

# Tencent Cloud Distributed Cache (Redis OSS-Compatible) Release Notes and Announcements Product Documentation



## Copyright Notice

©2013–2026 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

## Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by the Tencent corporate group, including its parent, subsidiaries and affiliated companies, as the case may be. Trademarks of third parties referred to in this document are owned by their respective proprietors.

## Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## Release Notes and Announcements

Release Notes

Announcements

Security Vulnerabilities CVE-2024-31449, CVE-2024-31227, and CVE-2024-31228

Classic Network Will Be Disused

API for Querying Instance Big Key Will Be Disused

1-Minute Granularity Will Be Disused

Monitoring Upgrade and Alarm Policy Changes

# Release Notes and Announcements

## Release Notes

Last updated: 2026-04-03 18:10:35

### March 2026

Update	Description	Reference
Product Name Change	<p>To unify the Tencent Cloud database brand and naming standards, TencentDB for Redis® has been officially renamed to Tencent Cloud Distributed Cache Database (Redis Compatible), effective March 17, 2026.</p> <div style="border: 1px solid #00aaff; padding: 10px; margin-top: 10px;"> <p><b>Note:</b></p> <p>This renaming only involves a change to the product brand name and does not affect the functionality, performance, API interfaces, or Service Level Agreement (SLA) of existing instances. No action is required on your part, and your existing services will continue to operate normally.</p> </div>	<a href="#">Product Overview</a>
Database Audit	<p>The audit feature is a comprehensive database operation logging and analysis system. By capturing all user-executed commands in real time, securely storing them, and providing efficient query capabilities, it enhances database security and observability across the board.</p>	<a href="#">Database Audit</a>
Full-Link analysis	<p>The Full-Link analysis feature collects latency data across the entire lifecycle of an instance and delivers it to Cloud Log Service (CLS) for centralized analysis. CLS aggregates and stores the monitoring data in a structured format, converting scattered latency metrics into a unified log format for display. This enables comprehensive monitoring and in-depth analysis of the distributed cache database performance.</p>	<a href="#">Full-Link analysis</a>
Instance Purchase Now Supports	<p>Instance purchase now supports cross-availability zone (Multi-AZ) deployment by default, enabling you to build geo-level disaster recovery with a single click and</p>	<a href="#">Instance Purchasing</a>

Multi-AZ Deployment by Default	ensuring high availability and smooth business operations for your cloud database.	
--------------------------------	--	--

## January 2026

Update	Description	Reference
Valkey 8.0 console feature goes live	Valkey 8.0 supports performing configuration update, monitoring, backup and recovery, parameter configuration, slow query and other features on the console.	<a href="#">Feature list</a>
DTS migration support for Valkey 8.0	Support through Data Transfer Service (DTS) to migrate self-built or other cloud platforms' Redis data online to Tencent Cloud Valkey 8.0, achieving smooth cloud migration.	<a href="#">Migration Solution Overview</a>
Add new active-expire-effort parameter	Support adjusting the maximum CPU usage when proactively purging expired keys on the console parameter settings page. <ul style="list-style-type: none"> <li>Value range: 1–10. Default value: 1.</li> <li>Upper limit of CPU usage = 25% + (n-1) x 2%. <ul style="list-style-type: none"> <li>The default value of 1 corresponds to a CPU usage limit of 25%.</li> <li>The maximum value of 10 corresponds to a CPU usage limit of 43%.</li> </ul> </li> </ul>	<a href="#">Configurable Parameter List</a>

## December 2025

Update	Description	Reference
Valkey 8.0 first release	Officially released Valkey 8.0, compatible with the open-source Valkey 8.0.6, supporting standard architecture and shard cluster architecture. On the basis of the community edition, it comprehensively enhances performance, ease of use, high availability and stability.	<ul style="list-style-type: none"> <li><a href="#">storage engine</a></li> <li><a href="#">Version update description</a></li> </ul>

## September 2025

Update	Description	Reference
--------	-------------	-----------

Support for instance termination protection	Instance deletion protection is provided. Once enabled, it confirms twice before deleting an instance to effectively prevent accidental deletion.	<a href="#">View Redis version instance</a>
Security vulnerability fix	Vulnerabilities CVE-2025-49844, CVE-2025-46817, CVE-2025-46818, and CVE-2025-46819 have been detected on Redis. TencentDB for Redis <sup>®</sup> has fixed these vulnerabilities.	<a href="#">Redis Minor Version Updates</a>

## July 2025

Update	Description	Reference
Support enable or disable Lazy deletion related parameters	<p>Support enabling/disabling the following Lazy deletion parameters on demand to address server blocking issues caused by synchronous deletion of large keys (high memory consumption).</p> <ul style="list-style-type: none"> <li>lazyfree-lazy-user-del: When a user or app explicitly uses the DEL command to delete a key. Redis marks it as logically deleted and returns immediately. The actual memory release is performed asynchronously in the backend.</li> <li>lazyfree-lazy-user-flush: When a user or app executes FLUSHDB (clear current database) or FLUSHALL (clear all data). Redis returns immediately and delegates the deletion of all keys in the entire database to a background thread for asynchronous processing.</li> </ul>	<a href="#">Configurable parameter list</a>
Security vulnerability fix	Vulnerabilities CVE-2025-27151, CVE-2025-32023, CVE-2025-48367, CVE-2025-21605, and CVE-2024-46981 have been detected on Redis. TencentDB for Redis <sup>®</sup> has fixed these vulnerabilities.	<a href="#">Redis Minor Version Updates</a>

## April 2025

Update	Description	Reference
Enable CAM authentication for key APIs of instance operations	Enable CAM authentication for key operation APIs of instance management (including DescribeCreateGoodsInfo (new purchase goods information query), DescribeModifyGoodsInfo (modification goods information query) and	–

DescribeRenewGoodsInfo (renewal goods information query)). Simplify instance lifecycle management complexity through unified permission control.

## February 2025

Update	Description	Reference
Building Redis Client Latency Monitoring Based on Spring Boot	This practice exposes full-link latency data through Spring Boot Actuator. Using monitoring components like Prometheus and Grafana, it enables monitoring of Redis command latency from the client, helping customers achieve full-link business monitoring and issue locating.	<a href="#">Building Redis Client Monitoring Based on Spring Boot</a>

## December 2024

Update	Description	Reference
Redis supports auto-scaling	The autonomy center provides auto-scaling for Redis instances. DBbrain automatically scales out based on configured strategies like average memory utilization and scale-out specifications to cope with constantly changing business requirements.	<a href="#">Enable Automatic Scaling (DBbrain Autonomy Task)</a>

## October 2024

Update	Description	Documentation
Security vulnerability fixes	Redis officially announced severe security vulnerabilities in versions 6.2 and 7.2, including CVE-2024-31449, CVE-2024-31227, and CVE-2024-31228. TencentDB for Redis <sup>®</sup> has addressed and resolved these vulnerabilities.	<a href="#">[October 28, 2024] CVE-2024-31449, CVE-2024-31227, and CVE-2024-31228 Security Vulnerabilities</a>

## September 2024

Update	Description	Documentation
Adjustment of additional bandwidth policy	For newly added instances, additional bandwidth is now enabled by default. Users no longer need to manually apply for additional bandwidth when the standard bandwidth is insufficient.	<a href="#">Purchase Method</a>

## August 2024

Update	Description	Documentation
Support for Memcached Edition instances	Compatible with Memcached protocol version 1.6, this high-performance, memory-level distributed storage service is specifically designed for high-speed caching scenarios.	<ul style="list-style-type: none"> <li><a href="#">Storage Engine</a></li> <li><a href="#">Create Redis Instance</a></li> <li><a href="#">View Memcached Instance</a></li> <li><a href="#">Clone Memcached Instance</a></li> <li><a href="#">Inquire Memcached Slow Log</a></li> <li><a href="#">SNAT Monitoring Memcached Instance</a></li> <li><a href="#">Memcached Command Compatibility</a></li> </ul>

## May 2024

Update	Description	Documentation
Redis 7.0	TencentDB for Redis 7.0 has been officially launched, fully compatible with community edition commands.	<ul style="list-style-type: none"> <li><a href="#">Create Redis Instance</a></li> <li><a href="#">Overview of Community Edition Command Support</a></li> </ul>

## April 2024

Update	Description	Documentation
Rehash maintenance time window	Added support for the <code>rehash-maintenance-time</code> parameter, allowing users to specify whether rehash operations can be performed during the maintenance time window.	<a href="#">Manage Instance Parameters</a>

## March 2024

Update	Description	Document
Added global replication monitoring metrics.	Supports graphical monitoring of the command count deviation during the global replication process between the primary instance and disaster recovery instances, as well as the synchronization latency in remote synchronization.	<a href="#">Monitoring Metrics</a>

## February 2024

Update	Description	Document
One-click connectivity checker	If the connection to TencentDB for Redis instance fails, it is recommended to first use the one-click connectivity checker to identify the cause.	<a href="#">One-Click Connectivity Checker</a>

## January 2024

Update	Description	Document
Added monitoring metrics.	Adds a monitoring metric for the number of keys with expiration time set.	<a href="#">Monitoring feature (5-second granularity)</a>

## February 2023

Update	Description	Document
Supported the configuration of backup file download rules	To prevent database backup data from being dragged, TencentDB for Redis allows you to download backup files from specific servers on the allowed private network.	
Supported the selection of billing	When cloning a complete new instance based on the current backup file, you can choose the pay-as-you-go billing mode.	

methods for  
clone  
instances

## November 2022

Update	Description	Document
Released Redis 6.2	Redis 6.2 is fully released, which supports standard and cluster architectures, read/write separation, scaling without disconnection, and ultra high availability.	<a href="#">Product Series</a>
Supported exporting parameters as a file	On the <b>Parameter Settings</b> page in the console, you can export the parameter file of the current instance to easily view them locally.	<a href="#">Setting Instance Parameters</a>

## October 2022

Update	Description	Document
Optimized the instance list in the console	<ul style="list-style-type: none"> <li>The <b>Deployment Mode</b> column is added in the instance list to indicate whether an instance is deployed across AZs.</li> <li>When exporting the instance list information, you can aggregate tags for display, which are separated by  .</li> </ul>	<a href="#">Viewing Instance Information</a>

## September 2022

Update	Description	Document
Added monitoring metrics	The <b>Max Connections Utilization of Node</b> metric is added to monitor the maximum connection utilization of all proxy nodes in an instance.	<a href="#">Monitoring at Five-Second Granularity</a>

## August 2022

Update	Description	Document
Supported querying slow logs of read-only replicas	TencentDB for Redis supports slow query of read-only replica to help you troubleshoot performance problems of read-only operations.	–

Supported smooth scale-in to the 256 MB specification	TencentDB for Redis in standard architecture supports smooth reduction of memory capacity from 1 GB or higher to 256 MB.	<a href="#">Changing Instance Specification</a>
---	--	---

## July 2022

Update	Description	Document
Optimized statistical algorithm for traffic and throttling that triggers monitoring metrics	TencentDB for Redis now provides optimized statistical algorithm for inbound/outbound traffic and throttling.	–
Supported the 256 MB specification for the standard architecture	The minimum memory specification (256 MB) is available for TencentDB for Redis 4.0 and 5.0 Memory Edition (Standard Architecture) instances in more AZs.	<a href="#">Performance</a>

## June 2022

Update	Description	Document
Supported SSL encryption	TencentDB for Redis supports SSL encryption to implement encrypted data transfer.	<a href="#">SSL Encryption</a>

## May 2022

Update	Description	Document
Supported minor version upgrade	TencentDB for Redis supports minor version upgrade, so that you can upgrade your instance to a newer version for more features.	<a href="#">Upgrading Instance Version</a>

Supported proxy upgrade	Minor versions of the TencentDB for Redis proxy are released from time to time to add more database features or fix known bugs.	<a href="#">Upgrading Proxy</a>
-------------------------	---	---------------------------------

## April 2022

Update	Description	Document
Upgraded the global replication feature	Multi-AZ instances can be added to a global replication group, and existing instances can be quickly upgraded to global replication instances.	<a href="#">Creating Global Replication Group</a>

## February 2022

Update	Description	Document
Supported architecture upgrade for multi-AZ deployed instances	Multi-AZ deployed instances can be upgraded from the standard architecture to cluster architecture.	<a href="#">Upgrading Instance Architecture</a>
Supported instance clone for multi-AZ deployed instances	Multi-AZ deployed instances allow you to clone a complete instance from a backup file.	<a href="#">Cloning Data</a>

## January 2022

Update	Description	Document
Supported global replication	The global replication feature allows you to add and remove instances, switch instance roles, and delete replication groups.	<a href="#">Overview</a>
Supported custom backup retention period	You can apply for customizing the retention period of backup files through an allowlist.	<a href="#">Backing up Data</a>

Added the API for failure simulation	Instances in the same AZ support calling the <code>KillMasterGroup</code> API for failure simulation testing.	<a href="#">KillMasterGroup</a>
Added the API for master-replica switch	Instances in the same AZ support calling the <code>ChangeReplicaToMaster</code> API to promote a replica node (group) to a master node for master-replica switch.	<a href="#">ChangeReplicaToMaster</a>

## December 2021

Update	Description	Document
Supported the 256 MB memory specification for TencentDB for Redis 4.0 and 5.0 Memory Edition (Standard Architecture)	The minimum memory specification of TencentDB for Redis 4.0 and 5.0 Memory Edition (Standard Architecture) instances can be 256 MB. Currently, this is supported only in Shanghai Zone 5, Beijing Zone 6, and Guangzhou Zone 6.	<a href="#">Performance</a>

## November 2021

Update	Description	Document
Supported system default templates	TencentDB for Redis now provides default parameter templates applicable to Redis 2.8 standard architecture, Redis 4.0 standard architecture, Redis 4.0 cluster architecture, Redis 5.0 standard architecture, and Redis 5.0 cluster architecture. You can view them on the parameter template page in the console.	<a href="#">Applying Parameter Templates</a>
Supported enabling public	TencentDB for Redis allows you to enable public network access in the <a href="#">console</a> , so that you can connect to your TencentDB for Redis instance from a CVM instance in different VPCs over the public network.	<a href="#">Configuring Public Network Address</a>

network  
access

## July 2021

Update	Description	Document
Supported parameter template	Besides the system parameter templates provided by TencentDB for Redis, you can create custom parameter templates to configure parameters in batches as needed.	<a href="#">Applying Parameter Templates</a>

## June 2021

Update	Description	Document
Supported automatic failover	TencentDB for Redis supports the automatic failover feature for proxy nodes and Redis server nodes to ensure service availability.	<a href="#">Failover</a>
Auto-Failback	TencentDB for Redis provides the automatic failback feature for instances deployed across AZs. After the feature is enabled, if the master node is switched from the master AZ or master node group (cluster architecture) to another AZ or group after a failover occurs, it will be automatically switched back, simplifying subsequent Ops operations.	<a href="#">Automatic Failback</a>
Manually Promoting to Master Node (Group)	For multi-AZ deployed TencentDB for Redis instances in the standard/cluster architecture, you can manually promote a replica node/node group to master node/node group. You can deploy the master node in a specified AZ/node group.	<a href="#">Manually Promoting to Master Node (Group)</a>
Reading Local Nodes Only	To reduce the access latency of a multi-AZ deployed instance, TencentDB for Redis allows you to read local nodes only.	<a href="#">Reading Local Nodes Only</a>

## March 2021

Update	Description	Document
Fully supported monitoring metrics at a 5-	TencentDB for Redis now supports the 5-second monitoring granularity. After the monitoring granularity of an instance is adjusted to five seconds, the monitoring metrics, proxy, and alarm policies of the instance, and the method of viewing monitoring data in the TCOP console will change.	<a href="#">Monitoring Upgrade and Alarm Policy Changes</a>

second  
granularity

## December 2020

Update	Description	Document
Renamed TencentDB for Redis Hybrid Storage Edition TencentDB for Tendis	TencentDB for Redis Hybrid Storage Edition has been renamed TencentDB for Tendis. You can access TencentDB for Tendis in its own console.	<a href="#">TencentDB for Tendis</a>
Supported multi-AZ deployment	You can now deploy TencentDB for Redis master and replica nodes in different availability zones (AZs) of the same region. Multi-AZ deployed instances have higher availability and better disaster recovery capability than single-AZ deployed instances.	<a href="#">Multi-AZ Deployment</a>

## September 2020

Update	Description	Document
Supported the 5-second granularity for monitoring data	The monitoring feature of TencentDB for Redis has been updated, with the monitoring granularity being narrowed down from one minute to five seconds.	<a href="#">Monitoring at Five-Second Granularity</a>

## July 2020

Update	Description	Document
Supported version upgrade	TencentDB for Redis now supports version upgrade, so you can upgrade from a lower version of Standard Edition to a higher one, including from 2.8 to 4.0, from 2.8 to 5.0, and from 4.0 to 5.0.	<a href="#">Upgrading Instance Version</a>
Supported architecture upgrade	TencentDB for Redis now supports quick upgrade from standard architecture to cluster architecture to help your business expand the performance and capacity with speed.	<a href="#">Upgrading Instance Architecture</a>

## June 2020

Update	Description	Document
Launched TencentDB for Redis Hybrid Storage Edition	TencentDB for Redis Hybrid Storage Edition is launched. With 100% compatibility with Redis protocols, it reduces memory costs by up to 80% and delivers a hot data performance comparable with that of Memory Edition, achieving a perfect balance among compatibility, performance, and costs.	<a href="#">Hybrid Storage Edition (Cluster Architecture)</a>

## April 2020

Update	Description	Document
Supported Redis 5.0 in DTS	You can now migrate data and upgrade instances to Redis 5.0 with DTS.	<a href="#">Migration with DTS</a> <a href="#">Version Upgrade with DTS</a>

## March 2020

Update	Description	Document
Supported the service unavailable alarm feature	TencentDB for Redis now supports the service unavailable alarm feature. In addition, users can also set alarms for instance master/replica switch, read-only replica failover, and read-only replica unavailability.	<a href="#">Configuring Alarm</a>
Supported adjusting bandwidth in the console	You can now adjust instance network bandwidth on the instance details page in the Redis console. The bandwidth specifications for Cluster Edition have been fully upgraded, with shard specifications upgraded to 384 Mbps.	–
Updated the monitoring view for Redis 2.8	To provide more accurate monitoring information, Tencent Cloud has updated the monitoring view of Redis 2.8 instance to cluster view. If you use APIs to get monitoring data from TCOP, you need to change the view parameter from <code>redisuuid</code> to <code>cluster</code> in the code.	–

## February 2020

Update	Description	Document
Upgraded the default number of databases in an instance to 256	For Redis 2.8, 4.0, and 5.0 Standard and Cluster Editions you purchased this month, the default number of databases in an instance has been upgraded from 16 to 256.	<a href="#">Memory Edition (Standard Architecture)</a> <a href="#">Memory Edition (Cluster Architecture)</a>

## January 2020

Update	Description	Document
Released Redis 5.0	Redis 5.0 Standard and Cluster Editions have been released with all features of Redis 4.0 reserved, and support the latest STREAM data structure, new ZSET commands <code>ZPOPMIN</code> and <code>ZPOPMAX</code> , scaling without disconnection, 4 TB of storage capacity, and tens of millions of concurrent QPS.	<a href="#">Memory Edition (Standard Architecture)</a> <a href="#">Memory Edition (Cluster Architecture)</a>

## October 2019

Update	Description	Document
Supported access management	You can now create policies through access management which grant sub-accounts permissions to use the resources they need.	<a href="#">Access Management</a>
Supported account management in the console	TencentDB for Redis provides read-write permission control and routing policy control through the account mechanism, which helps meet the needs of business permission management in complex scenarios. Currently, only the TencentDB for Redis Community Edition (excluding Redis 2.8) supports account settings.	<a href="#">Managing Account</a>

## September 2019

Update	Description	Document
Released monitoring 2.0	Redis monitoring 2.0 has been released, adding more than 16 monitoring metrics, including network delay, response error and other monitoring metrics.	<a href="#">Monitoring</a>

## August 2019

Update	Description	Document
Supported password-free access	To enable password-free access, <a href="#">submit a ticket</a> . We recommend that you limit server access by using security groups after this feature is enabled.	<a href="#">Password-Free Access</a>
Supported disabling high-risk commands online	TencentDB for Redis is now able to disable some Redis commands that may cause service instability or accidentally delete data, including <code>flushall</code> , <code>flushdb</code> , <code>keys</code> , <code>hgetall</code> , <code>eval</code> , <code>evalsha</code> , and <code>script</code> .	<a href="#">Disabling Commands</a>
Supported Cluster Edition in DTS	You can now migrate self-created Redis Cluster (3.0, 3.2, and 4.0) or Codis (2.8 and 3.2) databases to TencentDB for Redis with DTS.	<a href="#">Migration with DTS</a>

## July 2019

Update	Description	Document
Supported version upgrade with DTS	TencentDB for Redis instance version can now be upgraded through DTS with hot migration, which guarantees instance service continuity during the upgrade process and can update incremental data in real time.	<a href="#">Version Upgrade with DTS</a>
Released Redis 4.0 Standard Edition	Redis 4.0 Standard Edition supports 1-master 5-replica mode, read/write separation, scaling without disconnection, and ultra high availability.	<a href="#">Memory Edition (Standard Architecture)</a>

## October 2018

Update	Description	Document
Released Redis 4.0 Cluster Edition	Redis 4.0 Cluster Edition has been released and supports 4 TB of storage capacity, tens of millions of concurrent access requests, scaling without storage capacity loss, and automatic read/write separation.	<a href="#">Memory Edition (Cluster Architecture)</a>

# Announcements

## Security Vulnerabilities CVE-2024-31449, CVE-2024-31227, and CVE-2024-31228

Last updated: 2025-07-11 14:34:28

Recently, vulnerabilities CVE-2024-31449, CVE-2024-31228, and CVE-2024-31227 have been detected on Redis. TencentDB for Redis<sup>®</sup> has fixed these vulnerabilities. It is recommended that you upgrade to the latest instance minor version promptly to ensure business security and stable operation.

### Vulnerability Impact

- [CVE-2024-31449](#): Users who pass authentication by executing the AUTH command can use a specially crafted Lua attack script to trigger a stack buffer overflow in the bit library. This may cause a process crash and remote code execution.
- [CVE-2024-31227](#): Users who pass authentication by executing the AUTH command can create an illegal ACL selector and use commands such as ACL LIST or ACL GETUSER to trigger an assertion failure. This will cause a process crash and denial-of-service (DoS) issues.
- [CVE-2024-31228](#): Users who pass authentication by executing the AUTH command can create a special long string key in the database. They can use commands supporting string match, such as KEYS or SCANS, and a special long string match mode to match extremely long strings. This may lead to infinite recursion that suspends the process, stack overflow, process crash, and DoS issues.

### Version Vulnerability Fixing Description

You can upgrade Redis instances to the minor versions shown in the table below to fix the security vulnerabilities. For detailed operations, see [Upgrading Instance Version](#).

Compatible Version	Minor Version	Update
Redis 7.0	7.0.20	Fixed the security vulnerabilities CVE-2024-31449, CVE-2024-31227, and CVE-2024-31228.
Redis 6.2	6.2.9	Fixed the security vulnerabilities CVE-2024-31449 and CVE-2024-31228.
Redis 5.0	5.2.11	Fixed the security vulnerabilities CVE-2024-31449 and CVE-2024-31228.

Redis 4.0	4.3.12	Fixed the security vulnerabilities CVE-2024-31449 and CVE-2024-31228.
-----------	--------	---

## Reference

[Redis official announcement: Security Advisory: CVE-2024-31449, CVE-2024-31227, CVE-2024-31228.](#)

# Classic Network Will Be Disused

Last updated: 2022-11-08 17:18:57

Dear user,

Tencent Cloud stopped the support of the creation of any new resources in the classic network on **March 31, 2022** and will officially discontinue the classic network product from **December 31, 2022**. After that, all classic network-based resources will be migrated to VPC.

As the current mainstream cloud network environment, VPC has all the features of the classic network while providing a more secure, flexible, and stable network environment and user experience. If your business is still using the classic network, we recommend you migrate it to VPC for a better service experience.

## Note:

For more information, see [Migration Solutions](#).

## Contact Us

For any questions, [contact us](#).

# API for Querying Instance Big Key Will Be Disused

Last updated: 2022-11-17 11:00:02

Dear user,

TencentDB for Redis will discontinue API [DescribeInstanceMonitorBigKey](#) for querying the big key of an instance on October 31, 2022, but this feature will continue to serve through the API [DescribeRedisTopBigKeys](#) of DBbrain.

TencentDB for DBbrain (DBbrain) is an intelligent database diagnosis and optimization product that provides real-time performance diagnosis and security protection services. It troubleshoots efficiently, offers solutions to database exceptions, and helps you prevent exceptions at the source. It can also help improve the overall database performance with its AI-powered parameter tuning capabilities.

[DescribeRedisTopBigKeys](#) of DBbrain can specify the sorting field and type of the [Key](#) to quickly query of big key information of an instance, including expiration time, memory size, element quantity, and the max element length.

## List of deprecated APIs

[DescribeInstanceMonitorBigKey](#) API and its associated APIs. [DescribeInstanceMonitorBigKeySizeDist](#) and [DescribeInstanceMonitorBigKeyTypeDist](#) will be deprecated simultaneously.

API Name	Description
<a href="#">DescribeInstanceMonitorBigKey</a>	Queries the big key of an instance
<a href="#">DescribeInstanceMonitorBigKeySizeDist</a>	Queries the big key size distribution of an instance
<a href="#">DescribeInstanceMonitorBigKeyTypeDist</a>	Queries the big key type distribution of an instance

# 1–Minute Granularity Will Be Disused

Last updated: 2023-03-29 15:36:39

CM will discontinue the 1–minute granularity alarm policy for TencentDB for Redis and QCE/REDIS namespace, which will have the following impacts:

- You can't receive the alarm notification of TencentDB for Redis Memory Edition (1–minute granularity).
- Data can't be displayed on the dashboard of TencentDB for Redis Memory Edition (1–minute granularity).
- You can't pull the monitoring data of TencentDB for Redis Memory Edition (1–minute granularity).

## Migration Schemes

- Alarm policy: You need to migrate the alarm policy of the existing TencentDB for Redis instances with 1–minute monitoring granularity to those with 5–second monitoring granularity. [Click to view the mappings of metrics.](#)
- Dashboard: You need to migrate the dashboard of the existing TencentDB for Redis instances with 1–minute monitoring granularity to 5–second monitoring granularity. [Click to view the mappings of metrics.](#)
- Pulling monitoring metric data through APIs: You need to switch QCE/REDIS to QCE/REDIS\_MEM, and modify the names of pulled metrics simultaneously.

Because 1–minute monitoring granularity of TencentDB for Redis is not completely the same as the metric names in [Mappings of Metrics](#). This document describes the mappings of monitoring metrics of TencentDB for Redis 1–minute monitoring granularity and 5–second monitoring granularity for you to compare and migrate.

## Monitoring Metric Mappings Between 1–Minute and 5–Second Alarm Policy

1–Minute Metric	1–Minute Metric Parameter	5–Second Metric	5–Second Metric Parameter	Unit	Description
CPU Load	cpu_us_min	CPU Utilization	cpu_util	%	Average CPU utilization
Max CPU Load	CpuMaxUsMin	Max CPU Utilization of Node	cpu_max_util	%	The maximum CPU utilization of a node (shard or replica) in an instance
Used Memory	storage_min	Used Memory	mem_used	MB	Actually used memory capacity, including the capacity for data and cache

Capacity Utilization	storage_us_min	Memory Utilization	mem_util	%	The ratio of the actually used memory to the requested total memory
Max Capacity Utilization	StorageMaxUsMin	Max Memory Utilization of Node	mem_max_util	%	The maximum among all node (shard or replica) memory utilizations in an instance
Private Network Inbound Traffic	in_flow_min	Inbound Traffic	in_flow	MB/s	Private network inbound traffic
Inbound Traffic Utilization	in_flow_us_min	Inbound Traffic Utilization	in_bandwidth_util	%	The ratio of the actually used private inbound traffic to the maximum traffic
Private Network Outbound Traffic	out_flow_min	Outbound Traffic	out_flow	MB/s	Private network outbound traffic
Outbound Traffic Utilization	out_flow_us_min	Outbound Traffic Utilization	out_bandwidth_util	%	The ratio of the actually used private outbound traffic to the maximum traffic
Connections	connections_min	Connections	connections	-	The number of TCP connections to an instance
Connection Utilization	connections_us_min	Connection Utilization	connections_util	%	The ratio of the number of TCP connections to the maximum number of connections
Slow Logs	slow_query_min	Slow Logs	cmd_slow	-	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Total Keys	keys_min	Total Keys	keys	-	The total number of keys (level-1 keys) in instance

					storage
Expired Keys	expired_keys_min	Expired Keys	expired	-	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
Evicted Keys	evicted_keys_min	Evicted Keys	evicted	-	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code> output by the <code>info</code> command.
Average Execution Latency	latency_min	Average Execution Latency	latency_avg	ms	The average execution latency between the proxy and the Redis server
Average Read Latency	latency_get_min	Average Read Latency	latency_read	ms	The average execution latency of read commands between the proxy and the Redis server
Average Write Latency	latency_set_min	Average Write Latency	latency_write	ms	The average execution latency of write command between the proxy and the Redis server
Average Latency of Other Commands	latency_other_min	Average Latency of Other Commands	latency_other	ms	The average execution latency of commands (excluding write and read commands) between the proxy and the Redis server
QPS	qps_min	Total Requests	commands	Counts/seconds	Queries per second
Read Requests	stat_get_min	Read Requests	cmd_read	Counts/seconds	The number of read command executions per second

				s e c	
Write Requests	stat_set_min	Write Requests	cmd_write	C o u n t s/ s e c	The number of write command executions per second
Other Requests	stat_other_min	Other Requests	cmd_other	C o u n t s/ s e c	The number of command (excluding write and read commands) executions per second
Big Value Requests	big_value_min	Big Value Requests	cmd_big_value	C o u n t s/ s e c	The number of executions of requests larger than 32 KB per second
Read Request Hits	stat_success_min	Read Request Hits	cmd_hits	-	The number of keys successfully requested by read commands, which is equal to the value of the <code>key space_hits</code> metric output by the <code>info</code> command.
Read Request Misses	stat_missed_min	Read Request Misses	cmd_miss	-	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>key space_misses</code> metric output by the <code>info</code> command.

Execution Errors	cmd_err_min	Execution Errors	cmd_err	-	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
Read Request Hit Rate	cache_hit_ratio_min	Read Request Hit Rate	cmd_hits_ratio	%	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss. When the access request quantity is 0, the value of this metric will be null.

## Mappings of Monitoring Metrics Between 1–Minute and 5–Second Dashboard

1–Minute Metric	1–Minute Metric Parameter	5–Second Metric	5–Second Metric Parameter	Unit	Description
CPU Utilization	cpu_us_min	Average CPU utilization	cpu_util	%	Average CPU utilization
Max CPU Utilization of Shard	cpu_max_us_min	Max CPU Utilization of Node	cpu_max_util	%	The maximum CPU utilization of a node (shard or replica) in an instance
Used Memory	storage_min	Used Memory	mem_used	MB	Actually used memory capacity, including the capacity for data and cache
Memory Utilization	storage_us_min	Memory Utilization	mem_util	%	The ratio of the actually used memory to the requested total memory
Max Memory Utilization of Shard	storage_max_us_min	Max Memory Utilization of Node	mem_max_util	%	The maximum among all node (shard or replica) memory utilizations in an instance

Inbound Traffic	in_flow_min	Inbound Traffic	in_flow	M B/s	Private network inbound traffic
Inbound Traffic Utilization	in_flow_us_min	Inbound Traffic Utilization	in_bandwidth_util	%	The ratio of the actually used private inbound traffic to the maximum traffic
Outbound Traffic	out_flow_min	Outbound Traffic	out_flow	M B/s	Private network outbound traffic
Outbound Traffic Utilization	out_flow_us_min	Outbound Traffic Utilization	out_bandwidth_util	%	The ratio of the actually used private outbound traffic to the maximum traffic
Connections	connections_min	Connections	connections	-	The number of TCP connections to an instance
Connection Utilization	connections_us_min	Connection Utilization	connections_util	%	The ratio of the number of TCP connections to the maximum number of connections
Slow Logs	slow_query_min	Slow Logs	cmd_slow	-	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Total Keys	keys_min	Total Keys	keys	-	The total number of keys (level-1 keys) in instance storage
Expired Keys	expired_keys_min	Expired Keys	expired	-	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
Evicted Keys	evicted_keys_min	Evicted Keys	evicted	-	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code>

					ys output by the info command.
Average Execution Latency	latency_min	Average Execution Latency	latency_avg	ms	The average execution latency between the proxy and the Redis server
Average Read Latency	latency_get_min	Average Read Latency	latency_read	ms	The average execution latency of read commands between the proxy and the Redis server
Average Write Latency	latency_set_min	Average Write Latency	latency_write	ms	The average execution latency of write command between the proxy and the Redis server
Average Latency of Other Commands	latency_other_min	Average Latency of Other Commands	latency_other	ms	The average execution latency of commands (excluding write and read commands) between the proxy and the Redis server
QPS	qps_min	Total Requests	commands	Counts/sec	Queries per second
Read Requests	stat_get_min	Read Requests	cmd_read	Counts/sec	The number of read command executions per second
Write Requests	stat_set_min	Write Requests	cmd_write	Counts/sec	The number of write command executions per second
get Requests	cmdstat_get_min	Other Requests	cmd_other	Counts/sec	The number of command (excluding write and read commands) executions per second
getbit Requests	cmdstat_getbit_min				

getrange Requests	cmdstat_getrange_min				
hget Requests	cmdstat_hget_min				
hgetall Requests	cmdstat_hmget_min				
hmget Requests	cmdstat_hmget_min				
hmset Requests	cmdstat_hmset_min				
hset Requests	cmdstat_hset_min				
hsetnx Requests	cmdstat_hsetnx_min				
lset Requests	cmdstat_lset_min				
mget Requests	cmdstat_mget_min				
mset Requests	cmdstat_mset_min				
msetnx Requests	cmdstat_msetnx_min				
set Requests	cmdstat_set_min				
setbit Requests	cmdstat_setbit_min				
setex Requests	cmdstat_setex_min				
setnx Requests	cmdstat_setnx_min				
setrange Requests	cmdstat_setrange_min				

Big Value Requests	big_value_min	Big Value Requests	cmd_big_value	Counts/sec	The number of executions of requests larger than 32 KB per second
Read Request Hits	stat_success_min	Read Request Hits	cmd_hits	-	The number of keys successfully requested by read commands, which is equal to the value of the <code>keyspace_hits</code> metric output by the <code>info</code> command.
Read Request Misses	stat_missed_min	Read Request Misses	cmd_miss	-	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>keyspace_misses</code> metric output by the <code>info</code> command.
Execution Errors	cmd_err_min	Execution Errors	cmd_err	-	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
Read Request Hit Rate	cache_hit_ratio_min	Read Request Hit Rate	cmd_hits_ratio	%	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss. When the access request quantity is 0, the value of this metric will be null.

## Mappings between 1-minute and 5-second API monitoring metrics

### Namespace

- 1-minute: Namespace=QCE/REDIS
- 5-second: Namespace=QCE/REDIS\_MEM

### Monitoring metrics

#### Instance dimension (standard architecture)

1-Minute Metric	1-Minute Metric Parameter	5-Second Metric	5-Second Metric Parameter	Unit	Description
CPU Utilization	CpuUsMin	CPU Utilization	CpuUtil	%	Average CPU utilization
Used Memory	StorageMin	Used Memory	MemUsed	MB	Actually used memory capacity, including the capacity for data and cache
Memory Utilization	StorageUsMin	Memory Utilization	MemUtil	%	The ratio of the actually used memory to the requested total memory
Total Keys	KeysMin	Total Keys	Keys	-	The total number of keys (level-1 keys) in instance storage
Expired Keys	ExpiredKeysMin	Expired Keys	Expired	-	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
Evicted Keys	EvictedKeysMin	Evicted Keys	Evicted	-	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code> output by the <code>info</code> command.
Connections	ConnectionsMin	Connections	Connections	-	The number of TCP connections to an instance
Connection Utilization	ConnectionsUsMin	Connection Utilization	ConnectionsUtil	%	The ratio of the number of TCP connections to the maximum number of connections
Inbound Traffic	InFlowMin	Inbound Traffic	InFlow	MB/s	Private network inbound traffic

Inbound Traffic Utilization	InFlowUsMin	Inbound Traffic Utilization	InBandwidth Util	%	The ratio of the actually used private inbound traffic to the maximum traffic
Outbound Traffic	OutFlowMin	Outbound Traffic	OutFlow	M B/s	Private network outbound traffic
Outbound Traffic Utilization	OutFlowUsMin	Outbound Traffic Utilization	OutBandwidthUtil	%	The ratio of the actually used private outbound traffic to the maximum traffic
Average Execution Latency	LatencyMin	Average Execution Latency	LatencyAvg	ms	The average execution latency between the proxy and the Redis server
Average Read Latency	LatencyGetMin	Average Read Latency	LatencyRead	ms	The average execution latency of read commands between the proxy and the Redis server
Average Write Latency	LatencySetMin	Average Write Latency	LatencyWrite	ms	The average execution latency of write command between the proxy and the Redis server
Average Latency of Other Commands	LatencyOtherMin	Average Latency of Other Commands	LatencyOther	ms	The average execution latency of commands (excluding write and read commands) between the proxy and the Redis server
Total Requests	QpsMin	Total Requests	Commands	Counts/sec	Queries per second
Read Requests	StatGetMin	Read Requests	CmdRead	Counts/s	The number of read command executions per second

				sec	
Write Requests	StatSetMin	Write Requests	CmdWrite	Count/s/second	The number of write command executions per second
Other Requests	StatOtherMin	Other Requests	CmdOther	Count/s/second	The number of command (excluding write and read commands) executions per second
Big Value Requests	BigValueMin	Big Value Requests	CmdBigValue	Count/s/second	The number of executions of requests larger than 32 KB per second
Slow Logs	SlowQueryMin	Slow Logs	CmdSlow	–	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Read Request Hits	StatSuccessMin	Read Request Hits	CmdHits	–	The number of keys successfully requested by read commands, which is equal to the value of the <code>keyspace_hits</code> metric output by the <code>info</code> command.
Read Request Misses	StatMissedMin	Read Request Misses	CmdMiss	–	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>keyspace_misses</code> metric output by the <code>info</code> command.

Execution Errors	CmdErrMin	Execution Errors	CmdErr	-	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
Read Request Hit Rate	CacheHitRatioMin	Read Request Hit Rate	CmdHitsRatio	%	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss. When the access request quantity is 0, the value of this metric will be null.

### Instance dimension (cluster architecture)

1-Minute Metric	1-Minute Metric Parameter	5-Second Metric	5-Second Metric Parameter	Unit	Description
Average CPU utilization	CpuUsMin	CPU Utilization	CpuUtil	%	Average CPU utilization
Max CPU Utilization of Shard	CpuMaxUsMin	Max CPU Utilization of Node	CpuMaxUtil	%	The highest CPU utilization value of all shards in a cluster
Used Memory	StorageMin	Used Memory	MemUsed	MB	Actually used memory capacity, including the capacity for data and cache
Memory Utilization	StorageUsMin	Memory Utilization	MemUtil	%	The ratio of the actually used memory to the requested total memory
Max Memory Utilization of Shard	StorageMaxUsMin	Max Memory Utilization of Node	MemMaxUtil	%	The highest memory utilization value of all shards in a cluster
Total Keys	KeysMin	Total Keys	Keys	-	The total number of keys (level-1 keys) in instance

					storage
Expired Keys	ExpiredKeysMin	Expired Keys	Expired	-	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
Evicted Keys	EvictedKeysMin	Evicted Keys	Evicted	-	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code> output by the <code>info</code> command.
Connections	ConnectionsMin	Connections	Connections	-	The number of TCP connections to an instance
Connection Utilization	ConnectionsUsMin	Connection Utilization	Connections Util	%	The ratio of the number of TCP connections to the maximum number of connections
Inbound Traffic	InFlowMin	Inbound Traffic	InFlow	M B/s	Private network inbound traffic
Inbound Traffic Utilization	InFlowUsMin	Inbound Traffic Utilization	InBandwidth Util	%	The ratio of the actually used private inbound traffic to the maximum traffic
Outbound Traffic	OutFlowMin	Outbound Traffic	OutFlow	M B/s	Private network outbound traffic
Outbound Traffic Utilization	OutFlowUsMin	Outbound Traffic Utilization	OutBandwidth Util	%	The ratio of the actually used private outbound traffic to the maximum traffic
Average Execution Latency	LatencyMin	Average Execution Latency	LatencyAvg	ms	The average execution latency between the proxy and the Redis server

Average Read Latency	LatencyGetMin	Average Read Latency	LatencyRead	ms	The average execution latency of read commands between the proxy and the Redis server
Average Write Latency	LatencySetMin	Average Write Latency	LatencyWrite	ms	The average execution latency of write command between the proxy and the Redis server
Average Latency of Other Commands	LatencyOtherMin	Average Latency of Other Commands	LatencyOther	ms	The average execution latency of commands (excluding write and read commands) between the proxy and the Redis server
Total Requests	QpsMin	Total Requests	Commands	Counts/sec	Queries per second
Read Requests	StatGetMin	Read Requests	CmdRead	Counts/sec	The number of read command executions per second
Write Requests	StatSetMin	Write Requests	CmdWrite	Counts/sec	The number of write command executions per second
Other Requests	StatOtherMin	Other Requests	CmdOther	Counts/sec	The number of command (excluding write and read commands) executions per second
Big Value Requests	BigValueMin	Big Value Requests	CmdBigValue	Counts/sec	The number of executions of requests larger than 32 KB per second

Slow Logs	SlowQueryMin	Slow Logs	CmdSlow	-	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Read Request Hits	StatSuccessMin	Read Request Hits	CmdHits	-	The number of keys successfully requested by read commands, which is equal to the value of the <code>keyspace_hits</code> metric output by the <code>info</code> command.
Read Request Misses	StatMissedMin	Read Request Misses	CmdMiss	-	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>keyspace_misses</code> metric output by the <code>info</code> command.
Execution Errors	CmdErrMin	Execution Errors	CmdErr	-	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
Read Request Hit Rate	CacheHitRatioMin	Read Request Hit Rate	CmdHitsRatio	%	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss. When the access request quantity is 0, the value of this metric will be null.

## Cluster sharding

1-Minute Metric	1-Minute Metric Parameter	5-Second Metric	5-Second Metric Parameter	Unit	Description
CPU Utilization	CpuUsNodeMin	CPU Utilization	CpuUtilNode	%	Average CPU utilization

Used Memory	StorageNodeMin	Used Memory	MemUsedNode	MB	Actually used memory capacity, including the capacity for data and cache
Memory Utilization	StorageUsNodeMin	Memory Utilization	MemUtilNode	%	The ratio of the actually used memory to the requested total memory
Total Keys	KeysNodeMin	Total Keys	KeysNode	-	The total number of keys (level-1 keys) in instance storage
Expired Keys	ExpiredKeysNodeMin	Expired Keys	ExpiredNode	-	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
Evicted Keys	EvictedKeysNodeMin	Evicted Keys	EvictedNode	-	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code> output by the <code>info</code> command.
Total Requests	QpsNodeMin	Total Requests	CommandsNode	Counts/sec	Queries per second
Read Requests	StatGetNodeMin	Read Requests	CmdReadNode	Counts/sec	The number of read command executions per second
Write Requests	StatSetNodeMin	Write Requests	CmdWriteNode	Counts/sec	The number of write command executions per second
Other Requests	StatOtherNodeMin	Other Requests	CmdOtherNode	Counts	The number of command (excluding write and read commands) executions per second

				/s ec	
Slow Logs	SlowQueryNodeMin	Slow Logs	CmdSlowNode	–	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Read Request Hits	StatSuccessNodeMin	Read Request Hits	CmdHitsNode	–	The number of keys successfully requested by read commands, which is equal to the value of the <code>keyspace_hits</code> metric output by the <code>info</code> command.
Read Request Misses	StatMissedNodeMin	Read Request Misses	CmdMissNode	–	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>keyspace_misses</code> metric output by the <code>info</code> command.
Execution Errors	CmdErrNodeMin	Execution Errors	CmdErr	–	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
Read Request Hit Rate	CacheHitRatioNodeMin	Read Request Hit Rate	CmdHitsRatioNode	%	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss. When the access request quantity is 0, the value of this metric will be null.

### Description of parameters in each dimension

Parameter (5–Second)	Dimension Name	Description	Format	Mapping
----------------------	----------------	-------------	--------	---------

Granularity)				
Instances.N.Dimensions.0.Name	instanceid	Dimension name of the instance ID	Enter a string-type dimension name: instanceid.	Unchanged
Instances.N.Dimensions.0.Value	instanceid	Specific instance ID	Enter a specific Redis instance ID, such as <code>tdsql-123456</code> , which can be queried through the <a href="#">DescribeInstances</a> API and can also be an instance string such as <code>crs-ifmymj41</code> .	Unchanged
Instances.N.Dimensions.1.Name	rnodeid	Dimension name of the Redis node ID	Enter a string-type dimension name: rnodeid.	Corresponding 1-minute dimension name: clusterid.
Instances.N.Dimensions.1.Value	rnodeid	Specific Redis node ID	Enter a specific Redis node ID, which can be queried through the <a href="#">DescribeInstanceNodeInfo</a> API.	Corresponding 1-minute dimension name: clusterid.

# Monitoring Upgrade and Alarm Policy Changes

Last updated: 2023-05-23 11:00:17

## Monitoring Granularity

TencentDB for Redis now supports one-minute and five-second monitoring granularities. Since October 2020, monitoring at the five-second granularity has been supported, providing more monitoring metrics and proxy monitoring data. For more information, see [Update Notes of Monitoring at Five-Second Granularity](#).

## Changes of monitoring granularity

### One-minute monitoring granularity

- Instances created before October 20, 2020 only support the [one-minute monitoring granularity](#), but they are gradually upgraded to support the five-second granularity.
- View the monitoring data in the TCOP console: [TCOP console](#) > **TencentDB** > \*\*Redis (1-minute granularity)\*\*.

### Five-second monitoring granularity

- Instances created after October 20, 2020 support both one-minute and [five-second monitoring granularities](#).
- View the monitoring data in the TCOP console: [TCOP console](#) > **TencentDB** > \*\*Redis (5-second granularity)\*\*.

## Notes of monitoring granularity upgrade

To support five-second monitoring granularity, the proxy of your TencentDB for Redis instances needs to be upgraded to the latest version.

Note that the proxy upgrade will cause a short disconnection. The business needs to reconnect to the proxy after the upgrade completes.

1. Upgrade by Tencent Cloud backend: Tencent Cloud is upgrading all instances to support five-second monitoring granularity. You will be notified via SMS, email, or Message Center before the upgrade starts.
2. Upgrade by yourself in the TencentDB console: you can soon manually upgrade instances in the console.
3. After all instances are upgraded, the one-minute monitoring granularity will be no longer supported.

## Changes of monitoring metrics

After the monitoring granularity is narrowed from one minute to five seconds, monitoring metric names are changed and some new metrics are supported, as shown below:

1-minute metric	5-second metric	Description
CpuUsMin	CpuUtil	Average CPU utilization
CpuMaxUs	CpuMaxUtil	The maximum CPU utilization of a node (shard or replica) in an instance
StorageMin	MemUsed	Actually used memory capacity, including the capacity for data and cache
StorageUsMin	MemUtil	The ratio of the actually used memory to the requested total memory
StorageMaxUs	MemMaxUtil	The maximum among all node (shard or replica) memory utilizations in an instance
KeysMin	Keys	The total number of keys (level-1 keys) in instance storage
ExpiredKeysMin	Expired	The number of keys expired in a time window, which is equal to the value of <code>expired_keys</code> output by the <code>info</code> command.
EvictedKeysMin	Evicted	The number of keys evicted in a time window, which is equal to the value of <code>evicted_keys</code> output by the <code>info</code> command.
ConnectionsMin	Connections	The number of TCP connections to an instance
ConnectionsUsMin	ConnectionsUtil	The ratio of the number of TCP connections to the maximum number of connections
InFlowMin	InFlow	Private network inbound traffic
InFlowUs	InBandwidthUtil	The ratio of the actually used private inbound traffic to the maximum traffic
–	InFlowLimit	The number of times inbound traffic triggers a traffic limit
OutFlowMin	OutFlow	Private network outbound traffic
OutFlowUs	OutBandwidthUtil	The ratio of the actually used private outbound traffic to the maximum traffic
–	OutFlowLimit	The number of times outbound traffic triggers a traffic limit
LatencyMin	LatencyAvg	The average execution latency between the proxy and the Redis server

-	LatencyMax	The maximum execution latency between the proxy and the Redis server
-	LatencyP99	The P99 latency between the proxy and the Redis server
LatencyGetMin	LatencyRead	The average execution latency of read commands between the proxy and the Redis server
LatencySetMin	LatencyWrite	The average execution latency of write command between the proxy and the Redis server
LatencyOtherMin	LatencyOther	The average execution latency of commands (excluding write and read commands) between the proxy and the Redis server
QpsMin	Commands	Queries per second
StatGetMin	CmdRead	The number of read command executions per second
StatSetMin	CmdWrite	The number of write command executions per second
StatOtherMin	CmdOther	The number of command (excluding write and read commands) executions per second
BigValueMin	CmdBigValue	The number of executions of requests larger than 32 KB per second
-	CmdKeyCount	The number of keys accessed by a command per second
-	CmdMget	The number of MGET commands executed per second
SlowQueryMin	CmdSlow	The number of command executions with a latency greater than the <code>slowlog-log-slower-than</code> configuration
StatSuccessMin	CmdHits	The number of keys successfully requested by read commands, which is equal to the value of the <code>keyspace_hits</code> metric output by the <code>info</code> command.
StatMissedMin	CmdMiss	The number of keys unsuccessfully requested by read commands, which is equal to the value of the <code>keyspace_misses</code> metric output by the <code>info</code> command.
CmdErrMin	CmdErr	The number of command execution errors. For example, the command does not exist, parameters are incorrect, etc.
CacheHitRatioMin	CmdHitsRatio	Key hits/(key hits + key misses). This metric can reflect the situation of cache miss.

## Viewing the monitoring granularity of an instance

- Check the value of the `InstanceSet.MonitorVersion` field returned by the [DescribeInstances](#) API. If the value is `5s`, this instance supports the monitoring granularity of five seconds; if the value is `1m`, it supports only the monitoring granularity of one minute.
- Log in to the [TencentDB for Redis console](#), click an instance name to enter the instance management page, select **System Monitoring** > **Monitoring Metrics**, and click the **Period** drop-down list at the top. If you can select **5 seconds** from the drop-down list, this instance supports the monitoring granularity of five seconds, or else it supports only the monitoring granularity of one minute.

## Alarm Changes

### Changes of alarm policy configurations

After the monitoring metrics are upgraded, you need to configure one-minute-granularity and five-second-granularity alarm policies in different windows in the [TCOP console](#).

**Basic Information**

Policy Name

Notes

Monitor Type  Cloud Product Monitoring

Policy Type

Project   
**Cloud Database / Redis / Memory Edition / Instance**  
Cloud Database / Redis / Memory Edition / Proxy  
Cloud Database / Redis / Memory Edition / Redis Node

**Configure Alarm**

The current account has dynamic threshold policies, and 20 more can be created.

### Impact of monitoring upgrade

After the monitoring granularity is narrowed from one minute to five seconds, you need to migrate the one-minute-granularity alarm policies to the five-second-granularity alarm policies. The monitoring metrics applicable to the five-second granularity alarm policies are different from those applicable to one-minute granularity alarm policies. For more information, see [Changes of monitoring metrics](#).

After the monitoring granularity is narrowed down to five seconds:

- Monitoring data at both one-minute and five-second granularities are reported temporarily, that is, Cloud Monitor will stop reporting the one-minute-granularity data in the future.
- One-minute-granularity alarm policies are valid temporarily.

- The default five-second-granularity alarm policy is associated. Please specify alarm recipients for the default policy.

## Migrating alarm policies

- Manual migration: copy the existing one-minute-granularity alarm policies as the five-second-granularity alarm policies, but you need to configure alarm recipients for the five-second-granularity alarm policies.
- Automatic migration: after the monitoring granularity upgrade completes, the existing one-minute-granularity alarm policies will be automatically migrated to the five-second-granularity alarm policies, and you will be notified via SMS, email, or Message Center.