MongoDB console.
2. In the left sidebar, choose **NoSQL Database** > **MongoDB**.
3. In the **MongoDB** dropdown list, select either **Replica Set Instance** or **Shard Instance**. The process for replica sets and sharded clusters is similar.
4. At the top of the instance list page, select the region.
5. In the instance list, find the target instance for which you want to create a disaster recovery instance.
You can use the search box in the top-right corner of the instance list to find the target instance by entering the instance ID, instance name, private IP address, or tag key.
If the instance is not found in the instance list, go to the left sidebar and choose **Recycle Bin** to check if the instance has been isolated due to expired payments. For more details, see Recycle Bin.
6. Click the target instance ID to enter the **Instance Details** page.
7. Select the **RO/DR** tab, then go to the **DR instance** page, and click **Create** .
8. On the **TencentDB for MongoDB DR Instance** purchase page, confirm the **Primary Instance Info** and select the required configuration.



See the table below to configure the instance specifications based on your actual needs.

Parameter Name	Description
Billing mode	Supports both monthly subscription and pay-as-you-go billing. For details on choosing a billing method, see Billing Overview.
Region	Select the region where the disaster recovery instance is located.
AZ	Choose whether to enable multi-AZ deployment, based on the actual high availability business needs.
Database version	The database version is fixed to match the source instance and cannot be changed.
Architecture type	The architecture type is fixed to match the source instance and cannot be changed. For more details on architecture types, see System Architecture.
Storage engine	The default storage engine is WiredTiger.

Mongod specifications	Select the computing specifications of the database instance from the dropdown list. The disaster recovery instance's CPU cores and memory capacity should be equal to or greater than those of the source instance. The higher the specifications, the higher the IOPS. For more details on supported specifications, see Product Specification. After creating the instance, you can adjust the instance's computing specifications. For more details, see Adjust Instance Configuration.
Mongod shard count	This parameter is displayed when the architecture type is set to Sharded Cluster. It is used to configure the number of shards in the sharded cluster, with a valid range of [1, 20]. The number of shards in the disaster recovery instance should be equal to or greater than that of the source instance. Each shard is a replica set, and increasing the number of shards can expand the storage capacity of the cluster. Select based on your needs. After creating the instance, you can adjust the number of Mongod shards. For detailed steps, see Adjust Instance Configuration.
Disk capacity	Select the storage capacity of the database instance using the slider. The disaster recovery instance's disk capacity should be equal to or greater than that of the source instance. The value range of disk capacity varies depending on the Mongod specifications. See Product Specifications. The system by default allocates 10% of the selected storage capacity for Oplog. The size of the Oplog can be adjusted in the console's instance list. For detailed directions, see Adjust Oplog Capacity. After creating the instance, you can adjust the disk capacity. For detailed directions, see Adjust Instance Configuration.
Number of primary and secondary nodes	This parameter is displayed when the architecture type is set to Replica Set. The default is 3 nodes (1 primary and 2 secondary), with 3 storage nodes forming the 1 primary and 2 secondary architecture. Customizing the number of replica nodes is currently not supported. You can select from the dropdown list to choose 5 nodes (1 primary and 4 secondary) or 7 nodes (1 primary and 6 secondary). After creating the disaster recovery instance, you can increase the number of secondary nodes. For detailed directions, see Add New Secondary Node.
Number of primary and secondary nodes per shard	This parameter is displayed when the architecture type is set to Sharded Cluster. It is used to configure the number of nodes per shard in the cluster. The system default is 3 nodes (1 primary and 2 secondary), meaning each shard follows a 1 primary and 2 secondary architecture. You can choose from the dropdown list to select 5 nodes (1 primary and 4 secondary) or 7 nodes (1 primary and 6 secondary), but customizing the number of nodes is not currently supported. After creating the disaster recovery instance, you can increase the number of secondary nodes per shard. For detailed directions, see Add New Secondary Node.
Number of read-only nodes	Configure the number of read-only nodes, supporting no read-only nodes or 1 to 5 read-only nodes. Only versions 4.0 and 4.2 support configuring the number of read-only nodes; version 3.6 does not support this feature. After creating the disaster recovery instance, you can increase the number of read-only nodes. For detailed directions, see Add New Read-Only Node.

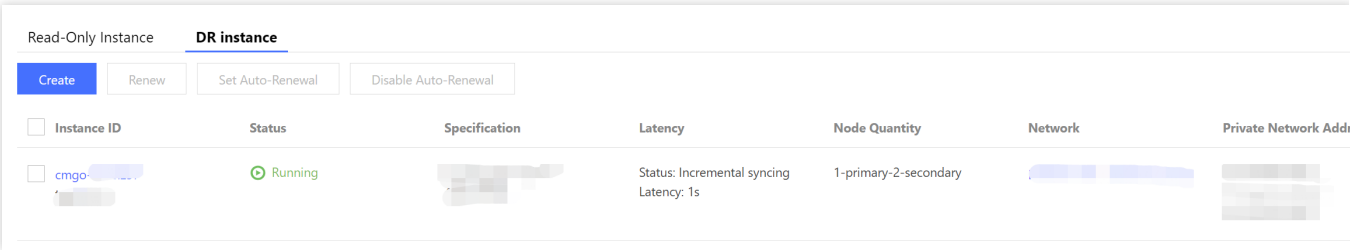
Configuration notes	The maximum number of connections for the instance is calculated based on the configured Mongod specifications, helping you predict whether the current specifications meet your expectations.
Mongos specifications	This parameter is displayed when the architecture type is set to Sharded Cluster. It is used to configure the specifications for Mongos. After the Mongod specifications are configured, Mongos will be automatically assigned default specifications. For example, if Mongod is configured with 2 cores and 4 GB, the default configuration for Mongos will be 1 core and 2 GB. Increasing the Mongos specifications will incur additional fees. For more details on pricing, see Product Pricing. The maximum number of connections for a sharded cluster is determined by the specifications and number of Mongos nodes. You can view the maximum number of connections in the configuration notes. After creating the instance, you can change the configuration of the Mongos nodes. For detailed directions, see Change Mongos Node Configuration Specifications.
Number of Mongos	This parameter is displayed when the architecture type is set to Sharded Cluster. It is used to configure the number of Mongos nodes. If the instance is deployed in the same AZ, the valid range for Mongos nodes is [3, 32]. If multi-AZ deployment is enabled, and the instance is deployed across different AZs, the valid range for Mongos nodes is [6, 32]. Increasing the number of Mongos nodes will incur additional fees. For billing details, see Product Pricing. After creating the instance, you can adjust the number of Mongos nodes. For detailed directions, see Add new Mongos nodes.
Network type	Only VPC is supported.
IPv4 network	Select the specific VPC and its subnet. It is recommended to choose the same VPC within the same region as that of your cloud server. The VPC has a region attribute (e.g., Guangzhou), while the subnet has an AZ attribute (e.g., Guangzhou Zone 1). A VPC can be divided into one or more subnets, and subnets within the same VPC have internal communication by default, whereas different VPCs (regardless of being in the same region) are isolated by default. After purchasing the instance, you can switch the VPC. For detailed directions, see Switch Instance Network. You can also click **Create VPC** and **Create Subnet** to recreate the required network environment. For more details, see Create VPC.
IPv6 network	Check whether to enable IPv6 access. Currently, it is not supported.
Security group	Set security group rules for the instance to control incoming traffic to the database. You can select an existing security group from the dropdown, or click **Customize Security Group** to set new inbound security group rules. For more details, see Configure Security Group.
Assign project	Assign the instance to the appropriate project. You can manage instances based on the assigned project.
Tag	Set tags for the instance. You can categorize and manage instances based on tags. Click **Add** to select a tag key and tag value.
Instance	Set the name for the instance. Choose a name that is easy to identify. It supports Chinese,

name	English, or numbers with a length of fewer than 60 characters, as well as hyphens (-) and underscores (_).
Quantity purchased	A maximum of 3 disaster recovery instances can be created for a single instance.
Total fees	For pay-as-you-go billing, the hourly fee is displayed. Click billing details for more information and see Product Pricing.

9. Once you have confirmed the parameter configuration is correct, click **Buy Now** to complete the purchase. After the purchase success prompt is prompted, click **Go to Console**. In the instance list, once the instance status shows **Running**, it is ready for use.

Viewing Disaster Recovery Instance

1. Log in to the MongoDB console.
2. In the dropdown list under **MongoDB** in the left sidebar, choose either **Replica Set Instance** or **Shard Instance**. The process for replica sets and sharded instances is similar.
3. At the top of the instance list page, select the region.
4. In the instance list, find the source instance to which the disaster recovery instance belongs.
You can use the search box in the top-right corner of the instance list to search for the target instance by entering the instance ID, instance name, private IP address, or tag key.
If the instance is not found in the instance list, choose **Recycle Bin** from the left sidebar to check whether the instance has been isolated due to expired fees. For more details, see Recycle Bin.
5. In the **Instance ID/Name** column of the source instance, click **Instance ID** to enter the **Instance Details** page.
6. Click the **RO/DR** tab, and then select the **DR Instance** tab.



7. In the disaster recovery instance list, view all disaster recovery instances under the source instance.

Parameter	Description
Instance ID	The disaster recovery instance ID and its name. Click the instance ID in blue to navigate to the disaster recovery instance details page. For more details, see View Instance Details.
Status	The current running status of the instance. The normal status is Running.

Specifications	Instance specification information, including memory and disk capacity.
Latency	The synchronization status and latency of the disaster recovery instance based on the source instance.
Number of nodes	The number of primary and secondary nodes in the disaster recovery instance.
Network	The name of the VPC to which the disaster recovery instance belongs.
Private IP address	The private IPV4 address assigned by the VPC. To access the database, you need to configure the private IP address and port information. For detailed directions, see Connecting Instances.
Region	Information about the region and AZ.
Expiration time	For monthly subscription billing, the specific expiration time of the instance is displayed. For pay-as-you-go billing, this field is blank.
Operations	Click Modify Configuration to adjust the specifications of the disaster recovery instance. When adjusting the specifications of the source instance, be sure to upgrade the disaster recovery instance as well; otherwise, data loss may occur.

Promoting Disaster Recovery Instance to Primary Instance

1. Log in to the MongoDB console.
2. In the dropdown list under **MongoDB** in the left sidebar, choose either **Replica Set Instance** or **Shard Instance**. The process for replica sets and sharded instances is similar.
3. At the top of the instance list page, select the region.
4. In the instance list, find the source instance to which the disaster recovery instance belongs.
You can use the search box in the top-right corner of the instance list to search for the target instance by entering the instance ID, instance name, private IP address, or tag key.
If the instance is not found in the instance list, choose **Recycle Bin** from the left sidebar to check whether the instance has been isolated due to expired fees. For more details, see Recycle Bin.
5. In the **Instance ID/Name** column of the source instance, click **Instance ID** to enter the **Instance Details** page.
6. Click the **RO/DR** tab, and then select the **DR instance** tab.
7. In the disaster recovery instance list, find the instance you want to promote.
8. In the **Operation** column, click **Promote**. In the **Promote DR Instance** dialog box, confirm the prompt information, and click **OK**.
9. The disaster recovery instance will immediately be converted to a regular instance and removed from the disaster recovery instance list.

Related API

API	Description
DescribeDBInstances	Query the list of MongoDB instances.
RenameInstance	Modify the instance name.
ModifyDBInstanceSpec	Adjust the MongoDB instance configuration.

Parameter Configuration
Adjusting Database Parameters

Last updated：2024-10-10 14:47:30

TencentDB for MongoDB allows you to adjust certain database parameters, so that the database features can better adapt to your business needs.

Overview

In the daily Ops process, quickly adjusting database parameters can optimize the query and management performance of the database in a targeted manner for ever-changing business scenarios. In addition, the parameter modification records can be viewed at any time for evidence-based problem locating.

Version Description

Currently, TencentDB for MongoDB 3.2 and later support database parameter modification. However, there are differences in the modifiable parameters on each version as displayed in the console.

Note

Currently, the parameter modification feature only supports parameters that can take effect without requiring a restart after modification. It will support other parameters in future iterations. You can also restart the instance on the MongoDB terminal, but the restart will cause a disconnection. Therefore, make business arrangements in advance and proceed with caution.
When updating the cluster architecture or configuration, such as adjusting specifications, nodes, or shards, upgrading nodes, and migrating nodes, you don't need to configure parameters repeatedly, as the system will automatically sync the parameter configuration data.

Prerequisites

You have created a TencentDB for MongoDB instance. For more information, see Creating TencentDB for MongoDB Instance.
The instance is running normally.

Directions

Querying the parameter configuration

1. Log in to the TencentDB for MongoDB console.
2. On the left sidebar, select **Replica Set Instance** or **Sharded Cluster Instance**. The directions for the two types of instances are similar.
3. In the instance list on the right, find the target instance.
4. Click the target instance ID to enter the **Instance Details** page.
5. Select the **Parameter Settings** tab to view the database parameter configuration.

Modifying the parameter configuration

1. On the **Modifiable Parameters** tab, click **Modify Current Value**.
2. In the input box in the **Current Value** column, set the desired parameter value as shown below:

Note:

You can modify multiple parameters at the same time.

When modifying parameters, be sure to set them according to the **acceptable values**.

In the **Restart upon Modification** column, check whether the instance will be restarted. The restart will cause a disconnection. Therefore, make business arrangements in advance and proceed with caution.

The acceptable values of parameters depend on the instance version and architecture. The parameters that can be modified on the current version are as listed below:

Parameter	Restart Required	Default Value	Acceptable Values	Suppor Version
operation.profiling.slowOpThresholdMs	No	100	[0-65536]	4.0, 4.2 4.4
operationProfiling.mode	No	off	[off | slowOp | all]	4.0, 4.2 4.4

setParameter.cursorTimeoutMillis	No	600000	[1-2147483647]	3.2, 3.6 4.0, 4.2 4.4

setParameter.intenalQueryExecMaxBlockingSortBytes	No	33554432	[33554432-268435456]	4.0, 4.2 4.4

setParameter.maxTransactionLockRequestTimeoutMillis	No	5	[0-60]	4.0, 4.2 4.4
setParameter.transactionLifetimeLimitSeconds	No	60	[5-300]	4.0, 4.2 4.4

setParameter.failIndexKeyTooLong	No	true	[true | false]	3.2, 3.6 4.0

balance.window	No	NULL	[00:00 | 23:00]	4.0, 4.2 4.4
openBalance.window	No	false	[true | false]	4.0, 4.2 4.4

3. Click **OK**.

Querying the parameter modification record

1. On the **Parameter Settings** tab, click **Modification Log**.
2. View the parameter modification log, including values before and after modification, modification status, and modification time.

Recycle Bin

Last updated：2024-01-15 14:40:06

Terminated instances will be put into the recycle bin and can be restored.

Overview

Tencent Cloud recycle bin offers a mechanism for repossessing cloud services. If your account balance is sufficient, you can restore terminated instances that are still in the recycle bin.

Version Description

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, and 5.0 support instance repossession.

Note

The repossession of instances in different billing modes is as described below:
Monthly Subscribed Instances in the Recycle Bin
Pay-as-You-Go Instances in the Recycle Bin
Retention period: Instances will be retained in the recycle bin for 7 calendar days.
Expiration processing: Instances that are not renewed within 7 calendar days will be released and cannot be restored.
Note:
The system will send you a renewal notification 7 days before the expiration of a TencentDB instance. On the 8th day after expiration, the instance will become unavailable and be moved to the recycle bin.

Retention period: If your account has no overdue payments, terminated instances will be retained in the recycle bin for 3 days.
Expiration processing: Instances that are not renewed before the retention period ends will be released and cannot be restored.
Note:
After the account balance becomes 0, instances will be automatically shut down and moved from the instance list to the recycle bin, and the billing will stop in 24 hours.
You cannot restore pay-as-you-go instances from the recycle bin if your account has overdue payments. You need to top up your account first.

Pay-as-you-go instances are retained in the recycle bin for a maximum of 3 days. You need to top up your account in time to restore the instances.

Prerequisites

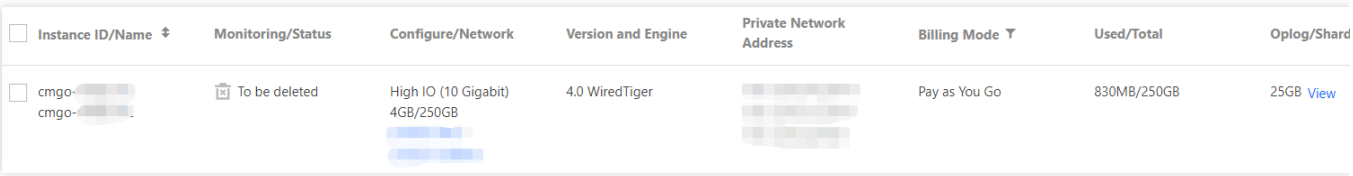
The TencentDB for MongoDB instance has been terminated.
Your Tencent Cloud account balance is sufficient.

Directions

Instances in the recycle bin can be **renewed**, **restored**, or **eliminated**.

Viewing an instance in the recycle bin

1. Log in to the TencentDB for MongoDB console.
2. On the left sidebar, select **MongoDB** > **Recycle Bin**.
3. Above the **Instance List** on the right, select the region.
4. On the **Recycle Bin** page on the right, you can see the list of instances in the recycle bin.



Restoring one instance

1. In the instance list in the recycle bin, find the target instance and click **Restore** in the **Operation** column.
2. In the **Restore Instance** window, confirm the instance information and click **OK**.

The instance will return to their original instance list from the recycle bin.

Batch restoring instances

1. In the instance list in the recycle bin, select the target instances.
2. Click **Batch Restore** above the list, confirm the instance information in the **Restore Instance** window, and click **OK**.

The instance will return to their original instance list from the recycle bin.

Eliminating an instance

1. In the instance list in the recycle bin, find the target instance and click **Eliminate Now** in the **Operation** column.
2. In the **Instance Elimination** window, confirm the instance information and click **OK**.

Note:

The instance will be completely eliminated, and its data will not be recoverable. Therefore, you need to back up the data in advance.

Task Management

Last updated：2024-01-15 14:40:06

TencentDB for MongoDB allows you to intuitively and quickly track the task execution progress in the console.

Background

Daily OPS involves massive and diverse tasks. Task management can help you quickly and efficiently find tasks and stay up to date with their execution status.

Version Description

Currently, TencentDB for MongoDB 4.4, 4.2, 4.0, 3.6, and 3.2 support viewing task execution records.

Prerequisites

You have applied for a TencentDB for MongoDB instance.
The TencentDB for MongoDB replica set or sharded instance is in **Running** status.

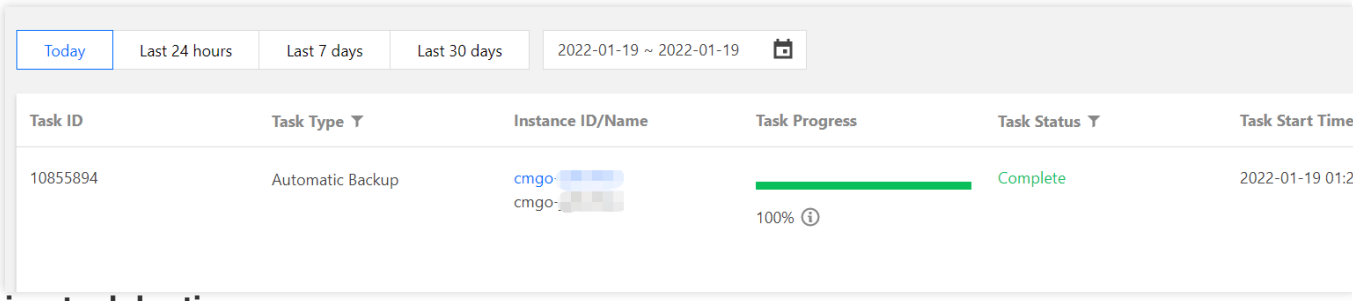
Directions

You can view the task records and details in the console.

Viewing task record

1. Log in to the TencentDB for MongoDB console.
2. On the left sidebar, select **MongoDB** > **Task Management**.
3. Above the instance list on the right, select the region.
4. On the **Task Management** page on the right, you can see all task records.

Hover over the **Task Progress** progress bar to view the task execution process.



Filtering task by time

1. Above the task list, you can select **Today**, **Last 24 hours**, **Last 7 days**, **Last 30 days**, or a time period to filter the tasks to be viewed.
2. In the task list, find the task record to be viewed.

Filtering task by instance name

1. In the search box in the top-right corner of the task list, you can filter the tasks to be viewed by instance name.
2. In the task list, find the task record to be viewed.

Viewing task details

1. In the task list, find the target task and click **Task Details** in the **Operation** column.
2. In the **Task Details** window, view the task execution details.
3. Then, click **Close**.

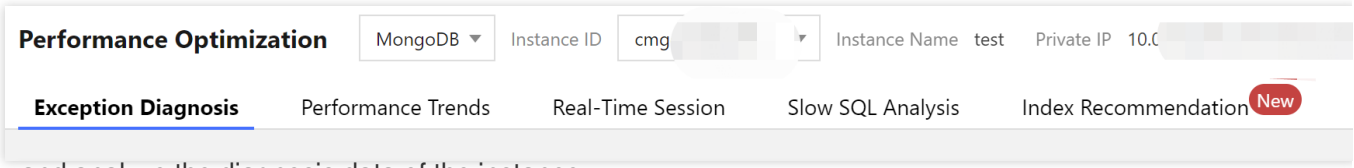
Performance Optimization

Last updated：2024-01-15 14:40:06

TencentDB for MongoDB is connected to the performance optimization feature of DBbrain. The feature monitors and diagnoses database instance exceptions in real time, automatically generates health reports, and gives expert optimization suggestions. This helps you stay on top of the running status of the current database, quickly locate and troubleshoot issues, and promptly optimize the database performance.

Viewing the performance optimization

1. Log in to the TencentDB for MongoDB console.
2. On the left sidebar, select **Performance Optimization**.
3. At the top of the **Performance Optimization** page of **DBbrain**, select the target instance in the **Instance ID** drop-down list.



4. View and analyze the diagnosis data of the instance.

Monitoring Type	Description
Exception diagnosis	Performs real-time performance monitoring and health inspections on the database and gives diagnosis prompts and optimization suggestions for failures.
Performance trends	Monitors performance metrics of instances and Mongod nodes, such as resources, requests, and primary/secondary delay.
Real-time session	Collects the information of database client sessions in real time, such as the sources and number of sessions as well as the number of active sessions.
Slow log analysis	Analyzes the number and duration of slow queries of instances and Mongod nodes in real time.
Space analysis	Analyzes the database space utilization, including the sizes of data and logs, the daily increase in space utilization, and the estimated number of available days.
MongoStatus	Collects and analyzes the number of requests, updates, deletions, and connections as well as outbound/inbound traffic of the database.
MongoTop	Collects the top data of the database in terms of write operation, read operation, and total

	request duration.
SQL throttling	Controls scenarios where excessive CPU resources are consumed due to high traffic. You can create SQL throttling tasks to control the number of access requests and SQL concurrency, thereby ensuring a high service availability.
Index recommendation	Collects the real-time information of slow queries, automatically analyzes it, and recommends the optimal global index.
Health report	Scores the instance health based on monitoring metrics and statistics.

Data Migration Guide
Creating a DTS Migration Task
Migration Overview

Last updated：2024-10-10 15:05:42

Migration Service

Tencent Cloud Data Transmission Service DTS is an integrated database data transmission service that provides data migration, synchronization, and subscription. It enables users to easily migrate their databases to the cloud without interrupting business operations. DTS for MongoDB supports one-time migration to cloud databases, allowing for both full and incremental data migration, meaning that it transfers both historical data from the source database and any new data written to the source database during the migration process.

Migration Overview

Content Overview	Note
Supported Features	Introduces the features currently supported by DTS for MongoDB and the applicable use cases.
Usage Instructions	Describes important considerations and related limitations during the use of DTS for MongoDB.
Migration Operation Guide	Describes the specific operations for using DTS to perform migrations.
Pre-check Failure Handling	Before a DTS migration task is started, performing necessary pre-checks is crucial to prevent and resolve potential issues such as database connection problems or table conflicts. This document introduces common issues encountered during the pre-check process.

Supported Capabilities

Last updated：2024-10-10 14:56:22

Supported Scenarios and Versions

Support migration between different architectures: Replica Set > Replica Set / Sharded Cluster, Sharded Cluster > Replica Set / Sharded Cluster, Single Node > Replica Set / Sharded Cluster.

Source Database	Target Database	Scene Description
Self-hosted MongoDB database (Self-hosted on IDC, Self-hosted on CVM) 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 4.0, 4.2, 4.4, 5.0, 6.0	TencentDB for MongoDB 4.0, 4.2, 4.4, 5.0, 6.0	Migrate on-premises databases to the cloud.
Third-Party cloud provider MongoDB 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 4.0, 4.2, 4.4, 5.0, 6.0	TencentDB for MongoDB 4.0, 4.2, 4.4, 5.0, 6.0	Migrate databases from other cloud providers to TencentDB for MongoDB.
TencentDB for MongoDB 3.2, 3.6, 4.0, 4.2, 4.4, 5.0, 6.0	TencentDB for MongoDB 4.0, 4.2, 4.4, 5.0, 6.0	Migration between Tencent Cloud regions or across different regions Database migration within the same Tencent Cloud root account or between different root accounts Cross-version migration between TencentDB for MongoDB instances Migration between Replica Set and Sharded Cluster within TencentDB for MongoDB instances

Supported Features

Feature Category	Feature Sub-Item or Description	Supported Capabilities
Migration Object	-	Databases, collections
Migration Type	-	Full migration Full + Incremental migration

Key Task Management Operations	Retry	Supported
	Create similar task	Supported
Incremental Synchronization	DML synchronization (INSERT / UPDATE / ELETE)	Supported
	DDL synchronization	INDEX: createIndexes, createIndex, dropIndex, and dropIndexes COLLECTION: createCollection, drop, collMod, renameCollection, and convertToCapped DATABASE: dropDatabase and copyDatabase Support DDL operations for both Replica Set and Sharded Cluster.
Consistency Check	Validation object	Entire migration object / Custom object
	Validation method	Row count comparison / Content validation / Sampling comparison

User Guide

Last updated：2024-10-10 14:58:25

Impact on the Source Database

When DTS performs full data synchronization, it consumes certain resources from the source database, which may lead to an increase in the source database's load and put additional pressure on it. If your database configuration is insufficient, it is recommended to perform data migration during off-peak hours.

Impact on the Target Database

During the migration process, DTS will use a system service account to create a table under the TencentDTSData database on the target end, using the task ID (e.g., table name dts-xxxxx). This table is used to record checkpoints, enabling checkpoint restart in case of task interruption.

Migration Architecture

1. Sharded Migration Related Notes:
1.1 Before a sharded cluster is migrated, it is recommended to clean up orphaned documents in the source cluster in advance. Otherwise, data inconsistency issues may arise after migration. For instructions on how to clean up orphaned documents, see the MongoDB official documentation cleanupOrphaned.
1.2 During a sharded migration, do not enable sharding on the source databases and tables being migrated. This is to prevent discrepancies in data distribution between the source and target databases. If sharding is enabled on the source during migration, check the sharding status on the target. If sharding is not enabled on the target, you will need to manually enable it. For detailed instructions on how to enable sharding, see the MongoDB official documentation on Shard a Collection.
1.3 If the source is a TencentDB for MongoDB 3.2 sharded cluster, all shard keys will be treated as hashed shard keys by default during migration. If you want to use range shard keys on the target, create the range shard keys on the target database in advance before the data migration is started.
2. Since a single node does not have an Oplog, incremental migration is not supported when the self-hosted instance is a single node.

Notes

1. Do not perform the following operations during migration, as they may cause the migration task to fail.
Do not modify or delete user information (including usernames, passwords, and permissions) or port numbers in the source or target databases.
Do not perform Oplog clearance operations on the source database.
During the data migration, do not delete the TencentDTSData database on the target database.
2. Be cautious when you operate on the target database during the data migration to avoid data inconsistencies.

Migration Operation Guide

Last updated：2024-10-10 15:00:17

Overview

DTS-based MongoDB data migration supports both full and incremental data migration, meaning that the historical data from the source database as well as the newly written data during the migration process can be migrated together.
This document provides instructions on how to use the DTS data migration feature to migrate data from MongoDB to TencentDB for MongoDB.

Preparation

1. Ensure that the access channel between DTS and the database is established in advance according to the type of access you plan to use. For detailed instructions, see Network Preparation Work.
IDC self-hosted databases/other cloud provider databases: The available access methods include Public Network/Direct Connect/VPN Access/CCN.
Self-built databases on CVM: The access method should be Self-Build on CVM.
TencentDB instances: The access method should be Database.
2. It is recommended to create a read-only account in the source database for migration purposes. See the following method.

```
use admin
db.createUser({user: "username",pwd: "password",roles:[{role: "readAnyDatabase", db
```

3. If the target database is a TencentDB instance, you can use the mongouser account for migration, or you can create your own account. The method for creating an account is as follows.

```
db.createUser({user:"username",pwd:"password",roles:[{role:"readWriteAnyDatabase",d
```

Operation Steps

1. Log in to the DTS console, choose the **Data Migration** page in the left sidebar, and click **Create Migration Task** to enter the Create Migration Task page.
2. On the Create Migration Task page, select the source instance type and region, the target instance type and region, and the specifications. Then, click **Purchase Now**.

Configuration Parameter	Description
Creation Mode	Create new task: Create a brand new task. Create similar task: Quickly create a task with the same configuration as a previous task. In the new task, the options for database type, access method, billing mode, and migration type are pre-filled to match the previous task. Users can modify these options as needed based on the current situation.
Source Instance Type	Select based on the type of your source database. This cannot be changed after purchase. For this scene, select MongoDB.
Source Instance Region	Select the region where the source database is located. If the source is a self-hosted database, select the region closest to the self-hosted database.
Target Instance Type	Select based on the type of your target database. This cannot be changed after purchase. For this scene, select MongoDB.
Target Instance Type	Select the region where the target database is located.
Version	The default is NewDTS; no modification is needed.
Specification	Currently, only the Medium specification is supported.
Task Name	Select **Name After Creation**. The task name will be the same as the task ID by default. After the migration task is created, you can rename the task. Select **Name Now**, and enter the task name in the input field below.

3. After the purchase is completed, the page will automatically redirect to the data migration task list. Select the task you just purchased to proceed with the configuration.

If you have purchased multiple regions or are configuring cross-region tasks, the task list will display tasks based on the region of the target instance. You can switch regions at the top of the page to find the tasks you have purchased.

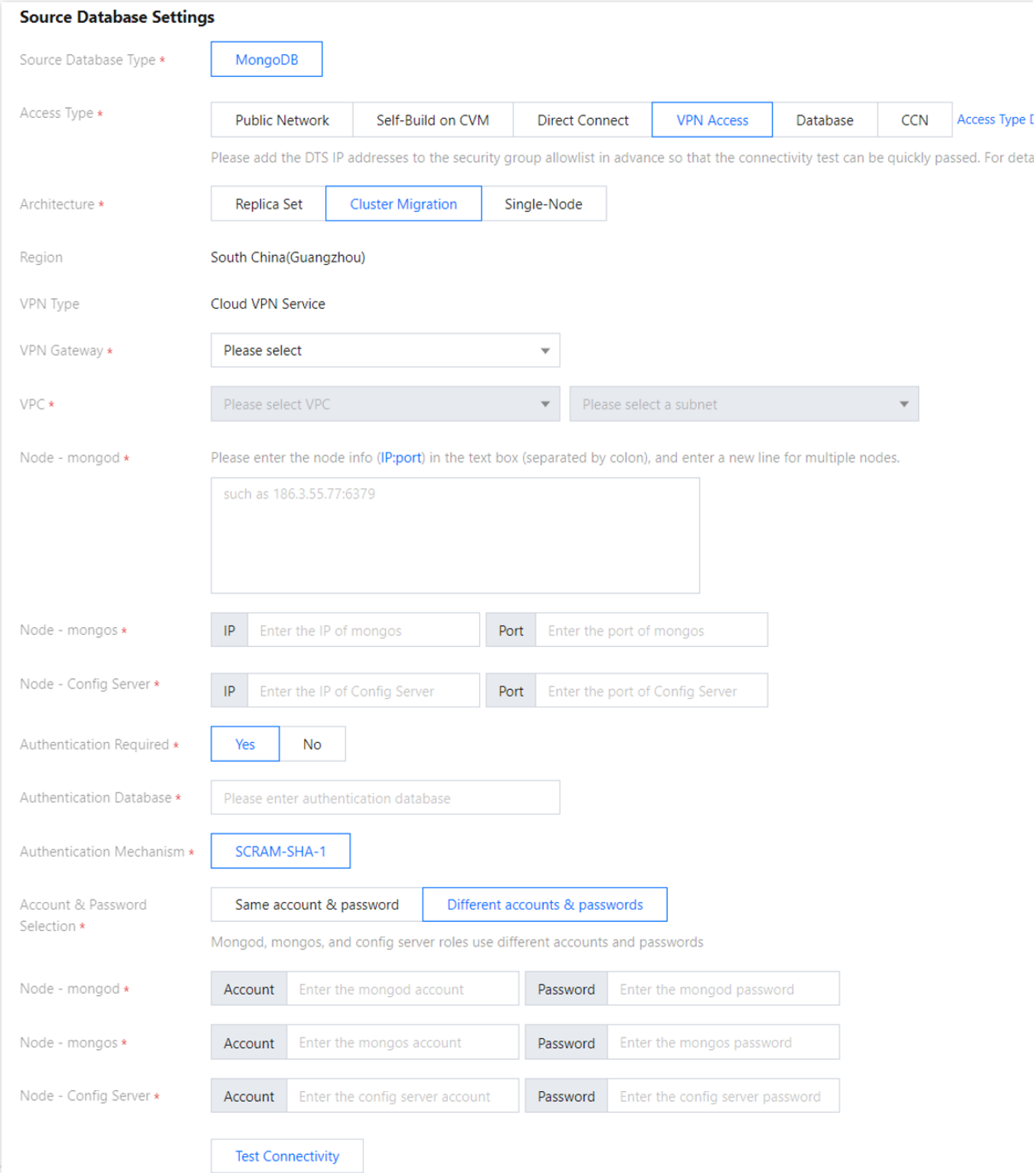
4. On the set source and target database page, complete the task settings, source database settings, and target database settings.

Note:

Enter the read-only account that was previously created for the source database; otherwise, the pre-check step will fail.

If the source or target database is a TencentDB instance, DTS will use a system service account to export or write data during the migration process. For example, if the source database is a TencentDB instance, DTS will connect to

the source database using the read-only account provided by the user, and also use a system service account to export data from the source. Similarly, if the target database is a TencentDB instance, DTS will connect to the target database using the account provided by the user, and use a system service account to write data to the target.



Configuration Parameter	Description
Task Name	Set a business-relevant name for easy task identification.

Run Mode	Immediate execution: The task will start immediately after the pre-check passes. Scheduled execution: Set a specific start time for the task. The task will not start immediately after the pre-check passes, but will instead start at the scheduled time.
Access Type	Select based on your specific scene. For preparation steps related to different access types, see the Overview. Public Network: The source database can be accessed via a public IP. Self-Build on CVM: The source database is deployed on a Tencent Cloud Service CVM. Direct Connect: The source database can connect to Tencent Cloud Virtual Private Cloud via Direct Connect. VPN Access: The source database can connect to Tencent Cloud Virtual Private Cloud via VPN Connections. Database: The source database is a TencentDB instance. CCN: The source database can connect to Tencent Cloud Virtual Private Cloud via Cloud Connect Network. If the source database is a self-hosted IDC database or a database from another cloud provider, you can choose Public Network/Direct Connect/VPN Access/CCN as the access method. If the source database is self-hosted on a CVM, select Self-Hosted on CVM as the access method. If the source database is a TencentDB instance, select Database as the access method.
Architecture	This parameter is displayed when **Access Type** is set to Self-Build on CVM/Direct Connect/VPN Access/CCN. Select it according to your actual situation. Replica set: Refers to an architecture where the source database is a replica set, consisting of one Primary node and one or more Secondary nodes. Cluster migration: Refers to an architecture where the source database is a sharded cluster, consisting of components such as mongos nodes, config servers, and shard nodes. Single node: Refers to an architecture where the source database cluster consists of a single node that handles both read and write operations. **Note:** Once you select an architecture type, and the connectivity test passes. It cannot be changed to another architecture type; otherwise, the task will encounter errors.
Cluster Migration	When **Architecture** is set to Cluster migration, the following parameters need to be configured. Node - mongod: Enter the IP and port, or the domain name and port, of the mongod node. For multiple nodes, enter each on a new line; only one mongod needs to be entered for each shard. Example: 1xx.xx.55.77:6xx9. Node - mongos: Enter the IP and port, or the domain name and port, of the mongos node. Node - Config Server: Enter the IP and port, or the domain name and port, of the Config Server node.
Public Network	When **Access Type** is set to Public Network, the following parameters need to be configured.

	Host address: The IP address or domain name of the source database. Port: The port used by the source database.
Self-Build on CVM	When **Access Type** is set to Self-Build on CVM", the following parameters need to be configured. CVM instance: The instance ID of the CVM. Port: The port used by the source database.
Direct Connect	When **Access Type** is set to Direct Connect, the following parameters need to be configured. VPC Direct Connect Gateway: Only the VPC Direct Connect Gateway is supported for Direct Connect. Confirm the network type associated with the gateway. VPC: Select the VPC and subnet. Host address: The IP address or the domain name of the source database. Port: The port used by the source database.
VPN Access	When **Access Type** is set to VPN Access, the following parameters need to be configured. VPN gateway: Select the VPN gateway instance that is accessed through the VPN gateway. VPC: Select the VPC and subnet. Host address: The IP address or the domain name of the source database. Port: The port used by the source database.
Database	When **Access Type** is set to Database, the following parameters need to be configured. Cloud database instance: Select the instance ID of the source database.
CCN	When **Access Type** is set to CCN, the following parameters need to be configured. When using CCN for access, both same-account CCN and cross-account CCN are supported. Due to the complexity of network configuration, see Migrating Self-built Databases to Tencent Cloud via Cloud Connect Network. Host network environment: Select according to your actual situation. For example, if the source database is a TencentDB instance, select Tencent Cloud; if the source database is a self-hosted IDC database, select Self-built IDC; if the source database is from another cloud provider, select the corresponding network. Host address: The host IP address or the domain name of the source database. Port: The port used by the source database. CCN instance account My account: The CCN resources and DTS belong to the same Tencent Cloud root account. Other account: The CCN resources and DTS belong to different Tencent Cloud root accounts. VPC CCN: The name of the CCN instance. Access VPC: Select the VPC and subnet to be accessed. The Access VPC refers to the VPC that is connected to the DTS migration linkage within the CCN. Choose a VPC other

	than the one that the source database belongs to from all the VPCs associated with the CCN. Access VPC region: The region of the Access VPC should match the region of the source database selected when purchasing the task. If they do not match, DTS will change the region of the source database selected in the purchased task to the Access VPC region.
Authentication Required	Whether security authentication for the username and password of the source database is required. If Required is selected, the following parameters need to be filled in. Authentication database: The name of the database that requires authentication, i.e., the database name associated with the account executing the migration task. Only admin is supported. Authentication mechanism: Currently, only SCRAM-SHA-1 is supported. Account and password options Same account and password: Select this parameter if the same account and password are used for the roles of mongod, mongos, and Config Server, and enter the unified account and password. Different accounts and passwords: Select this parameter if different accounts and passwords are used for the roles of mongod, mongos, and Config Server, and enter the account and password for each mongod, mongos, and Config Server separately.
Account/Password	Account/Password: The account and password of the source database.

5. Test the connectivity between the source and target instances.

If the connectivity test fails, see Connectivity Test Failure for troubleshooting.

6. On the Set migration options and select migration objects page, configure the migration options and select the migration objects.

Configuration Parameter	Description
Migration Type	Select based on your scene. Full migration: Migrates the entire database, including only the data that exists in the source database at the time the task is initiated, and not including any new data written to the source database after the task starts. Full + Incremental migration: Migrates both the existing data in the source database at the time the task is initiated and any new data written to the source database in real-time after the task starts. If data is being written to the source database during the migration and you need a seamless migration without downtime, select this option.
Data Consistency Check	Data consistency check is supported only when the **Migration Type** is configured as Full + Incremental Migration, allowing a detailed comparison of data between the source and target databases after migration. When the **Migration Type** is configured as Full Migration, consistency check is disabled by default. Full object consistency check: After the incremental synchronization is completed, DTS will automatically trigger a consistency check task.

	No check: Data consistency will not be checked. If needed, you can manually trigger a consistency check after the incremental synchronization is completed. For details, see Creating Data Consistency Check Task.
Data Check	When **Data Consistency Check** is set to Full migration object detection, the default consistency check type displayed is Content Validation.
Migration Objects	Entire instance: Migrates the entire instance, excluding system databases, such as system objects in PostgreSQL. However, it will migrate roles and user metadata definitions. Specified objects: Migrates only the specified objects.
Specified Objects	Select the objects to be migrated from the source database and move them to the Selected Object box.

7. On the validation task page, complete the pre-migration validation, and then click **Start Task**.

If the validation task fails, you can see Handling Methods for Validation Failure to resolve the issue and then reinitiate the validation task.

Failed: Indicates that the validation item did not pass the check, blocking the task. The issue should be fixed before re-running the validation task.

Warning: Indicates that the validation item does not fully meet the requirements. You can continue the task, but it may have some impact on the business. The user needs to evaluate whether to ignore the warning or fix the issue before proceeding, based on the provided information.

8. Return to the migration task list, where the synchronization task will be running.

If you need to view task details, delete tasks, or perform other actions, click the corresponding task operation. For more information, see Task Management. If a task error occurs, see Error Handling.

9. End the task.

Select **Full migration**: The task will automatically end upon completion, with no need for manual intervention.

Select **Full + Incremental migration**: After the full migration is completed, the task will enter the incremental data synchronization phase. Incremental data synchronization will not end automatically and requires the user to manually end the task.

Once incremental synchronization is completed (i.e., the status is Ready to Complete) and the time delay between the target and source databases is 0 seconds, click **Complete** in the **Operation** column to end the migration task.

10. (Optional) If a cutover is required, you can perform the official cutover of your business once the task status changes to **Task successful** after the task is completed. For more details, see Cutover Instructions.

Fix for Verification Failure
MongoDB Connection Check

Last updated：2024-10-10 15:03:35

Check Details

The source and target databases need to be normally connected, and if not, a connection failure will be reported.

Causes

The network or server where the source database resides has a security group or firewall configured. For more information, see Failed Connectivity Test > Security Group or Firewall Configured in Network or Server of Source Database.
The source IP addresses are blocked by the source database. For more information, see Failed Connectivity Test > Source IP Addresses Blocked in Source Database.
The network port is closed. For more information, see Failed Connectivity Test > Closed Network Port.
The database account or password is incorrect.

Troubleshooting

Refer to the causes above based on the actual scenario and troubleshoot as instructed.

Database/Table Content Conflict Check

Last updated：2024-10-10 15:07:48

Check Details

In MongoDB data migration scenarios, the target database can contain collections with the same name as those in the source database, but the collections must be empty.

Troubleshooting

If a conflict causes an error, you can delete collections with the same name in the target database or clear their data.

Source Database Node Role Check

Last updated：2024-10-10 15:09:34

Check Details

Check requirements: in a MongoDB migration task, if the source database is a sharded database, you need to enter the `mongos`, `config server`, and `mongod` node information.
Check description: the information of the `mongos`, `config server`, and `mongod` nodes cannot be disordered; otherwise, data migration will also be disordered; for example, the `mongos` node information should not be entered in the box for the `mongod` node. Note that you only need to enter the information of one `mongod` node for each shard.

Fix

Enter the correct node information in the DTS task configuration items.
Enter only one `mongod` for each shard.

Oplog Check

Last updated：2024-10-10 15:11:03

Check Details

Check requirements: oplogs can be obtained from the source database during full + incremental migration.
Check description: incremental migration requires oplogs for replay. If the `oplog.rs` or `oplog.$main` table does not exist in the source `local` database, oplogs cannot be obtained.

Fix

Start the source database as a replica set or in primary/secondary mode to ensure that oplogs can be generated for operations and recorded in the source `local` database.

Source/Target Database Account Permission Check

Last updated：2024-10-10 15:12:41

Check Details

Check whether you have the operation permissions of the database as described below:
Permission requirements for data migration: Migration from MongoDB to TencentDB for MongoDB.

Troubleshooting

If you don't have the operation permissions, get authorized based on the permission requirements in the check details, and run the verification task again.

Database Version Check

Last updated：2024-10-10 15:14:36

The source and target database versions must be supported by MongoDB.

Database Capacity Check

Last updated：2024-10-10 15:16:09

Check Details

In MongoDB data migration scenarios, the storage space of the target database needs to be at least 1.3 times the size of the collections to be migrated in the source database.

Troubleshooting

Delete some data from the target database to free up space.

Upgrade the storage specification of the target database to use an instance with a larger capacity for migration.

Target Database Load Check

Last updated：2024-10-10 15:17:46

Check Details

Check requirements: DTS migration will increase the load in the target database. If there is a business running in the target database during migration, a verification warning will be triggered. It will not block the task but will affect the business. You need to assess and determine whether to ignore the warning.
Impact on business: MongoDB DTS uses logical sync for data migration, which will cause certain pressure on the CPU load of the target database. If there is a business running in the target database, you need to assess and initiate the migration task with caution.

Fix

Stop any business running in the target database and run the verification task again.

ShardKey Check

Last updated：2024-10-10 15:19:15

Check Details

Check requirement: if the target database is a sharded instance, you can preset the shardkey in it. If the table shardkeys in the source and target databases are different, an warning will be triggered. It will not block the task but will affect the business. You need to assess and determine whether to ignore the warning.
Impact on the business: if some shardkeys are different, the migration or sync task will fail.

Fix

If the target database has preset shardkeys, run the following command to shard the source database:

```
sh.shardCollection("<database>.<collection>", { <shard key> : "hashed" } , false, {
```

Run the verification task again.

Source Database Balancer Check

Last updated：2024-10-10 15:21:20

Check Details

Check requirements: if the source database is a sharded instance, you need to disable the balancer in it before you can start migration.
Check description: an incremental migration task will get the oplog. If the balancer is enabled, `moveChunk` in the source database may cause data inconsistency in the target database.

Fix

1. Log in to the source database.
2. Run the following command to disable the source database balancer:

```
sh.stopBalancer()
sh.getBalancerState()
```

3. Run the verification task again.

Time Series Collection Verification

Last updated：2024-10-10 15:22:55

Verification Details

MongoDB 5.0 and later versions support time series collection. When users migrate from version 5.0 and later to an earlier version, if the source database has time series collection, this verification item fails.

Fixing Solution

In scenes of migrating from version 5.0 and later to earlier versions, when users configure the task and select migration objects, they only choose the non-time series collection.

Compression Algorithm Verification

Last updated：2024-10-10 15:24:07

Verification Details

Verify whether the compression algorithms used by the source and target databases are the same. If they are different, a warning is generated. The warning will not block the migration, and the user can ignore the warning to proceed with the task.
Note that when the compression algorithm used by the target database is verified, a random system table will be selected for verification. Since the compression algorithm of the system table does not change after modification, the warning might be inaccurate. If you are sure that the new compression algorithm is used in the target database, you can ignore the warning.

Fixing Solution

The disk size occupied by the same data varies under different compression algorithms. If the user wants the target database to use the same compression algorithm as the source database, modify the compression algorithm of the target database.

Consistency Check After Migration Description of Consistency Verification Function

Last updated：2024-10-10 15:28:56

Overview

During data consistency check, DTS compares the collection data between the source and target databases and outputs the comparison result and inconsistency details for you to perform a business cutover stably and reliably.

Notes

1. Data consistency check compares only the objects selected in the source database and objects migrated to the target database. If you write data into the target database during migration, then the written data will not be included in the consistency check.
2. A data consistency check task may increase the load in the source database instance. Therefore, you need to perform such tasks during off-peak hours.
3. A data consistency check task can be executed repeatedly, but one DTS instance can initiate only one such task at any time.
4. If you choose to **complete** or **terminate** a DTS task before a data consistency check task is completed, the check task will fail.
5. When creating a consistency check, the system will automatically create the `dts_verify_result` library on the target end to record content related to the consistency check. The table styles created under the `dts_verify_result` library are as follows:
diff_5xxxxxxxx4231: Saves inconsistent data detected
diff_meta_5xxxxxxxxx4231: Saves inconsistent metadata detected
result_5xxxxxxxxx4231: Records the results after phase validation
status_5xxxxxxxxx4231: Records validation progress

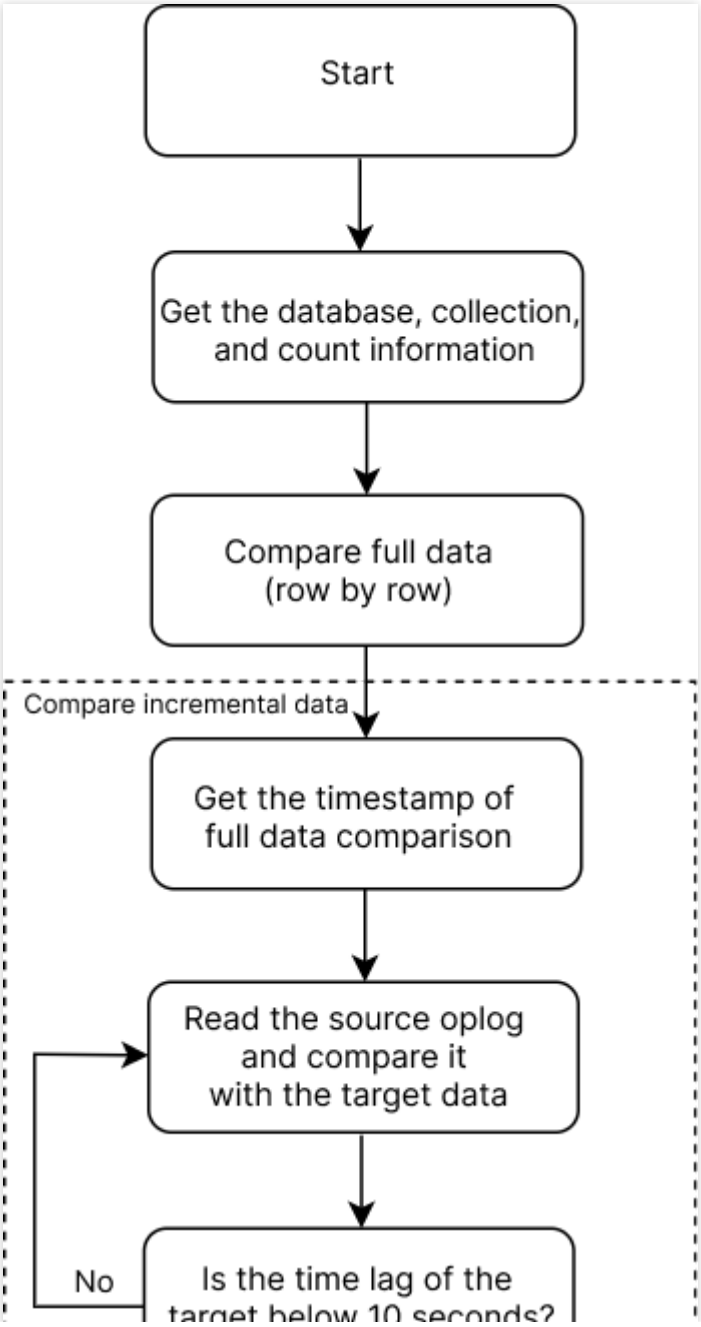
Restrictions

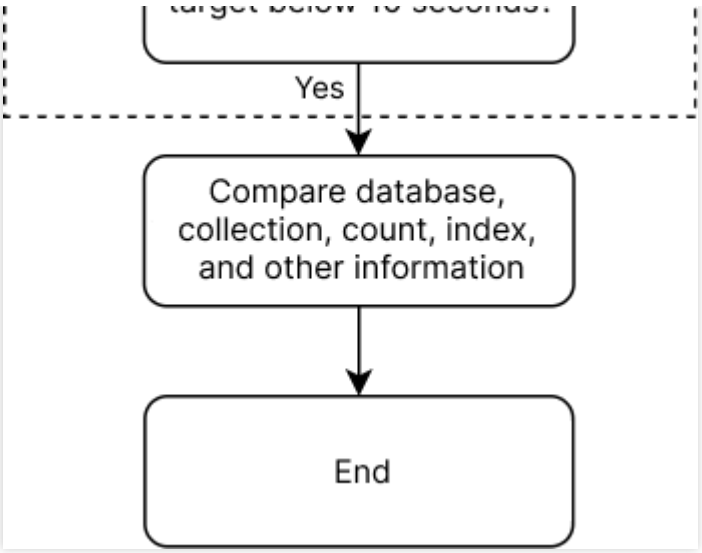
Currently, check tasks are imperceptible to the DDL operations. If you perform DDL operations in the source database during migration, the check result will be inconsistent with the actual data, and you need to initiate another check task

to get the accurate comparison result.

Check Scheme

DTS checks and compares all the data migrated during full migration and incremental migration from the source database. A full data check compares the data in the source and target databases row by row. Once the thread of the incremental data check finds that the full data comparison is completed, it immediately starts the incremental data check to get the start timestamp of the full data check, get the incremental oplog in the source database in a loop, and compare the differences between the source and target databases. When the time lag of data in the source and target databases is below 10 seconds, the comparison ends, and the check result is output.





Creating Data Consistency Check Task

Last updated：2024-10-10 15:35:43

Overview

During data consistency check, DTS compares the table data between the source and target databases and outputs the comparison result and inconsistency details for you to determine the business cutover time. A data consistency check task is independent of the normal business in the source database or other DTS tasks.
Data consistency check tasks can be triggered automatically or created manually.
Automatic triggering: During migration task configuration, if **Full check** is selected for **Data Consistency Check**, a data consistency check task will be triggered automatically when the migration task enters the **incremental sync** step.
Manual creation: When the DTS task enters the **incremental sync** step, you can manually create one or multiple data consistency check tasks.

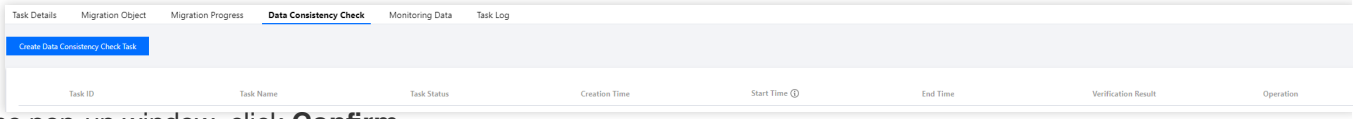
Triggering a data consistency check task automatically

On the **Set migration options and select migration objects** page of a data migration task, select **Full check** for **Data Consistency Check**. In this way, a data consistency check task will be triggered automatically when the migration task enters the **incremental sync** step.
Note:
In this case, the full data and all the database information will be checked by default. If you need to filter check objects, create a data consistency check task manually.

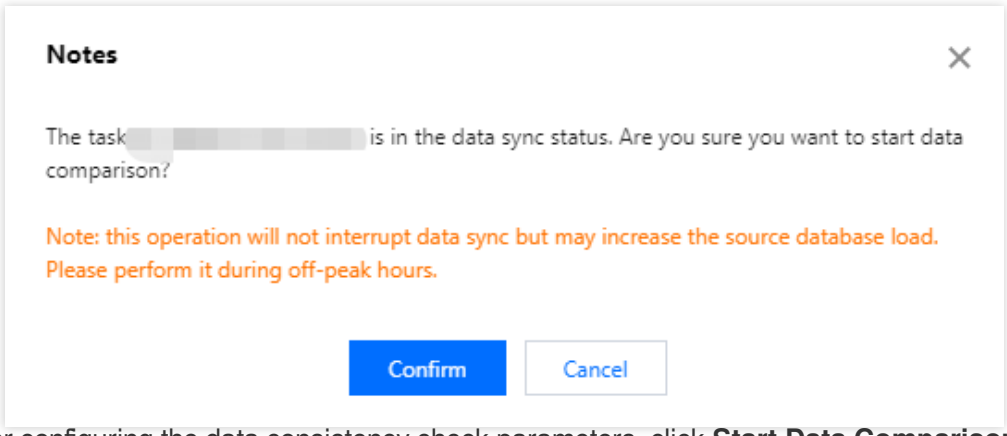
Creating a data consistency check task manually

1. Log in to the DTS console.
2. On the **Data Migration** page, select the target migration task and click **More** > **Create Data Consistency Check Task** in the **Operation** column.
3. Click **Create Data Consistency Check Task**.

Note:
A data consistency check task can be created only when the corresponding DTS task is in the **incremental sync** step. If the button is grayed out, the DTS task status does not meet the requirement; for example, the task has not entered the **incremental sync** step, has failed, or is terminated.



4. In the pop-up window, click **Confirm**.



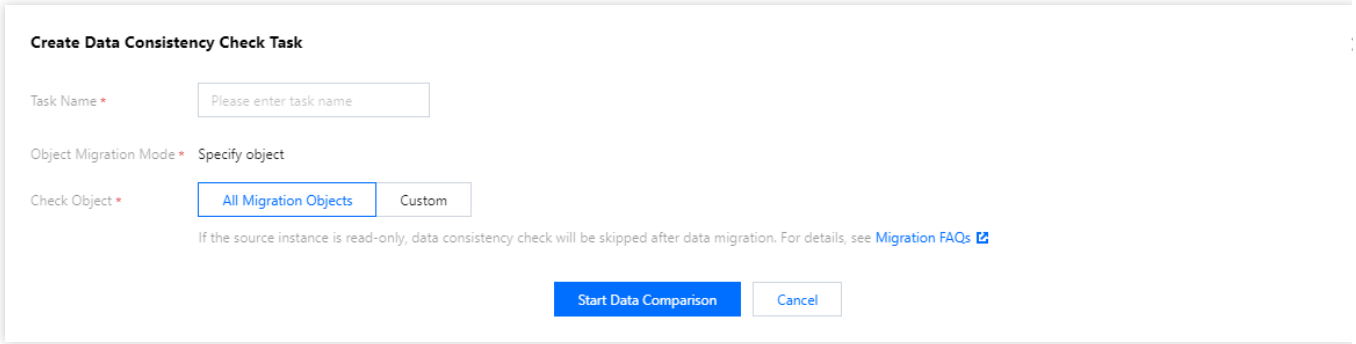
5. After configuring the data consistency check parameters, click **Start Data Comparison**.

Check Object: Select **All Migration Objects** or **Custom**.

Database Information: Select **Index**, **Shard key** (if both the source and target databases are sharded clusters), or **Database and table** for check.

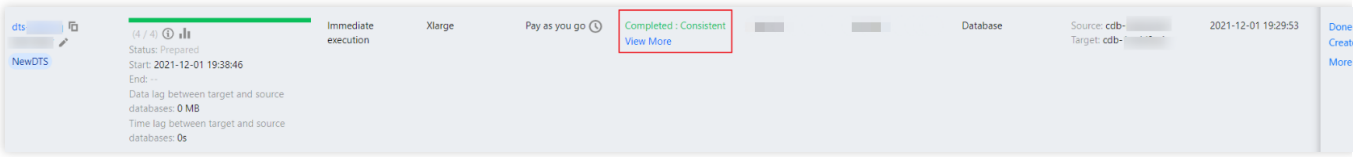
Data Check: The **Row count check** option compares the number of data rows in the source and target databases. The **Content check** option compares the data content of the source and target databases.

Sampling: In scenarios with a high data volume, extracting all the data for check may increase the load of the source database. If you select **Content check**, you can set an appropriate percentage based on your business conditions to extract a certain proportion of data for comparison.

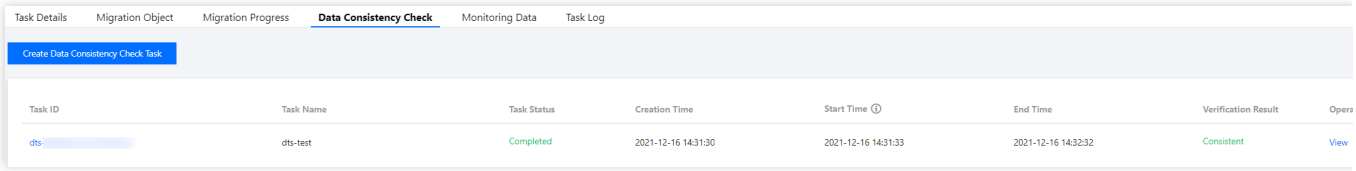


Viewing the data consistency check result

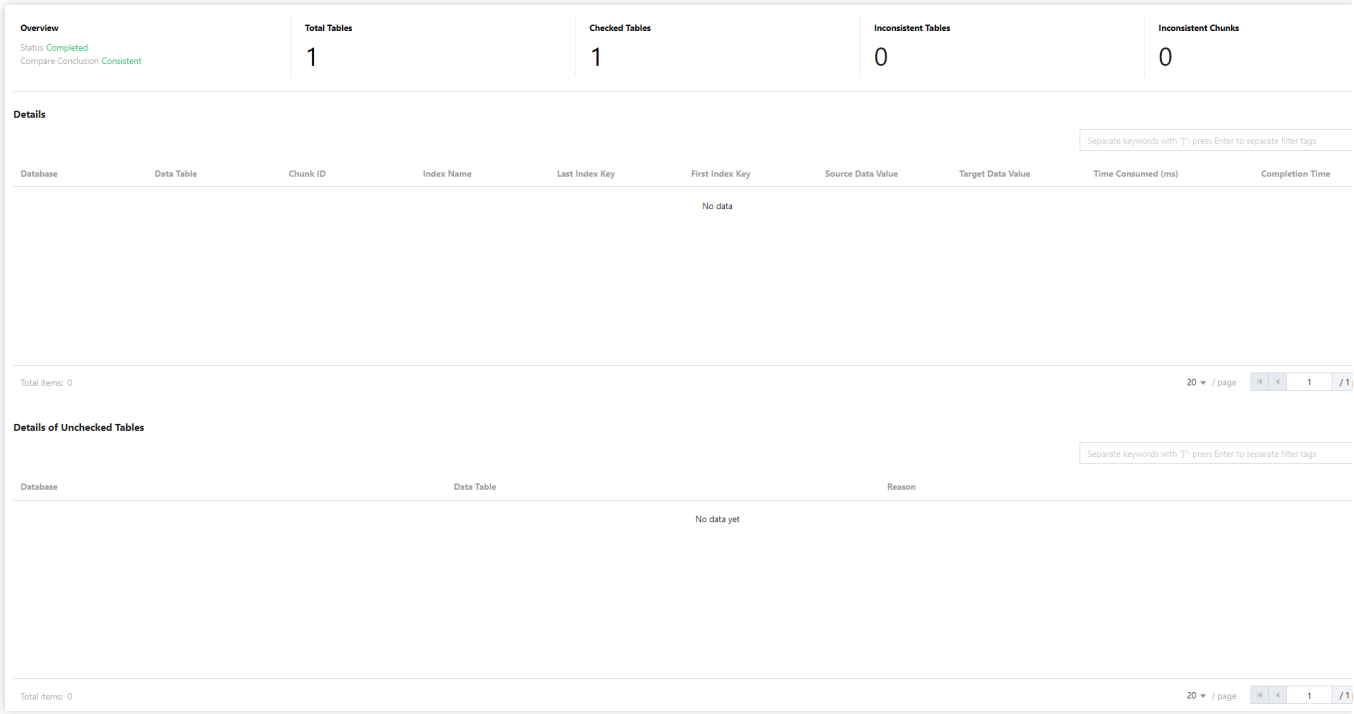
1. On the migration task homepage, view whether the check result is **Consistent** or **Inconsistent** in the **Last Check Result** column. Click **View More** to enter the **Verification Details** page.



2. Click **View** to view the check result.



If the data is consistent, the result will be like:



Inconsistency check result:

Note：

For inconsistent data, you need to manually confirm the corresponding data content of the source and target databases as prompted. For more details, please refer to Common Consistency Check Issues.

Common Consistency Verification Issues

Last updated：2024-10-10 15:31:46

Data content inconsistency

Issue

Cause analysis

During a full data check, data is continuously written to the source database, and the oplog keeps being updated. The incremental data check task continuously reads the oplog from the source database. If the new oplog generated in the source database has not reached the timestamp in the target database, the data content may be inconsistent for a short period of time, which is normal.

Solution

You can check the inconsistent data row by row. You can also initiate a new check task to perform another manual check. When all the incremental data is migrated to the target database, the content will be consistent.

Data row count inconsistency

Issue

Cause 1

During a full data check, data is continuously written to the source database, and the oplog keeps being updated. The incremental data check task continuously reads the oplog from the source database. If the new oplog generated in the source database has not reached the timestamp in the target database, the number of data rows may be inconsistent for a short period of time, which is normal.

Cause 2

The TencentDB for MongoDB row count check collects the row count in **metadata** through `db.collection.estimatedDocumentCount()` or `db.collection.stats()` for comparison, which may be inconsistent with the actual row count under specific circumstances such as unexpected instance shutdown or the presence of orphaned documents.

Solution

In this case, you can use `db.collection.countDocuments()` to compare the row count. Note that this requires collection scanning, which may affect the performance. For more information, see db.collection.countDocuments().

Index check

Issue

If you select **Index** for **Database Info** when creating a consistency check task, the indexes in the source and target databases will be compared. You may find that the content of the `v` and `background` fields in the source and target databases are inconsistent, but the check result does not indicate any inconsistency.

Cause analysis

The TencentDB for MongoDB index check policy ignores the inconsistency in the `v` (version information) and `background` (creation in the background) fields, which will not be indicated in the check result.

Creating MongoDB Data Subscription

Last updated：2024-01-15 14:49:55

This document describes how to create a data subscription task in DTS for TencentDB for MongoDB.

Version description

Data subscription is supported for TencentDB for MongoDB 3.6, 4.0, 4.2, and 4.4.
TencentDB for MongoDB 3.6 only supports collection-level subscription.

Prerequisites

You have prepared a TencentDB instance to be subscribed to, and the database version meets the requirements. For more information, see Databases Supported by Data Subscription.
We recommend that you create a read-only account in the source instance by referring to the following syntax. For operations in the console, see Account Management.

```
# Create an instance-level read-only account
use admin
db.createUser({
user: "username",
pwd: "password",
roles:[
       {role: "readAnyDatabase",db: "admin"}
      ]
})

# Create a database-specific read-only account
use admin
db.createUser({
user: "username",
pwd: "password",
roles:[
       {role: "read",db: "Name of the specified database"}
      ]
})
```

Restrictions

Currently, the subscribed message content is retained for 1 day by default. Once expired, the data will be cleared. Therefore, you need to consume the data promptly.
The region where the data is consumed should be the same as that of the subscribed instance.
The Kafka built in DTS has a certain upper limit for the size of processed individual messages. When a single row of data in the source database exceeds 10 MB, this row may be discarded in the consumer.
If the subscribed database or collection specified in the data subscription task is deleted from the source database, the subscription data (change stream) of the database or collection will be invalidated. Even if the database or collection is rebuilt in the source database, the subscription data cannot be resubscribed. In this case, you need to reset the subscription task and select the subscription object again.

SQL operations for subscription

Operation Type	Supported SQL Operations
DML	INSERT, UPDATE, DELETE
DDL	INDEX: createIndexes, createIndex, dropIndex, dropIndexes; COLLECTION: createCollection, drop, collMod, renameCollection; DATABASE: dropDatabase, copyDatabase

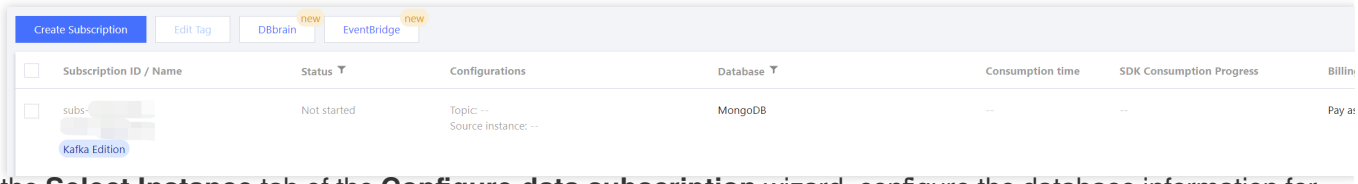
Subscription configuration

1. Log in to the DTS console, select **Data Subscription** on the left sidebar, and click **Create Subscription**.
2. On the **DTS** page, configure parameters as follows and click **Buy Now**.

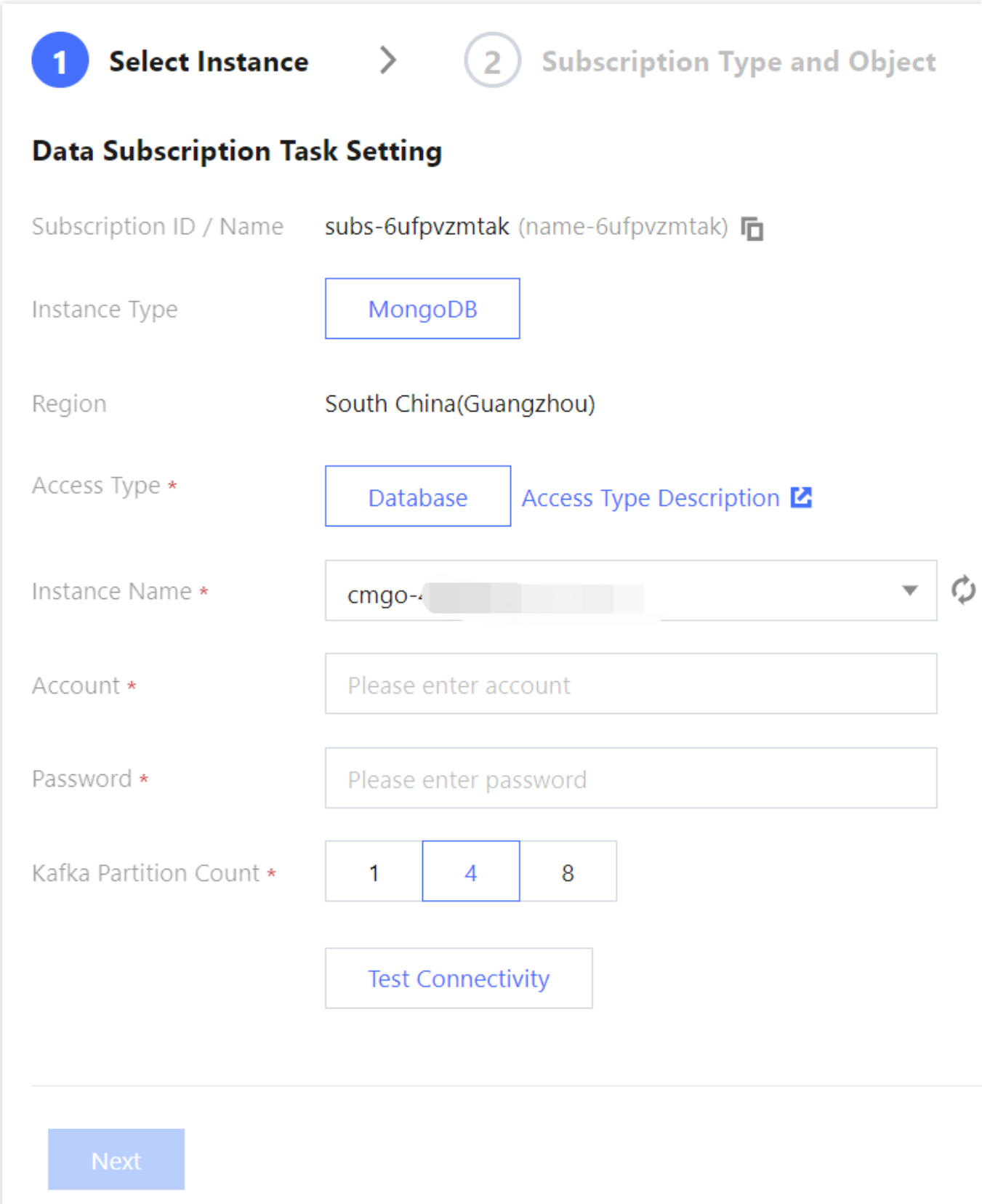
Parameter	Description	Configuration Method
Service Type	Select the service type. This document describes the **Data Subscription** service.	Select **Data Subscription**.
Billing Mode	Select the billing mode of the service. For billing details, see Billing Overview.	**Pay-as-you-go** is supported. For more information, see Billing Overview.
Region	Select the region where the subscription service resides.	It must be the same as that of the database instance to be subscribed to.
Database	Select the type of the database for the data subscription service.	Select **MongoDB**.
Version	Data can be directly consumed through the Kafka	Select **Kafka Edition**.

	client.	
Tag	Specify the tag for the data subscription service.	Click **Add** and select **Tag Key** and **Tag Value** in the drop-down list.
Subscribed Instance Name	Specify how to name the data subscription service.	**Name after Creation**: Set the name after the data subscription service is created, which is name-*subscription ID* by default. The subscription ID is randomly assigned by the system. **Name Now**: Enter the name of the data subscription service directly in the input box below.
Quantity	Select the number of tasks to be purchased.	You can purchase up to 10 tasks at a time.

3. After successful purchase, return to the data subscription list, select the purchased subscription task, and click **Configure Subscription** in the **Operation** column.



4. On the **Select Instance** tab of the **Configure data subscription** wizard, configure the database information for the data subscription task and perform the connectivity test.



Parameter	Description	Configuration Method
Subscription ID / Name	ID and name of the subscription task. The task name is name-subscription ID by default.	Confirm the ID and name of the data subscription task.

Instance Type	It is **MongoDB** by default.	-
Region	The region where the subscription service resides.	Confirm the region.
Access Type	Select the type of source database access to the data subscription service.	Currently, only **Database** is supported, i.e., a TencentDB instance.
Instance Name	Select the specific MongoDB instance for the data subscription service.	Select a specific instance ID in the drop-down list.
Account	Set the access account information of the MongoDB instance.	Enter the prepared read-only account information in the input box.
Password	Set the password of the access account of the MongoDB instance.	Enter the password of the read-only account in the input box. Password-free access is not supported.
Kafka Partition Count	Select the number of Kafka partitions for the data subscription task. In Kafka, a consumer can get data by subscribing to one or more topics and then consuming data from one or more partitions of each topic.	You can select **1**, **4**, or **8**. A single partition can guarantee the order of messages, while multiple partitions cannot. If you have strict requirements for the order of messages during consumption, set this value to 1. Increasing the number of partitions can improve the throughput and parallelism of the Kafka cluster, because multiple consumers can consume different partitions at the same time. However, doing so will also increase the management and maintenance costs of the Kafka cluster and may cause data imbalance or delays.
Test Connectivity	Test the connectivity between the data subscription service and the TencentDB for MongoDB instance.	Click **Test Connectivity** and wait for the test result. If the test fails, troubleshoot as prompted. Then, click **Test Again** to test the connectivity again. After the test passes, proceed to the next step.

5. Click **Next**, configure the parameters as follows on the **Subscription Type and Object** page, and click **Save**.

Parameter	Description	Configuration Method

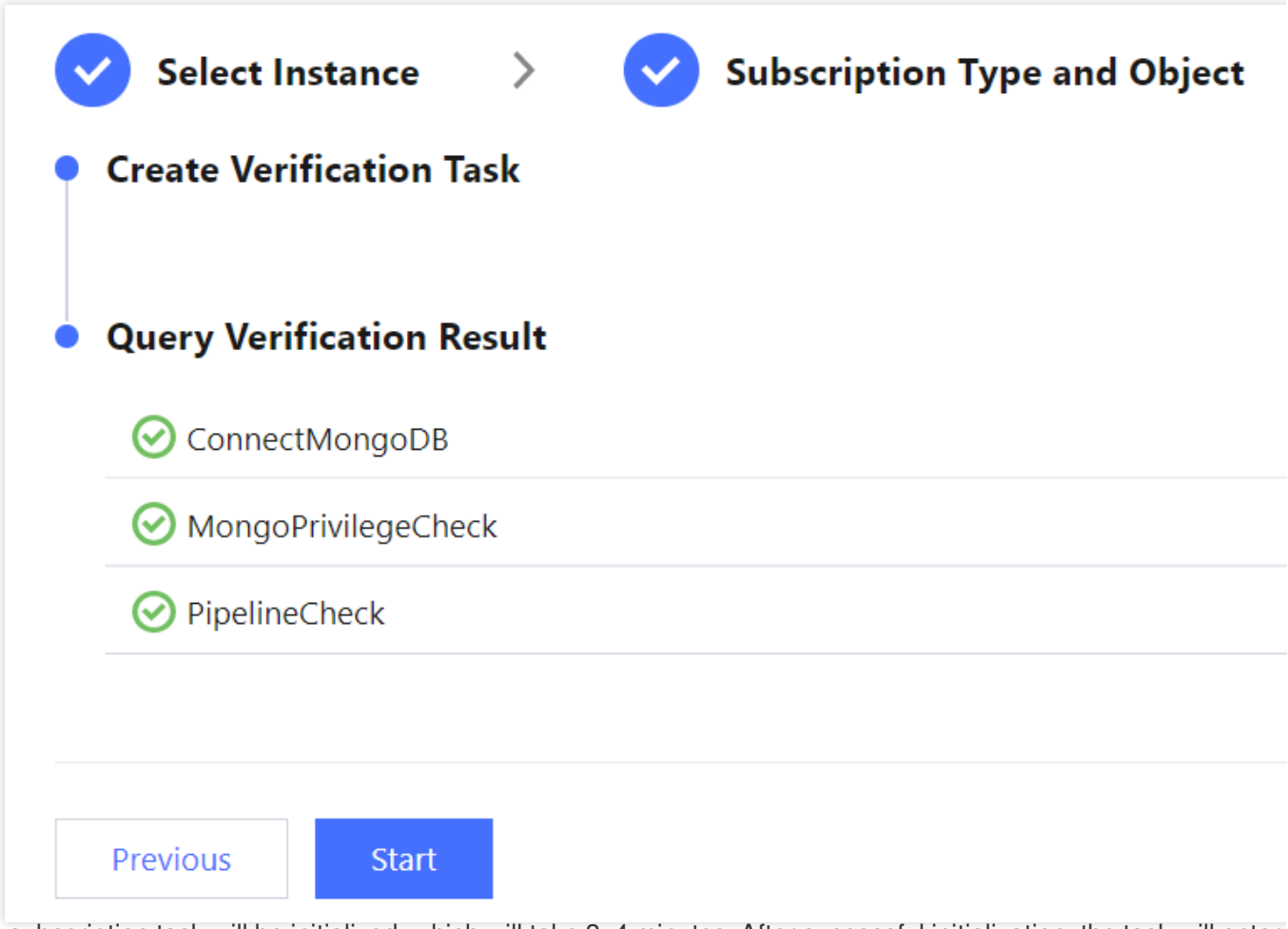
Subscription ID / Name	ID and name of the subscription task.	Check whether the subscription task information is correct.
Subscribed Instance	Instance ID of the subscribed MongoDB database.	Check whether the instance information is correct.
Data Subscription Type	The type of data that the subscriber can choose to subscribe to. MongoDB uses the change stream feature to monitor data changes and implement data subscription.	It is **Change Stream** by default and cannot be modified.
Subscription Object Level	Level of the subscribed data, including **Full instance**, **Database**, and **Collection**. **Full instance**: Subscribe to the data in the entire instance. **Database**: Subscribe to the data in the specified database. After selecting this option, you can select only one database in **Task Configuration**. **Collection**: Subscribe to the data in the specified collection. After selecting this option, you can select only one collection in **Task Configuration**.	Select the level of data subscription as needed. System databases `admin`, `local`, and `config` cannot be selected.
Task Configuration	When **Subscription Level** is **Database** or **Collection**, this parameter will be displayed for you to specify the database or collection.	In the **Select databases and tables** section, select the database or collection to be subscribed. In the **Selected** section, check whether the selected database or collection is correct.
Output Aggregation Settings	This parameter configures whether to enable aggregation for the subscribed data before sending it to the subscriber.	Enable: Click  to enable aggregation. Click **Add**, select an operator in the **Aggregation Operator** drop-down list, and enter an expression in the **Aggregation Expression** input box. Click **Add** to add multiple aggregation expressions. The aggregation pipeline will be executed based on the order of added aggregation operations. For more information, see Modify Change Stream Output.
Kafka	If **Kafka Partition Count** is not 1 in the previous	Select **Custom Partitioning Policy**,

Partitioning Policy	step, you need to set the partition policy. **By Collection Name**: Partitions the subscribed data from the source database by collection name. With this policy, data with the same collection name is written to the same Kafka partition. **Custom Partitioning Policy**: Database and collection names of the subscribed data are matched through a regex first. Then, matched data is partitioned by collection name or collection name + `objectid` .	click **Add** in **Custom Partitioning Policy** below, set the matching mode of database name or collection name in the form of regular expression in the **Database Name Match** or **Table Name Match** input box below, and select **By Collection Name** or **By Collection Name + ObjectId** in the **Partitioning Policy** drop-down list. When you enable the custom partitioning policy option, your custom partitioning policies will be applied first, followed by the Kafka partitioning policies. The data in a collection that cannot be partitioned using the custom partitioning policies will be routed to Kafka partitions by default policy **By Collection Name**.
Custom Partitioning Policy	This parameter will be displayed if **Custom Partitioning Policy** is selected in **Kafka Partitioning Policy**. It sets the custom partitioning policy.	
Policy Combo Result	This parameter will be displayed if **Custom Partitioning Policy** is selected in **Kafka Partitioning Policy**. It indicates the combo result of the custom partitioning policy.	

6. On the **Pre-verification** page, a pre-verification task will run for 2–3 minutes. After the pre-verification is passed, click **Start** to complete data subscription task configuration.

Note:

If the verification fails, fix the problem as instructed in Database Connection Check and initiate the verification again.



7. The subscription task will be initialized, which will take 3–4 minutes. After successful initialization, the task will enter the **Running** status, and data consumption will start.

Subsequent operations

1. Adding Consumer Group.

The consumption in data subscription (Kafka Edition) depends on the consumer groups of Kafka; therefore, you must create a consumer group first before data can be consumed. Data subscription (Kafka Edition) allows you to create multiple consumer groups for multi-point consumption.

2. Consuming Subscribed Data with Kafka Client (ProtoBuf).

After the subscription task enters the **Running** status, you can start consuming data. For consumption in Kafka, you need to verify the password. For code samples in different programming languages, see the demo in Consuming Subscribed Data with Kafka Client (ProtoBuf).